STOPPD RCT Analysis Index

Erin Dickie and Navona Calarco

March 19, 2019

Contents

1 The index page

2 Verifying number of scans

2.1 Checking the TIGRLab "/archive/data"

This script pulls in and cleans up the naming of STOPPD scans as they exist in the Kimel lab file system. At earlier stages, this script helped us identify naming errors in the file system (all have since been fixed).

Purpose: The contents of the file system will, in other scripts, be checked against (1) the scans we have in XNAT, to ensure that there are no discrepancies between these databases, and also against (2) our subject inclusion list.

```
library('stringi')
library('stringr')
library('plyr')
library('tidyr')
```

```
#import spreadsheet ('ls' of file system)
terminal <- read.csv('../data/stoppd_NiiFolderContents_2018-01-25.csv', header = TRUE, stringsAsFactors
```

```
#make a new column for site component of ID
terminal$site <- str_sub(terminal$scan_id, 8, 10)

#cut out study and site component from ID (first 11 characters)
terminal$scan_id <- substring(terminal$scan_id, 12)

#make a new column for session component of ID
terminal$session <- str_sub(terminal$scan_id, -2)

#cut out session information from ID (last 3 characters)
terminal$scan_id <- stri_sub(terminal$scan_id, 1, -4)

#make a new column that captures alphabetic component of ID ('R')
terminal$contains_R <- grepl('R', terminal$scan_id, fixed=TRUE) #36 participants

#cut out the 'R' in some participant IDs (indicates repeat for controls)
terminal$scan_id <- gsub("[R]", "", terminal$scan_id)
```

```
#make a 'group' column to capture case vs. control information
terminal$group <- stri_sub(terminal$scan_id, 2, 2) #note: 1 or 2 is patient, 6 is control

#for clarity, change values in 'group' column to labels for clarity
terminal$group[terminal$group == 1] <- "patient"
terminal$group[terminal$group == 2] <- "patient"
terminal$group[terminal$group == 6] <- "control"

#make a variable that combines unique ID and session number
terminal$id_session <- paste(terminal$scan_id, '_', terminal$session, sep='')

#write csv
write.csv(terminal, '../generated_csvs/terminal_clean_2018-01-25.csv', row.names=FALSE)

#cleanup
rm(terminal)
```

2.2 Checking XNAT

This script pulls in and cleans up the naming of STOPPD scans as they exist in XNAT. At earlier stages, this script helped us identify naming errors in XNAT (all have since been fixed).

Purpose: The contents of XNAT will, in other scripts, be checked against (1) the scans we have in our file system, to ensure that there are no discrepancies between these databases, and also against (2) our subject inclusion list.

```
#import spreadsheets (exported from XNAT)
xnat_camh <- read.csv('../data/xnat_records/xnat_cmh_2018-01-25.csv')
xnat_nki <- read.csv('../data/xnat_records/xnat_nki_2018-01-25.csv')
xnat_pitt <- read.csv('../data/xnat_records/xnat_pmc_2018-01-25.csv')
xnat_umass <- read.csv('../data/xnat_records/xnat_umas_2018-01-25.csv')

#combine XNAT spreadsheets, take only ID and date columns
xnat <- Reduce(function(x, y) merge(x, y, all=TRUE), list(xnat_camh, xnat_nki, xnat_pitt, xnat_umass))
xnat <- xnat[c('MR.ID', 'Date') ]

#cleanup
rm (xnat_camh, xnat_nki, xnat_pitt, xnat_umass)

#import spreadsheet of data in file system (made in script 01_STOPPD_terminal)
terminal <- read.csv('../generated_csvs/terminal_clean_2018-01-25.csv')
```

```
#remove all CAMH scans with '00' as timepoint (NOTE: '00' this is a consequence of creative naming to a
xnat$timepoint <- str_sub(xnat$MR.ID, start= -2) #make column with timepoint data
xnat <- xnat[-grep('00', xnat$timepoint),] #remove those with 00

#cut out timepoint info from subject ID string - now meaningless - and remove timepoint column
xnat$MR.ID  <- str_sub(xnat$MR.ID, 1, -4)
xnat <- xnat[, -grep('timepoint', colnames(xnat))]

#cut out study and site info from subject ID string - not needed
xnat$MR.ID <- substring(xnat$MR.ID, 12)
```

```
#make a new column for session component of ID
xnat$session <- str_sub(xnat$MR.ID, -2)
table(xnat$session)
```

```
##
##  00  01  02  03
##  17 222  77   7
```

```
#cut out session from subject ID string - not needed
xnat$MR.ID <- str_sub(xnat$MR.ID, 1, -4)

#make a new column that captures alphabetic component of ID ('R')
xnat$contains_R <- grepl('R', xnat$MR.ID, fixed=TRUE)

#cut out the 'R' in some participant IDs (indicates repeat for controls)
xnat$MR.ID <- gsub("[R]", "", xnat$MR.ID)

#make a variable that combines unique ID and session number
xnat$id_session <- paste(xnat$MR.ID, '_', xnat$session, sep='')

#check for consistency between file system and XNAT
X <- terminal$id_session %in% xnat$id_session
  which(X == FALSE) #identical
```

```
## integer(0)
```

```
Y <- xnat$id_session %in% terminal$id_session
  which(Y == FALSE) #identical
```

```
## integer(0)
```

```
#write csv
write.csv(xnat, '../generated_csvs/xnat_clean_2018-01-25.csv', row.names=FALSE)

#cleanup
rm(terminal, xnat)
```

3 Decoding the Master Scan Log

This script combines information in XNAT/file system (which have already been established to be identical in script 02_STOPPD_xnat) and study logs, and randomization (recently unblinded), into a single, master spreadsheet.

Purpose: the output csv (STOPPD_participantList_2018-11-05.csv) is meant to serve as a master reference sheet for all participants that were randomized (irrespective of scan completion).

This script now also adds a column relating to whether or not the subject is ok for MR analysis (i.e. not excluded for later identified neurological condition)

Note: this script does not remove individuals who failed preprocessing, QC, or should be removed from the dataset for any other reason.

```
library('stringi')
library('plyr')
library('tidyr')
library('stringr')
```

```r
#import spreadsheets
xnat <- read.csv('../generated_csvs/xnat_clean_2018-01-25.csv', stringsAsFactors = FALSE) #generated by
randomization <- read.csv('../data/clinical/randomization.csv', stringsAsFactors = FALSE) #from Judy (S
log <- read.csv('../data/clinical/master_log.csv', fileEncoding="latin1", na.strings=c("","  ","NA", "N/

#transform XNAT df from long to wide format
xnat <- xnat[!names(xnat) %in% c('contains_R', 'id_session')] #remove unnecessary variables
xnat <- reshape(xnat, idvar = "MR.ID", timevar = 'session', direction = "wide")
```

```
## Warning in reshapeWide(data, idvar = idvar, timevar = timevar, varying =
## varying, : multiple rows match for session=1: first taken
```

```
## Warning in reshapeWide(data, idvar = idvar, timevar = timevar, varying =
## varying, : multiple rows match for session=2: first taken
```

```r
colnames(xnat) <- c('subject_id', 'first_date_xnat', 'second_date_xnat', 'third_date_xnat', 'acute_date_

#merge xnat with randomization - will get rid of controls, etc
df <- merge(randomization[c('STUDYID', 'BLINDMED')], xnat, all.x=TRUE, by.x='STUDYID', by.y = 'subject_i

#rename randomization column
colnames(df)[colnames(df)=="BLINDMED"] <- "randomization"

#combine the 'notes' columns in the log file (easier to read for now)
log$Comments.1 <- paste(log$Specify.reason.if.scan.not.completed.1, log$Comments.1)
log$Comments.2 <- paste(log$Specify.reason.if.scan.not.completed.2, log$Comments.2)
log$Comments.3 <- paste(log$Specify.reason.if.scan.not.completed.3, log$Comments.3)

#make subset of log variables from log we want to merge with randomization info
log <- log[c(
  "STOPPD.clinical.Trial.ID.Imaging.ID",
  'Sex',
  'Age',
  "Date.of.randomization...Stop.PD",
  "Date.of.consent.to.imaging.study",
  "If.not.enrolled.to.imaging.study..specify.reason.",
  "Study.day.of.acute.phase.MRI",
  "Scan.completed.Y.N",
  "Date.of.MRI..1" ,
  "Study.week",
  "Scan.completed.Y.N.1",
  "Comments.1",
  "Date.of.MRI..2",
  "Study.week.1",
  "Scan.completed",
  "Comments.2",
  "Date.of.MRI..3",
  "Study.week.2",
  "Scan.completed.1",
  "Comments.3")]

#rename the columns of the variables from log we want to merge with randomization info, for clarity
colnames(log) <- c(
  'subject_id',
  'sex',
```

```
  'age',
  'randomization_date',
  'imaging_consent_date',
  'imaging_nonconsent_reason',
  'acute_date_log',
  'acute_complete_log',
  'first_date_log',
  'first_timepoint_log',
  'first_complete_log',
  'first_notes',
  'second_date_log',
  'second_timepoint_log',
  'second_complete_log',
  'second_notes',
  'third_date_log',
  'third_timepoint_log',
  'third_complete_log',
  'third_notes')

#merge the df and log data
df <- merge(df, log, all.x=TRUE, by.x = 'STUDYID', by.y='subject_id')

#reorder df columns, for clarity
df <- df[c(
  "STUDYID",
  'sex',
  'age',
  "randomization",
  'randomization_date',
  'imaging_consent_date',
  'imaging_nonconsent_reason',
  "acute_date_log",
  "acute_complete_log",
  "acute_date_xnat",
  "first_date_log",
  "first_timepoint_log",
  "first_complete_log",
  'first_notes',
  "first_date_xnat",
  "second_date_log",
  "second_timepoint_log",
  "second_complete_log",
  'second_notes',
  "second_date_xnat",
  "third_date_log",
  "third_timepoint_log",
  "third_complete_log",
  'third_notes',
  "third_date_xnat")]

#make sure dates, etc. are characters (not factors) by converting all factors to characters
i <- sapply(df, is.factor)
df[i] <- lapply(df[i], as.character)
```

```r
#clean up the NA-related values (which exist in the 3 notes columsn, 'first_notes', 'second_notes', 'th
df <- data.frame(lapply(df, function(x) {
      gsub("NA NA", NA, x)
      }))

df <- data.frame(lapply(df, function(x) {
      gsub("NA", NA, x)
      }))

#alter incorrect/unclear values as required
  #acute scan
  df$acute_complete_log <- as.character(df$acute_complete_log)
  df$acute_complete_log[df$acute_complete_log == 'Y'] <- "Yes"
  df$acute_complete_log[df$acute_complete_log == "N" & df$STUDYID == '420043'] <- NA #(replace 'no' wit

  #first scan (replace 'no' with NA, to take care of inconsistent notation)
  df["first_complete_log"] <- lapply(df["first_complete_log"], function(x) {
    gsub("No", NA, x)
    })

  #second scan
    df["second_complete_log"] <- lapply(df["second_complete_log"], function(x) {
    gsub("No", NA, x)
    })

  #third scan
  df$third_timepoint_log[df$third_timepoint_log == "what would be RCT Week 36"] <- "RCT Week 36"

#remove 'day' information from 'acute_date_log' and turn into integer
df$acute_date_log <- sub('\\,.*', '', df$acute_date_log) #strip out day info
df$acute_date_log <- as.numeric(substr(df$acute_date_log, 11, 12)) #remove number, make numeric
names(df)[names(df) == 'acute_date_log'] <- 'acute_week_log' #change name of variable for clarity

#separate timepoint source and week information in 'first_timepoint_log' variable
df <- cbind(df, as.data.frame(matrix(str_split_fixed(df$first_timepoint_log, " Week ", 2), ncol = 2, by
df <- subset(df, select = -first_timepoint_log)
colnames(df)[colnames(df)=="V1"] <- "first_timepoint_log"
colnames(df)[colnames(df)=="V2"] <- "first_week_log"

#remove accidental extra space in character
df$second_timepoint_log <- as.character(df$second_timepoint_log)
df$second_timepoint_log[df$second_timepoint_log == 'Off protocol '] <- 'Off protocol'

#separate timepoint source and week information in 'second_timepoint_log' variable
df <- cbind(df, as.data.frame(matrix(str_split_fixed(df$second_timepoint_log, " Week ", 2), ncol = 2, by
df <- subset(df, select = -second_timepoint_log )
colnames(df)[colnames(df)=="V1"] <- "second_timepoint_log"
colnames(df)[colnames(df)=="V2"] <- "second_week_log"

#recode anything containing 'relapse' in 'second_timepoint_log' variable as simply 'relapse'
df$second_timepoint_log <- as.character(df$second_timepoint_log)
df$second_timepoint_log <- ifelse(grepl('Relapse', df$second_timepoint_log), "Relapse", df$second_timepo
```

```r
#recode anything containing 'Protocol' in 'second_timepoint_log' variable as simply 'off protocol'
df$second_timepoint_log <- ifelse(grepl('Protocol', df$second_timepoint_log), "Off protocol", df$second

#separate timepoint source and week information in 'third_timepoint_log' variable
df <- cbind(df, as.data.frame(matrix(str_split_fixed(df$third_timepoint_log, " Week ", 2), ncol = 2, by
df <- subset(df, select = -third_timepoint_log )
colnames(df)[colnames(df)=="V1"] <- "third_timepoint_log"
colnames(df)[colnames(df)=="V2"] <- "third_week_log"

#compare dates in df that comes from log vs. XNAT (in new column)
df$first_dateDiff <- round(difftime(df$first_date_log, df$first_date_xnat, units = "days"), 2)
df$second_dateDiff <- round(difftime(df$second_date_log, df$second_date_xnat, units = "days"), 2)
df$third_dateDiff <- round(difftime(df$third_date_log, df$third_date_xnat, units = "days"), 2)

#make sure new variables are characters (not factors), and turn blank values into NA
i <- sapply(df, is.factor)
df[i] <- lapply(df[i], as.character)
df[df == ""] <- NA

#calculate the difference in weeks between scan 2 and scan 1 (i.e., calculate 'second week log' when ab
df$dateDiff_first_second <- round(difftime(df$second_date_log, df$first_date_log, units = "weeks"), 0)
df$dateDiff_first_second <- as.numeric(df$dateDiff_first_second) #turn variables into integers
df$first_week_log <- as.numeric(df$first_week_log) #turn variables into integers
df$second_week_log <- ifelse(is.na(df$second_week_log) & !is.na(df$second_timepoint_log), paste(df$datel

#reorder df columns
df <- df[c(
  "STUDYID",
  'sex',
  'age',
  "randomization",
  'randomization_date',
  'imaging_consent_date',
  'imaging_nonconsent_reason',
  "acute_week_log",
  "acute_complete_log",
  "first_date_log",
  "first_timepoint_log",
  "first_week_log",
  "first_complete_log",
  'first_notes',
  "second_date_log",
  "second_timepoint_log",
  "second_week_log",
  "second_complete_log",
  'second_notes',
  "third_date_log",
  "third_timepoint_log",
  "third_week_log",
  "third_complete_log",
  'third_notes'
)]
```

```
#remove '_log' component of all variable names, for clarity
names(df) = gsub(pattern = "_log", replacement = "", x = names(df))
```

3.1 Exclusions from MR analysis and reasons

subject 320032 (PMC): incidental findings more atrophy, should excluded **subject 410012 (CMH)**: another incidental finding, case may have effected longitudinal brain morphomentry

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
df <- df %>% mutate(MR_exclusion = if_else(STUDYID %in% c("320032", "410012"), "Yes", "No"))
```

```
#make a smaller df of minimally necessary information from participants that completed 2 scans, as requ
write.csv(df, '../generated_csvs/STOPPD_masterDF_2018-11-05.csv', row.names=FALSE)

#remove participants that don't have 'Yes' in 'first_complete'
df <- df %>% filter(first_complete == "Yes") #nrow = 88, which is correct

#remove participants that don't have 'Yes' in 'second_complete'
df <-df %>% filter(second_complete == "Yes") #nrow = 74, which is correct

#remove redundant columns
df <- df %>% select(STUDYID, age, sex, randomization, MR_exclusion, first_timepoint, second_timepoint,

df <- df %>% dplyr::rename("offlabel_timepoint" = third_timepoint)
```

```
#write.csv
write.csv(df, '../generated_csvs/STOPPD_participantList_2018-11-05.csv', row.names=FALSE)
```

4 Report Randomization numbers

This script identifies the number of participants in olanzapine vs. placebo by scan timepoint, using the logic of group inclusion that Judy and Dielle provided, and Nick and Aristotle have agreed to.

Note: this script includes data from all participants with data in Judy's master log and our file system. It has not excluded participants on any other basis (e.g., QC fail, processing fail, post-hoc clinical trial ineligibility, etc.)

4.1 Identify baseline scans

First - identify the number of baseline scans (i.e., scans completed at week 20).

```
#count the number of participants that have a 'yes' for 'completed' in "Scan.completed.1"
n_first_complete = sum(na.omit(df$first_complete == "Yes")) #88 participants completed week 20 scan

#for clarity, print the IDs of the N=88 participants that completed week 20 scans
(df %>% filter(first_complete == "Yes"))$STUDYID
```

```
##  [1] 110008 110009 110013 110016 110022 110025 110028 110030 110031 110034
## [11] 120011 120012 120015 120016 120017 120021 120026 210012 210013 210014
## [21] 210017 210020 210022 210024 210026 210030 210033 210036 210038 210042
## [31] 210048 210049 210051 220002 220003 220004 220006 220008 220009 310010
## [41] 310015 310025 310037 310051 310070 320006 320013 320021 320022 320032
## [51] 320041 320042 320043 320045 410004 410008 410009 410010 410011 410012
## [61] 410013 410015 410019 410022 410023 410029 410030 410031 410037 410039
## [71] 410040 410043 410045 410047 420005 420007 420013 420016 420018 420019
## [81] 420020 420023 420029 420032 420039 420042 420043 420044
```

The number of participants who completed their first scan is 88

RANDOMIZATION - as expected, there's no difference in first scan completion between those randomized to O vs. P group

```
#RANDOMIZATION - as expected, there's no difference in first scan completion between those randomized t
(R <- addmargins(table(df$first_complete == 'Yes', df$randomization))) #O = 45; P = 43 (total = 88)
```

```
##
##         O  P Sum
##   TRUE 45 43  88
##   Sum  45 43  88
```

4.2 Identify week 56 scans

Second - identify the number of week 56 scans (i.e., 36 weeks after week 20).

```
#make sure that all the participants that completed week 56 scan also completed week 20
all_second_complete <- all((df$second_complete == "Yes") %in% (df$first_complete== "Yes")) #all TRUE

#count the number of participants that have a 'yes' for 'completed' in "Scan.completed" - but this incl
(n_second_complete <- sum(na.omit(df$second_complete == "Yes"))) #74 completed week 56 scan
```

```
## [1] 74
```

Subject ids of the n = 74 who completed their second scan. Note: it is TRUE that all participants who completed their second scan have baseline data.

```
#for clarity, print the IDs of the N=74 participants that completed week 56 scans
(df %>% filter(second_complete == "Yes"))$STUDYID
```

```
##  [1] 110008 110009 110013 110022 110031 110034 120011 120012 120015 120016
## [11] 120017 120021 120026 210012 210013 210014 210017 210020 210022 210026
## [21] 210030 210033 210038 210042 210049 210051 220002 220003 220004 220006
## [31] 220009 310010 310015 310025 310037 310051 320006 320013 320021 320022
## [41] 320032 320042 320043 320045 410004 410008 410009 410010 410011 410012
## [51] 410013 410015 410019 410022 410023 410029 410030 410031 410037 410039
## [61] 410040 410043 420007 420013 420016 420018 420019 420020 420023 420029
```

Table 1: breakdown of those who where scanned at two timepoints

second_timepoint	n
Off protocol	5
RCT	41
Relapse	28

```
## [71] 420032 420039 420042 420043
```

```
#count how many participants that completed week 56 scan are classified as RCT
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'RCT')) #RCT = 41
```

```
## [1] 41
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'RCT']))) #for cla
```

```
##  [1] 110008 110009 110013 110022 110031 110034 120011 120012 120015 210012
## [11] 210013 210014 210017 210020 210030 210051 220004 310051 320006 320021
## [21] 320032 320042 320043 320045 410004 410008 410010 410013 410022 410023
## [31] 410029 410030 410037 410039 410043 420013 420020 420029 420039 420042
## [41] 420043
```

```
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'Relapse')) #Relapse = 28
```

```
## [1] 28
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'Relapse']))) #for
```

```
##  [1] 120016 120017 120021 120026 210022 210026 210033 210038 210042 210049
## [11] 220002 220003 220006 220009 310010 310015 310025 310037 320013 410009
## [21] 410011 410012 410031 410040 420007 420016 420023 420032
```

```
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'Off protocol')) #Off protocol = 5
```

```
## [1] 5
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'Off protocol'])))
```

```
## [1] 320022 410015 410019 420018 420019
```

```
df %>%
  filter(second_complete == "Yes") %>%
  count(second_timepoint) %>%
  kable(caption = "breakdown of those who where scanned at two timepoints")
```

```
#RANDOMIZATION- look at randomization info for those who completed a second timepoint RCT scan
(R <- addmargins(table(df$second_complete == 'Yes' & df$second_timepoint == 'RCT', df$randomization))) 
```

```
##
##          O  P Sum
##   FALSE 14 24  38
##   TRUE  27 14  41
##   Sum   41 38  79
```

```
df %>%
  filter(second_complete == "Yes") %>%
  count(second_timepoint, randomization) %>%
  kable(caption = "breakdown of those who where scanned at two timepoints, by arm")
```

Table 2: breakdown of those who where scanned at two timepoints, by arm

second_timepoint	randomization	n
Off protocol	O	4
Off protocol	P	1
RCT	O	27
RCT	P	14
Relapse	O	8
Relapse	P	20

Table 3: Breakdown of thrid timepoint off-label scans 8 total

randomization	n
O	3
P	5

4.3 Identify off label scans

Third - identify the number of "off label" scans also at week 56.

```
#make sure timepoint is a character
df$second_timepoint <- as.character(df$second_timepoint)

#count the number of scans completed at *third* timepoint, which are by definition "off label"
n_offlable <- sum(na.omit(df$third_complete == 'Yes')) #8 off-label scans

#for clarity, print the IDs of the N=8 participants that completed off-label scans
(as.vector(na.omit(df$STUDYID[df$third_complete == "Yes"])))
```

```
## [1] 110016 210033 210049 220006 310037 320022 410019 420032
```

```
#of these, determine how many "off protocol" vs. "relapse", based on second timepoint scan
sum(na.omit(df$third_complete == 'Yes' & df$second_timepoint  == 'Off protocol')) #2 "off protocol" sca
```

```
## [1] 2
```

```
  (as.vector(na.omit(df$STUDYID[df$third_complete == "Yes" & df$second_timepoint  == 'Off protocol'])))
```

```
## [1] 320022 410019
```

```
sum(na.omit(df$third_complete == 'Yes' & df$second_timepoint == 'Relapse')) #6 relapse scans
```

```
## [1] 6
```

```
  (as.vector(na.omit(df$STUDYID[df$third_complete == "Yes" & df$second_timepoint  == 'Relapse'])))
```

```
## [1] 110016 210033 210049 220006 310037 420032
```

```
#RANDOMIZATION
df %>%
  filter(third_complete == "Yes") %>%
  count(randomization) %>%
  kable(caption = str_c("Breakdown of thrid timepoint off-label scans ", n_offlable, " total"))
```

```
df %>%
  filter(df$second_timepoint == 'Off protocol') %>%
  count(randomization, third_complete) %>%
  kable(caption = str_c("Breakdown of off-protocol scans by presence of third timepoint"))
```

Table 4: Breakdown of off-protocol scans by presence of third timepoint

randomization	third_complete	n
O	Yes	2
O	NA	2
P	NA	1

Table 5: Breakdown of thrid timepoint 'Relapse' scans by presence of third timepoint

randomization	third_complete	n
O	Yes	1
O	NA	9
P	Yes	5
P	NA	18

```
df %>%
  filter(df$second_timepoint == 'Relapse') %>%
  count(randomization, third_complete) %>%
  kable(caption = str_c("Breakdown of thrid timepoint 'Relapse' scans by presence of third timepoint"))
```

4.4 Identify "Relapse" Scans

Identify the scans completed between week 20 and week 56 which are the relapse scans (and in a small minority of cases may be a scan when somebody is moving or wants out of the study despite being well).

```
#count relapse - note: both 'relapse' and 'off protocol' is included here (everything other than 'RCT')
sum(na.omit((df$second_timepoint == 'Relapse' | df$second_timepoint == 'Off protocol') & df$second_compl
```

```
## [1] 33
```

```
#of these, count how many were "relapse" and how many were "off protocol"
sum(na.omit(df$second_timepoint == 'Relapse' & df$second_complete == 'Yes'))# 28 relapse
```

```
## [1] 28
```

```
sum(na.omit(df$second_timepoint == 'Off protocol' & df$second_complete == 'Yes'))#5 off protocol
```

```
## [1] 5
```

```
#RANDOMIZATION
(R <- addmargins(table((df$second_timepoint == 'Relapse' | df$second_timepoint == 'Off protocol') & df$s
```

```
##
##          O  P Sum
##   FALSE 28 15  43
##   TRUE  12 21  33
##   Sum   40 36  76
```

```
(R <- addmargins(table(df$second_timepoint == 'Relapse' & df$second_complete == 'Yes', df$randomization
```

```
##
##          O  P Sum
##   FALSE 32 16  48
##   TRUE   8 20  28
##   Sum   40 36  76
```

```
(R <- addmargins(table(df$second_timepoint == 'Off protocol' & df$second_complete == 'Yes', df$randomiza
```

```
##
##          0  P Sum
##   FALSE 38 38  76
##   TRUE   4  1   5
##   Sum   42 39  81
```

```
rm(df, R)
```

5 Mangle Freesurfer Outputs

This script pulls together completion information alongside cortical thickness (CT) values and demographic information, for statistical purposes (error calculations). It is required for subsequent CT analyses. It was made in preparation for, and discussed at, the meeting with Jason Lerch.

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------
```

```
## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.2.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
df <- read_csv("../generated_csvs/STOPPD_masterDF_2018-11-05.csv",na = "empty") #spreadsheet created by
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   STUDYID = col_integer()
## )
```

```
## See spec(...) for full column specifications.
```

```
CT <- read_csv('../data/fs-enigma-long_201811/CorticalMeasuresENIGMA_ThickAvg.csv') #bring in CT data,
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   SubjID = col_character()
## )
## See spec(...) for full column specifications.
```

```
# remove participants that did not complete first and second scan (n=74)
# then add offlabel and dateDiff (in days columns)
# + a scan is by definition offlabel if it is the third scan
# then select the cols for analysis
df <- df %>%
  filter(first_complete == "Yes",
         second_complete == "Yes",
         MR_exclusion == "No") %>%
```

```
  mutate(offLabel  = if_else(third_complete == "Yes", "Yes", ''),
         dateDiff = round(difftime(second_date, first_date, units = "days"), 0),
         STUDYID = parse_character(STUDYID)) %>%
  rename(category = "second_timepoint") %>%
  select(STUDYID, randomization, sex, age, category, offLabel, dateDiff)
```

5.1 cleaning the CT data

```
# separating the subject id and anything afterwards to identify the longtudinal pipeline participants
# separating the subject id into site, "STUDYID" and timepoint columns
# filtering (two steps) to only include the longitudinal pipeline data
CT_long <- CT %>%
  separate(SubjID, into = c("subid", "longitudinal_pipe"), sep = '\\.', extra = "drop", fill = "right") %>%
  separate(subid, into = c("study", "site", "STUDYID", "timepoint"), fill = "right") %>%
  filter(longitudinal_pipe == "long") %>%
  filter(timepoint != "00", timepoint != "03", timepoint != "")


# move CT from long to wide format
CT_wide <- CT_long %>%
  gather(region, thickness, ends_with('thickavg'), LThickness, RThickness, LSurfArea, RSurfArea, ICV) %>%
  spread(timepoint, thickness) %>%
  mutate(change = `02` - `01`) %>%
  gather(timepoint, thickness, `01`, `02`, change) %>%
  unite(newcolnames, region, timepoint) %>%
  spread(newcolnames, thickness)
```

```
# merge CT values with df
ana_df <- inner_join(df, CT_wide, by='STUDYID')

# write.csv
write_csv(ana_df, '../generated_csvs/STOPPD_participantsCT_20181111.csv')
```

5.2 report any mising values from clinical trial sample

```
anti_join(df, CT_wide, by='STUDYID') %>%
  summarise(`Number of participants missing` = n()) %>%
  knitr::kable()
```

Number of participants missing
0

```
ana_df %>%
  filter(is.na(LThickness_01)) %>%
  summarise(`Number of participants missing timepoint 01` = n()) %>%
  knitr::kable()
```

Number of participants missing timepoint 01
0

```
ana_df %>%
  filter(is.na(LThickness_02)) %>%
```

```
  summarise(`Number of participants missing timepoint 02` = n()) %>%
  knitr::kable()
```

Number of participants missing timepoint 02
0

5.3 creating an control error term calculating spreadsheet

```
## identify the repeat control in a column and mangle the STUDYID to match in a new column
CT_long1 <- CT_long %>%
  mutate(repeat_run = if_else(str_sub(STUDYID,1,1)=="R", "02", "01"),
         STUDYID = str_replace(STUDYID, 'R',""))

## extra the repeat study ids as a character vector
repeat_ids <- filter(CT_long1, repeat_run == "02")$STUDYID

## filter for only the subjects who are in the repeats list then switch to wide format
CT_wide_controls <- CT_long1 %>%
  filter(STUDYID %in% repeat_ids) %>%
  gather(region, thickness, ends_with('thickavg'), LThickness, RThickness, LSurfArea, RSurfArea, ICV) %
  unite(newcolnames, region, repeat_run) %>%
  spread(newcolnames, thickness)

#write.csv
  write.csv(CT_wide_controls, '../generated_csvs/STOPPD_errorControls_2018-11-05.csv', row.names = FALS
```

6 Cortical Thickness Analysis

This section runs the stats for average (by hemisphere) Cortical Thickness calculated with Freesurfer

```
#load libraries
library(tidyverse)
library(broom)
library(lmerTest)
library(tableone)


#bring in data
df <- read_csv('../generated_csvs/STOPPD_participantsCT_20181111.csv') #generated by 05_STOPPD_error in
```

```
#make sure that STUDYID is an interger not a number
  df$STUDYID <- as.character(df$STUDYID)

#make sure that dateDiff is a number, not an interger
  df$dateDiff <- as.numeric(df$dateDiff)

# label the randomization variable
df$RandomArm <- factor(df$randomization,
                       levels = c("O", "P"),
                       labels = c("Olanzapine", "Placebo"))

RandomArmColors = c( "#FFC200", "#007aa3")
```

```r
# set category levels so that RCT and Relapse are at the top
df <- df %>%
  mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")))

#restructure data for RCT completers' only (N=40)
  RCT_CT <- df %>%
    filter(category == "RCT")



#write out clean dataframe
 # write.csv(RCT_CT, '../generated_data/df_leftCT.csv', row.names=FALSE)
```

6.0.0.1 baseline measures (table1 part 1)

```r
CreateTableOne(data = df,
               strata = "randomization",
               vars = c("category", "LThickness_01", "RThickness_01", "LSurfArea_01", "RSurfArea_01"))
```

```
##                            Stratified by randomization
##                             0                  P                  p
##   n                               38                 34
##   category (%)                                                   0.008
##      RCT                          26 (68.4)          14 (41.2)
##      Relapse                       8 (21.1)          19 (55.9)
##      Off protocol                  4 (10.5)           1 ( 2.9)
##   LThickness_01 (mean (sd))     2.41 (0.09)        2.41 (0.11)      0.927
##   RThickness_01 (mean (sd))     2.42 (0.08)        2.40 (0.11)      0.450
##   LSurfArea_01 (mean (sd))  83584.98 (9259.48) 81101.75 (10210.53)  0.283
##   RSurfArea_01 (mean (sd))  83263.46 (9340.81) 81436.34 (10146.92)  0.429
##                            Stratified by randomization
##                             test
##   n
##   category (%)
##      RCT
##      Relapse
##      Off protocol
##   LThickness_01 (mean (sd))
##   RThickness_01 (mean (sd))
##   LSurfArea_01 (mean (sd))
##   RSurfArea_01 (mean (sd))
```

```r
CreateTableOne(data = df,
               vars = c("category", "LThickness_01", "RThickness_01", "LSurfArea_01", "RSurfArea_01"))
```

```
##
##                             Overall
##   n                               72
##   category (%)
##      RCT                          40 (55.6)
##      Relapse                      27 (37.5)
##      Off protocol                  5 ( 6.9)
##   LThickness_01 (mean (sd))     2.41 (0.10)
##   RThickness_01 (mean (sd))     2.41 (0.10)
```

Table 6: t.test for baseline group differences

thick	estimate	estimate1	estimate2	statistic	p.value	parameter	conf.low	
LSurfArea_01	2483.2345201	83584.981579	81101.747059	1.0763573	0.2856251	67.05205	-2121.6362282	70
LThickness_01	0.0021686	2.412625	2.410456	0.0908530	0.9278969	63.17323	-0.0455277	
RSurfArea_01	1827.1199690	83263.455263	81436.335294	0.7918366	0.4312325	67.43641	-2778.0130382	64
RThickness_01	0.0171918	2.416510	2.399318	0.7478608	0.4573935	61.51826	-0.0287677	

```
##   LSurfArea_01 (mean (sd))  82412.34 (9731.16)
##   RSurfArea_01 (mean (sd))  82400.65 (9703.97)
```

6.0.0.2 baseline stats (part 2)

```
df %>%
  select(randomization, LThickness_01, RThickness_01, LSurfArea_01, RSurfArea_01) %>%
  gather(thick, mm, -randomization) %>%
  group_by(thick) %>%
  do(tidy(t.test(mm~randomization, data = .))) %>%
  knitr::kable(caption = "t.test for baseline group differences")
```

6.1 RCT only

```
RCT_CT %>% count(randomization) %>% knitr::kable()
```

randomization	n
O	26
P	14

```
fit_all <- lmer(LThickness_change ~ RandomArm + sex + age + (1|site), data= RCT_CT)
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LThickness_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_CT
##
## REML criterion at convergence: -143.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.89976 -0.57933  0.06257  0.56839  1.99100
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  site     (Intercept) 0.0001067 0.01033
##  Residual             0.0006323 0.02515
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                     Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      -0.0116931  0.0176082 28.2935474  -0.664    0.512
## RandomArmPlacebo  0.0401890  0.0085744 34.3857775   4.687 4.26e-05 ***
## sexM              0.0113835  0.0082471 34.2168472   1.380    0.176
```

```
## age              -0.0003708  0.0003075 35.9997290  -1.206    0.236
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.007
## sexM        -0.066  0.076
## age         -0.881 -0.194 -0.178
```

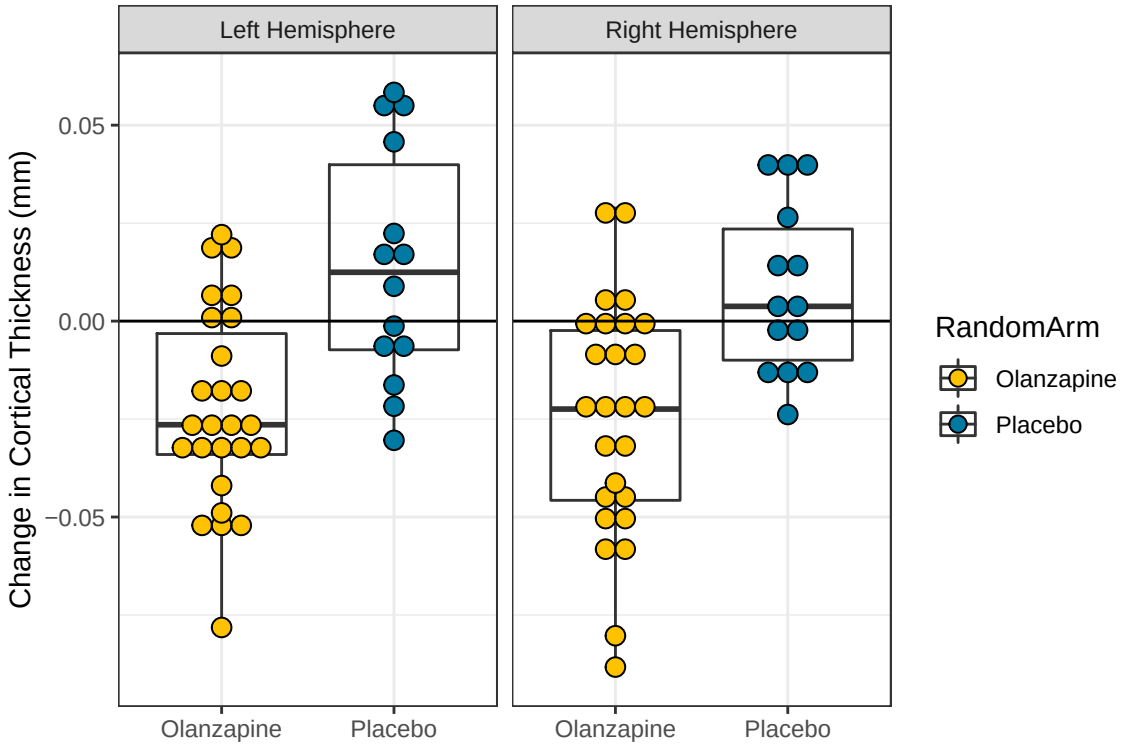
```
fit_all <- lmer(RThickness_change ~ RandomArm + sex + age + (1|site), data= RCT_CT)
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RThickness_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_CT
##
## REML criterion at convergence: -139
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4823 -0.6212 -0.0391  0.6947  1.7575
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  site     (Intercept) 1.569e-05 0.003961
##  Residual             7.557e-04 0.027490
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                     Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      -0.0112925  0.0176225 28.1556699  -0.641 0.526835
## RandomArmPlacebo  0.0338477  0.0092972 35.0818648   3.641 0.000869 ***
## sexM             -0.0039112  0.0089510 35.0847652  -0.437 0.664819
## age              -0.0002273  0.0003235 35.3460195  -0.703 0.486881
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.020
## sexM        -0.050  0.070
## age         -0.915 -0.184 -0.196
```

6.1.1 looking at the same thing for Right CT

```
#boxplot of difference in thickness (y axis) by randoMR_exclusion == "No"mization group (x axis)
RCT_CT %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere"))) %>%
ggplot(aes(x= RandomArm, y = mm, fill = RandomArm)) +
```

```
    geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
    geom_dotplot(binaxis = 'y', stackdir = 'center', binwidth = 0.005) +
    geom_hline(yintercept = 0) +
    labs(x = NULL, y = "Change in Cortical Thickness (mm)") +
    scale_fill_manual(values = RandomArmColors) +
    scale_shape_manual(values = c(21)) +
    facet_wrap(~ ThickChange) +
    theme_bw()
```



6.2 RCT & Relapse (with time as factor)

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_LCT <- df %>%
    gather(thick_oldcolname, thickness, LThickness_01, LThickness_02) %>%
    mutate(model_days = if_else(thick_oldcolname == "LThickness_01", 1, dateDiff)) %>%
    mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")),
           hemi = "Left Hemisphere")

RCTRelapse_LCT %>% filter(model_days == 1) %>% count(RandomArm, offLabel) %>% knitr::kable()
```

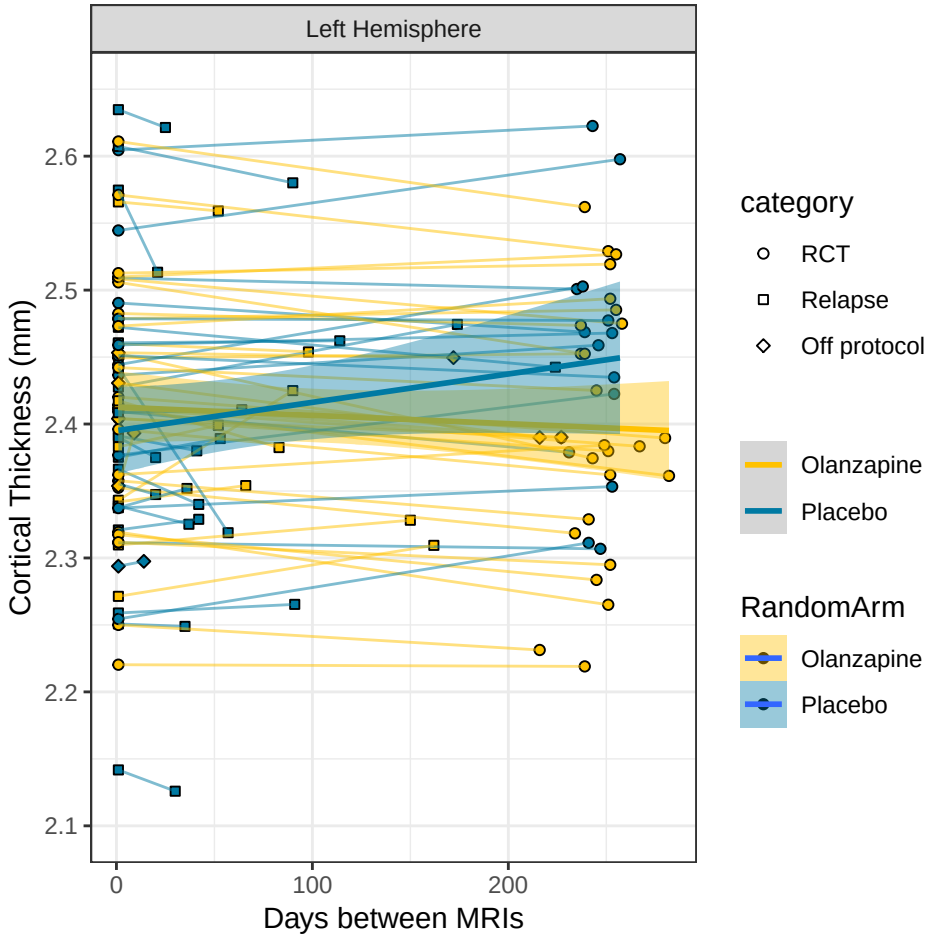
RandomArm	offLabel	n
Olanzapine	Yes	3
Olanzapine	NA	35
Placebo	Yes	4
Placebo	NA	30

```
RCTRelapse_LCT_sensitivety <- RCTRelapse_LCT %>%
  filter(category != "Off protocol" )
```

```
RCTRelapse_LCT_sensitivety %>% filter(model_days == 1) %>% count(RandomArm, offLabel) %>% knitr::kable()
```

RandomArm	offLabel	n
Olanzapine	Yes	1
Olanzapine	NA	33
Placebo	Yes	4
Placebo	NA	29

```
RCTRelapse_LCT %>%
  ggplot(aes(x=model_days, y=thickness, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  labs(x = "Days between MRIs", y = "Cortical Thickness (mm)", colour = NULL) +
  scale_color_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  scale_y_continuous(limits = c(2.1,2.65)) +
  theme_bw()  +
  facet_wrap(~hemi)
```




```
#run mixed linear model, with covariates
  fit_all <- lmer(thickness ~ RandomArm*model_days + age + sex + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: thickness ~ RandomArm * model_days + age + sex + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_LCT
##
## REML criterion at convergence: -396
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.89073 -0.39603 -0.02082  0.40944  2.76834
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  STUDYID  (Intercept) 0.0054535 0.07385
##  site     (Intercept) 0.0000000 0.00000
##  Residual             0.0004953 0.02225
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 2.639e+00  3.484e-02  6.864e+01  75.756
## RandomArmPlacebo           -2.035e-03  1.816e-02  7.233e+01  -0.112
## model_days                 -8.012e-05  2.340e-05  7.056e+01  -3.424
## age                        -4.053e-03  5.853e-04  6.785e+01  -6.924
## sexM                       -6.099e-03  1.792e-02  6.784e+01  -0.340
## RandomArmPlacebo:model_days  1.297e-04  3.942e-05  7.148e+01   3.291
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo             0.91106
## model_days                   0.00103 **
## age                         1.96e-09 ***
## sexM                         0.73470
## RandomArmPlacebo:model_days  0.00155 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy age    sexM
## RndmArmPlcb -0.206
## model_days  -0.077  0.133
## age         -0.901 -0.053  0.008
## sexM        -0.171  0.036  0.001 -0.079
## RndmArmPl:_  0.043 -0.176 -0.594 -0.003  0.004
```

```
#run mixed linear model, with covariates
  fit_all <- lmer(thickness ~ RandomArm*model_days + age + sex + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: thickness ~ RandomArm * model_days + age + sex + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_LCT_sensitivety
##
```

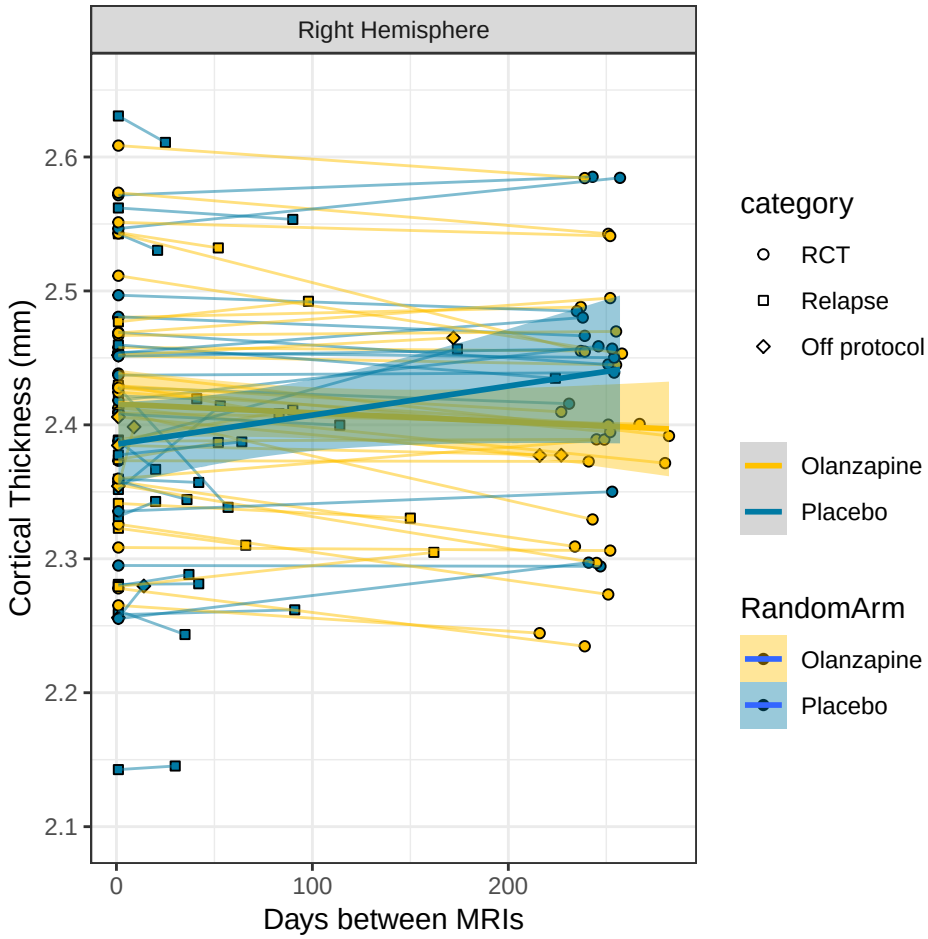
```
## REML criterion at convergence: -362.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.83429 -0.38168 -0.01433  0.38915  2.71136
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  STUDYID  (Intercept) 5.347e-03 7.313e-02
##  site     (Intercept) 1.702e-17 4.126e-09
##  Residual             5.157e-04 2.271e-02
## Number of obs: 134, groups:  STUDYID, 67; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 2.651e+00  3.527e-02  6.373e+01  75.157
## RandomArmPlacebo            2.523e-03  1.865e-02  6.737e+01   0.135
## model_days                 -7.971e-05  2.477e-05  6.544e+01  -3.218
## age                        -4.352e-03  6.040e-04  6.285e+01  -7.205
## sexM                       -6.065e-04  1.854e-02  6.285e+01  -0.033
## RandomArmPlacebo:model_days  1.293e-04  4.076e-05  6.632e+01   3.172
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo             0.89280
## model_days                   0.00201 **
## age                         8.88e-10 ***
## sexM                         0.97402
## RandomArmPlacebo:model_days  0.00229 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy age    sexM
## RndmArmPlcb -0.211
## model_days  -0.084  0.141
## age         -0.899 -0.052  0.011
## sexM        -0.115 -0.008 -0.003 -0.126
## RndmArmPl:_  0.051 -0.184 -0.608 -0.008  0.009
```

6.2.1 Running the right hemisphere RCTRelapse

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_RCT <- df %>%
    gather(thick_oldcolname, thickness, RThickness_01, RThickness_02) %>%
    mutate(model_days = if_else(thick_oldcolname == "RThickness_01", 1, dateDiff)) %>%
    mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")),
           hemi = "Right Hemisphere")

RCTRelapse_RCT_sensitivety <- RCTRelapse_RCT %>%
  filter(category != "Off protocol" )
```

```
RCTRelapse_RCT %>%
  ggplot(aes(x=model_days, y=thickness, fill = RandomArm)) +
```

```
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  labs(x = "Days between MRIs", y = "Cortical Thickness (mm)", colour = NULL) +
  scale_color_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  scale_y_continuous(limits = c(2.1,2.65)) +
  theme_bw()  +
  facet_wrap(~hemi)
```




```
#run mixed linear model, with covariates
  fit_all <- lmer(thickness ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: thickness ~ RandomArm * model_days + sex + age + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_RCT
##
## REML criterion at convergence: -409
##
## Scaled residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -2.34720 -0.42608 -0.01215  0.43733  2.27881
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  STUDYID  (Intercept) 0.0057442 0.07579
##  site     (Intercept) 0.0000000 0.00000
##  Residual             0.0003947 0.01987
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 2.618e+00  3.554e-02  6.847e+01  73.658
## RandomArmPlacebo           -1.455e-02  1.847e-02  7.131e+01  -0.788
## model_days                 -8.813e-05  2.090e-05  7.041e+01  -4.216
## sexM                       -7.789e-03  1.830e-02  6.786e+01  -0.426
## age                        -3.588e-03  5.975e-04  6.786e+01  -6.004
## RandomArmPlacebo:model_days  1.281e-04  3.524e-05  7.112e+01   3.635
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo            0.433361
## model_days                  7.28e-05 ***
## sexM                        0.671706
## age                         8.40e-08 ***
## RandomArmPlacebo:model_days 0.000522 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.205
## model_days  -0.067  0.117
## sexM        -0.172  0.036  0.001
## age         -0.902 -0.053  0.007 -0.079
## RndmArmPl:_  0.037 -0.155 -0.593  0.003 -0.002
```

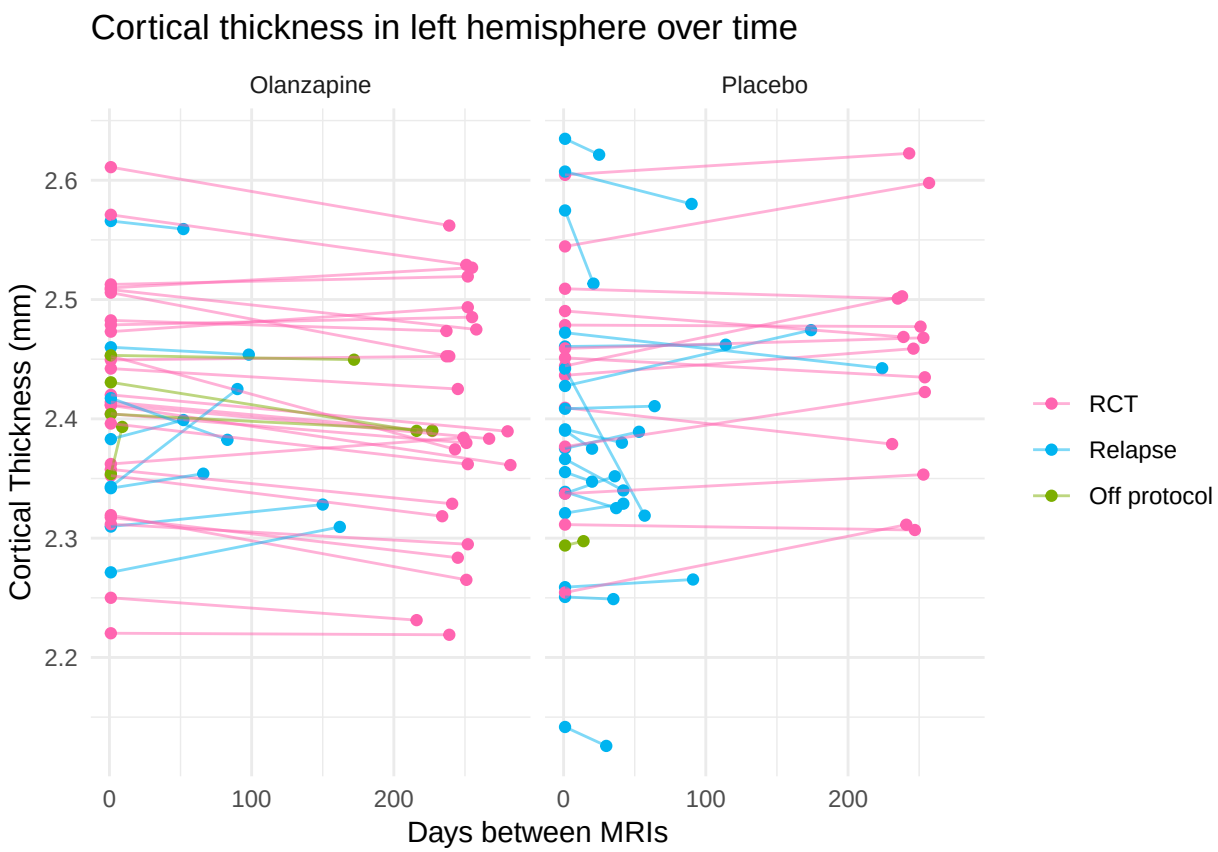
```
#run mixed linear model, with covariates
  fit_all <- lmer(thickness ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: thickness ~ RandomArm * model_days + sex + age + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_RCT_sensitivety
##
## REML criterion at convergence: -376.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.33892 -0.43051 -0.02586  0.45841  2.27007
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  STUDYID  (Intercept) 0.0056833 0.07539
```

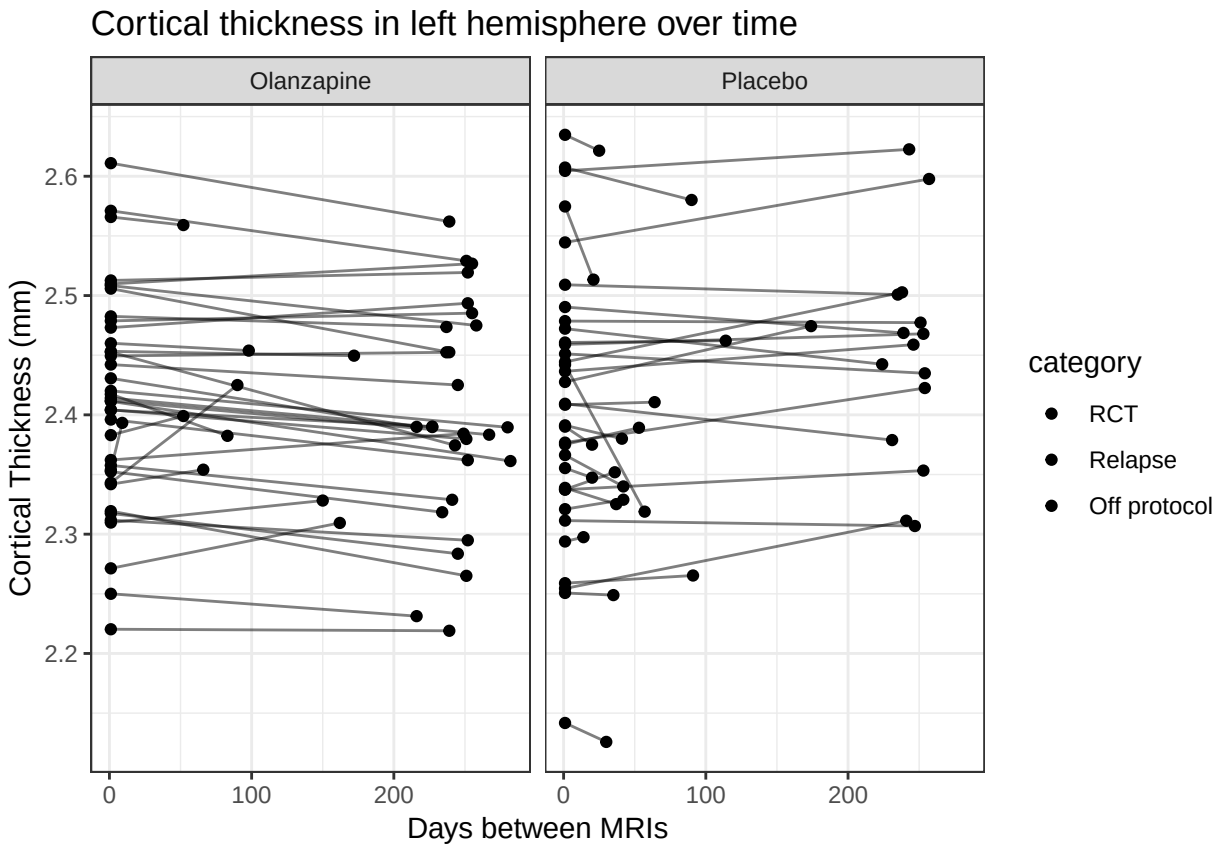
```
##  site     (Intercept) 0.0000000 0.00000
##  Residual             0.0003973 0.01993
## Number of obs: 134, groups:  STUDYID, 67; site, 4
##
## Fixed effects:
##                               Estimate Std. Error         df t value
## (Intercept)                  2.628e+00  3.609e-02  6.353e+01  72.807
## RandomArmPlacebo            -1.094e-02  1.902e-02  6.621e+01  -0.575
## model_days                  -9.212e-05  2.176e-05  6.532e+01  -4.234
## sexM                        -2.307e-03  1.900e-02  6.288e+01  -0.121
## age                         -3.829e-03  6.187e-04  6.288e+01  -6.188
## RandomArmPlacebo:model_days  1.313e-04  3.583e-05  6.596e+01   3.664
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo            0.567192
## model_days                  7.33e-05 ***
## sexM                        0.903719
## age                         5.10e-08 ***
## RandomArmPlacebo:model_days 0.000496 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.209
## model_days  -0.072  0.121
## sexM        -0.115 -0.008 -0.002
## age         -0.900 -0.052  0.009 -0.126
## RndmArmPl:_  0.043 -0.158 -0.607  0.008 -0.007
```

6.2.2 Playing with other ways to layout the plots

```
RCTRelapse_LCT %>%
  ggplot(aes(x=model_days, y=thickness, colour = category)) +
  geom_point() +
  geom_line(aes(group=STUDYID), alpha = 0.5) +
  ggtitle("Cortical thickness in left hemisphere over time") +
  labs(x = "Days between MRIs", y = "Cortical Thickness (mm)", colour = NULL) +
  theme_minimal() + facet_wrap(~ RandomArm) +
  scale_colour_discrete(direction = -1)
```



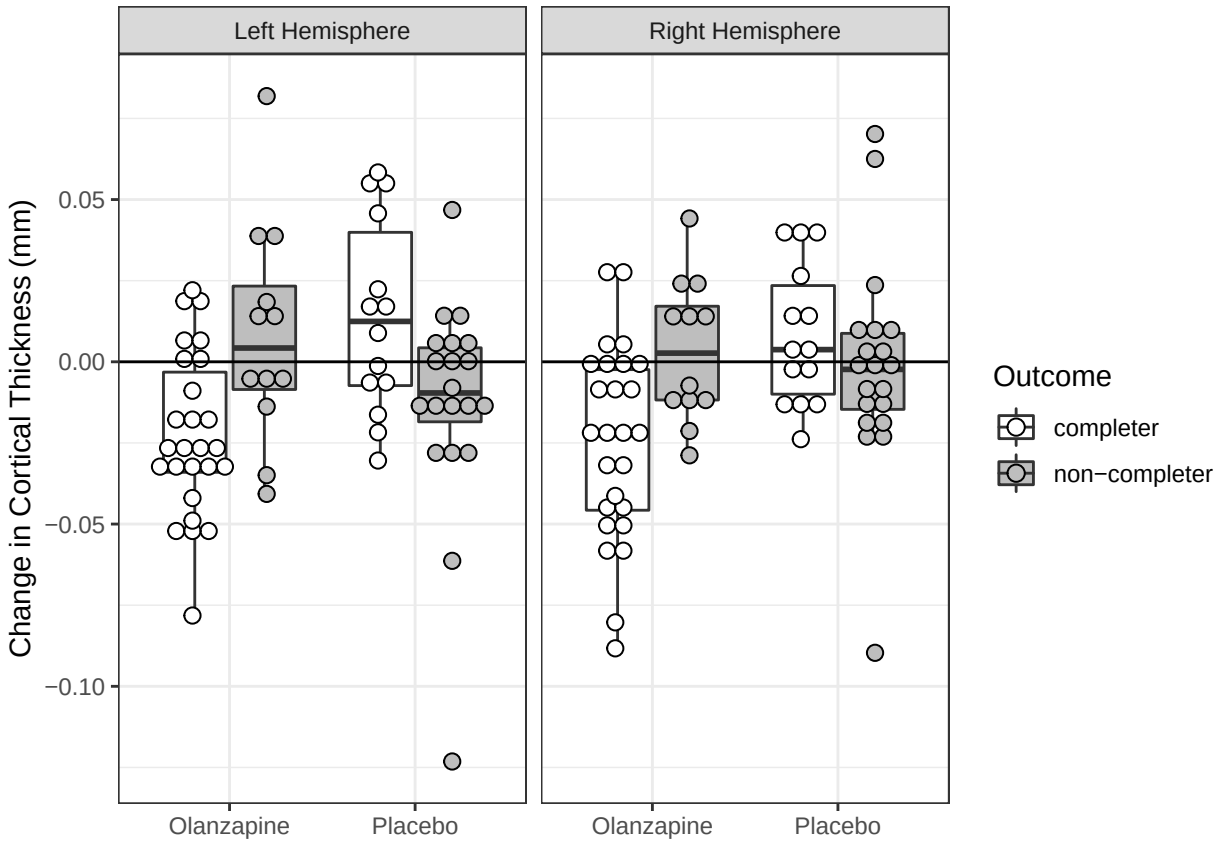

```
RCTRelapse_LCT %>%
  mutate(Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  ggplot(aes(x=model_days, y=thickness, fill = category)) +
  geom_point() +
  geom_line(aes(group=STUDYID), alpha = 0.5) +
  ggtitle("Cortical thickness in left hemisphere over time") +
  labs(x = "Days between MRIs", y = "Cortical Thickness (mm)", colour = NULL) +
  theme_bw() + facet_wrap(~ RandomArm) +
  scale_colour_manual(values = c("black", "white", "grey"))
```



6.3 Post-Hoc - looking at subgroups

plotting change for all participants

```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  ggplot(aes(x= RandomArm, y = mm, fill = Outcome)) +
     geom_boxplot(outlier.shape = NA) +
     geom_dotplot(binaxis = 'y', stackdir = 'center',
                  position=position_dodge(0.8), binwidth = 0.005) +
     geom_hline(yintercept = 0) +
     xlab(NULL) +
     ylab("Change in Cortical Thickness (mm)") +
     theme_bw() +
     scale_fill_manual(values = c('white','grey')) +
     facet_wrap(~ThickChange)
```




```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  filter(category != "Off protocol") %>%
  group_by(ThickChange, RandomArm, category) %>%
  do(tidy(t.test(.$mm, mu = 0, alternative = "two.sided"))) %>%
  knitr::kable(digits = 3)
```

ThickChange	RandomArm	category	estimate	statistic	p.value	parameter	conf.low	conf.high	method
Left Hemisphere	Olanzapine	RCT	-0.023	-4.542	0.000	25	-0.033	-0.012	One Sa
Left Hemisphere	Olanzapine	Relapse	0.015	1.208	0.266	7	-0.014	0.044	One Sa
Left Hemisphere	Placebo	RCT	0.014	1.754	0.103	13	-0.003	0.031	One Sa
Left Hemisphere	Placebo	Relapse	-0.013	-1.678	0.111	18	-0.030	0.003	One Sa
Right Hemisphere	Olanzapine	RCT	-0.024	-4.156	0.000	25	-0.036	-0.012	One Sa
Right Hemisphere	Olanzapine	Relapse	0.002	0.380	0.715	7	-0.013	0.018	One Sa
Right Hemisphere	Placebo	RCT	0.008	1.428	0.177	13	-0.004	0.021	One Sa
Right Hemisphere	Placebo	Relapse	-0.003	-0.363	0.721	18	-0.019	0.013	One Sa

6.4 Exploratory within Treatment Arm tests

```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
```

```
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  filter(category != "Off protocol") %>%
  group_by(ThickChange, RandomArm) %>%
  do(tidy(t.test(mm~category, var.equal = FALSE, data = .))) %>%
  knitr::kable(digits = 3)
```

ThickChange	RandomArm	estimate	estimate1	estimate2	statistic	p.value	parameter	conf.low	conf.hi
Left Hemisphere	Olanzapine	-0.037	-0.023	0.015	-2.822	0.019	9.402	-0.067	-0.0
Left Hemisphere	Placebo	0.027	0.014	-0.013	2.427	0.021	30.107	0.004	0.0
Right Hemisphere	Olanzapine	-0.027	-0.024	0.002	-3.061	0.006	19.342	-0.045	-0.0
Right Hemisphere	Placebo	0.011	0.008	-0.003	1.158	0.256	30.715	-0.008	0.0

```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  filter(category != "Off protocol") %>%
  group_by(ThickChange, RandomArm) %>%
  do(tidy(lm(mm~category + age + sex + site, var.equal = FALSE, data = .))) %>%
  filter(term == "categoryRelapse") %>%
  knitr::kable(digits = 3)
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'var.equal' will be disregarded

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'var.equal' will be disregarded

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'var.equal' will be disregarded

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'var.equal' will be disregarded
```

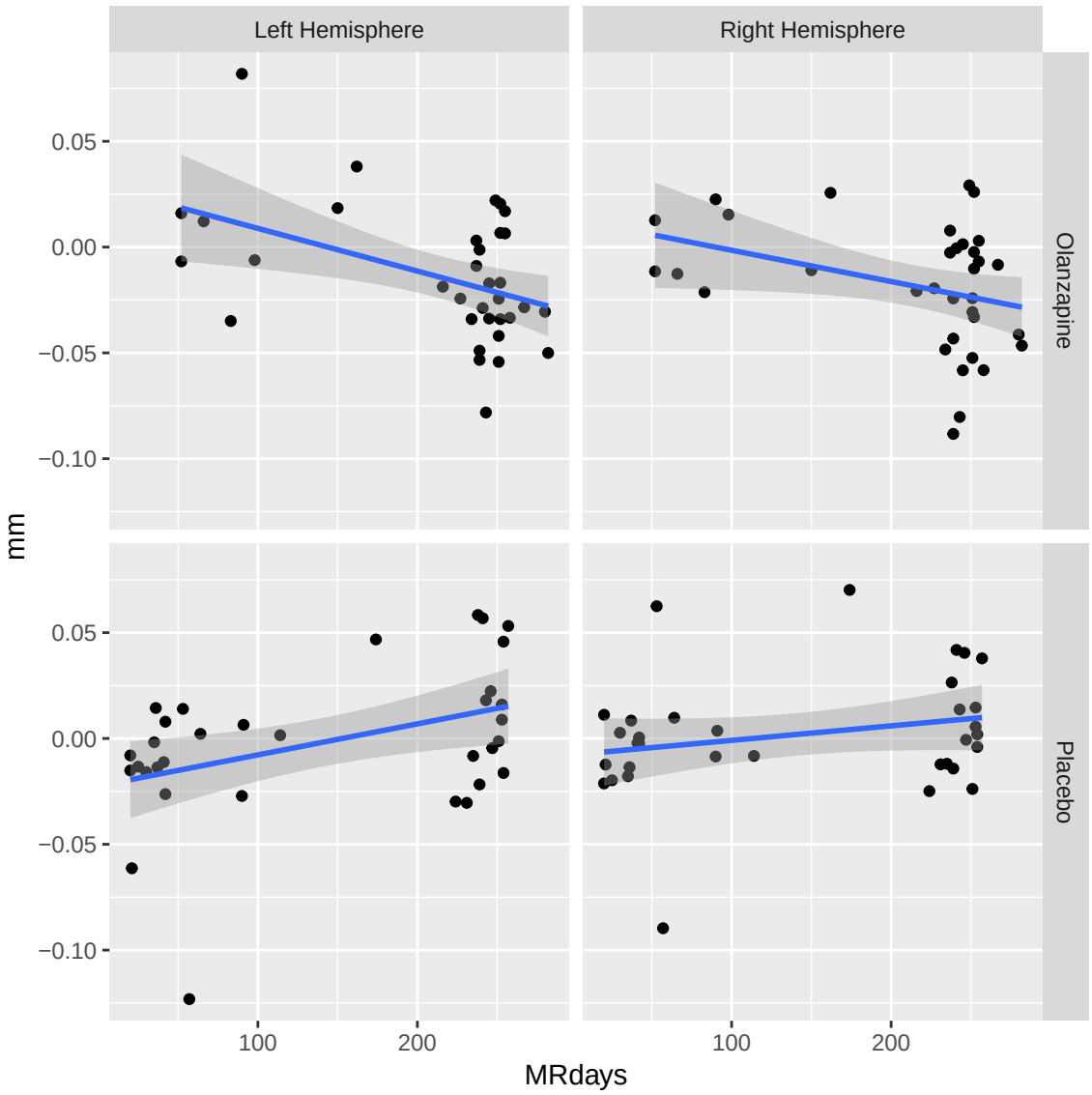
ThickChange	RandomArm	term	estimate	std.error	statistic	p.value
Left Hemisphere	Olanzapine	categoryRelapse	0.043	0.013	3.368	0.002
Left Hemisphere	Placebo	categoryRelapse	-0.025	0.011	-2.327	0.028
Right Hemisphere	Olanzapine	categoryRelapse	0.032	0.013	2.549	0.017
Right Hemisphere	Placebo	categoryRelapse	-0.009	0.010	-0.915	0.369

```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         MRdays = as.numeric(dateDiff)) %>%
  filter(category != "Off protocol") %>%
  group_by(ThickChange, RandomArm) %>%
  do(tidy(lm(mm~MRdays, data = .))) %>%
```

```
  filter(term == "MRdays") %>%
  knitr::kable(digits = 3)
```

ThickChange	RandomArm	term	estimate	std.error	statistic	p.value
Left Hemisphere	Olanzapine	MRdays	0	0	-2.832	0.008
Left Hemisphere	Placebo	MRdays	0	0	2.567	0.015
Right Hemisphere	Olanzapine	MRdays	0	0	-2.106	0.043
Right Hemisphere	Placebo	MRdays	0	0	1.373	0.179

```
df %>%
  gather(TCT, mm, LThickness_change, RThickness_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LThickness_change", "RThickness_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere")),
         MRdays = as.numeric(dateDiff)) %>%
  filter(category != "Off protocol") %>%
  ggplot(aes(x = MRdays, y = mm)) +
  geom_point() +
  geom_smooth(method = "lm") +
  facet_grid(RandomArm ~ ThickChange)
```



6.5 Exporatory ROI Analysis..

Running the RCT analysis ROI-wise with FDR correction.

```
library(ggseg)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
library(dplyr)
library(broom)
```

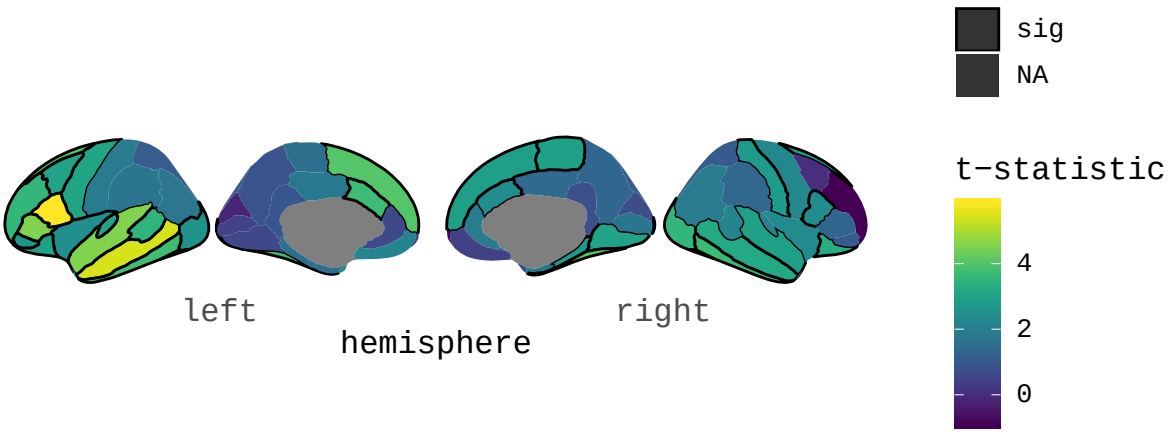
```r
RCT_ROIwise <- RCT_CT %>%
  gather(elabel, change_mm, ends_with('_thickavg_change')) %>%
  group_by(elabel) %>%
  do(tidy(lm(change_mm ~ RandomArm + sex + age + site, data= .))) %>%
  ungroup() %>% group_by(term) %>%
  mutate(p_FDR = p.adjust(p.value, method = 'fdr'))

RCT_ROIwise_supptable = RCT_ROIwise %>%
  filter(p_FDR < 0.06) %>%
  arrange(p.value) %>%
  mutate(ROI = str_replace(elabel, '_thickavg_change', '')) %>%
  ungroup() %>%
  select(ROI, estimate, std.error, statistic, p_FDR)

RCT_ROIwise_supptable %>% write_csv('../generated_csvs/supptable4_thickbyROI.csv')

RCT_ROIwise_supptable %>%
  knitr::kable(digits = 3)
```

ROI	estimate	std.error	statistic	p_FDR
L_parsopercularis	0.069	0.012	5.816	0.000
L_middletemporal	0.069	0.013	5.403	0.000
L_superiortemporal	0.059	0.013	4.554	0.001
L_parstriangularis	0.060	0.013	4.545	0.001
R_fusiform	0.070	0.017	4.178	0.003
L_superiorfrontal	0.054	0.013	4.037	0.003
L_inferiortemporal	0.070	0.019	3.780	0.005
L_fusiform	0.060	0.016	3.770	0.005
L_caudalanteriorcingulate	0.077	0.021	3.713	0.005
R_inferiortemporal	0.058	0.016	3.707	0.005
R_lateraloccipital	0.049	0.014	3.557	0.007
L_rostralmiddlefrontal	0.040	0.011	3.534	0.007
L_bankssts	0.056	0.016	3.483	0.007
R_lateralorbitofrontal	0.052	0.016	3.305	0.011
R_middletemporal	0.044	0.014	3.178	0.015
L_transversetemporal	0.096	0.031	3.076	0.017
L_precentral	0.065	0.021	3.072	0.017
R_lingual	0.052	0.017	3.025	0.018
L_caudalmiddlefrontal	0.044	0.015	3.005	0.018
R_paracentral	0.059	0.020	2.953	0.019
L_lateralorbitofrontal	0.057	0.019	2.947	0.019
R_superiortemporal	0.051	0.017	2.925	0.019
R_superiorfrontal	0.043	0.015	2.893	0.020
R_postcentral	0.030	0.011	2.771	0.026
R_parahippocampal	0.085	0.031	2.746	0.026
L_parsorbitalis	0.052	0.019	2.678	0.030
L_lateraloccipital	0.038	0.015	2.543	0.040
L_insula	0.058	0.024	2.433	0.049
R_parsopercularis	0.037	0.015	2.418	0.049
R_caudalanteriorcingulate	0.072	0.030	2.415	0.049
R_insula	0.051	0.022	2.335	0.057

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
RCT_ROIwise %>%
  ungroup() %>%
  filter(term == "RandomArmPlacebo") %>%
  mutate(label = str_replace(elabel, '_thickavg_change', '') %>%
           str_replace('L','lh') %>%
           str_replace('R','rh'),
         is_sig = if_else(p_FDR < 0.055, "sig", NA_character_)) %>%
  inner_join(atlas.info$data[2][[1]], by = "label") %>%
  ggseg(atlas="dkt", mapping=aes(fill=statistic, color = is_sig)) +
  scale_fill_viridis() +
  scale_color_manual(values = c("black", NULL)) +
  labs(color = NULL, fill = "t-statistic")
```



6.5.1 Figure Caption (brain plots)

Mapping the effect of olanzapine vs placebo on cortical thinning over 36 weeks in participant who remained clinically stable. The color scale represents the t-statistic for the effect of treatment (Placebo vs Olanzapine) where brighter colors represents greater cortical thinning. Areas outlined in black are those where the treatment effects was significant after correction for multiple comparisons (across 68 brain regions) using False Discovery Rate.

7 Surface Area Analysis

This script analyses hemisphere wide surface area

```
#load libraries
library(tidyverse)
library(broom)
library(lmerTest)
```

```
#bring in data
df <- read_csv('../generated_csvs/STOPPD_participantsCT_20181111.csv') #generated by 05_STOPPD_error in

RandomArmColors = c( "#FFC200", "#007aa3")
```

```
#make sure that STUDYID is an interger not a number
  df$STUDYID <- as.character(df$STUDYID)

#make sure that dateDiff is a number, not an interger
  df$dateDiff <- as.numeric(df$dateDiff)

# label the randomization variable
df$RandomArm <- factor(df$randomization,
                       levels = c("O", "P"),
                       labels = c("Olanzapine", "Placebo"))

#restructure data for RCT completers' only (N=40)
  RCT_SA <- df %>%
    filter(category == "RCT")
```

```
#write out clean dataframe
 # write.csv(RCT_CT, '../generated_data/df_leftCT.csv', row.names=FALSE)
```

7.1 RCT only

```
RCT_SA %>% count(RandomArm) %>% knitr::kable()
```

RandomArm	n
Olanzapine	26
Placebo	14

```
fit_all <- lmer(LSurfArea_change ~ RandomArm + sex + age + (1|site), data= RCT_SA)
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LSurfArea_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SA
##
## REML criterion at convergence: 564.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0838 -0.5189  0.1080  0.6790  2.0453
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  site     (Intercept) 5.515e-09 7.426e-05
##  Residual             2.363e+05 4.861e+02
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                   Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)         525.53     305.14  36.00   1.722  0.09360 .
## RandomArmPlacebo    477.83     163.94  36.00   2.915  0.00609 **
## sexM                -37.09     157.81  36.00  -0.235  0.81551
## age                 -15.79       5.66  36.00  -2.791  0.00836 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

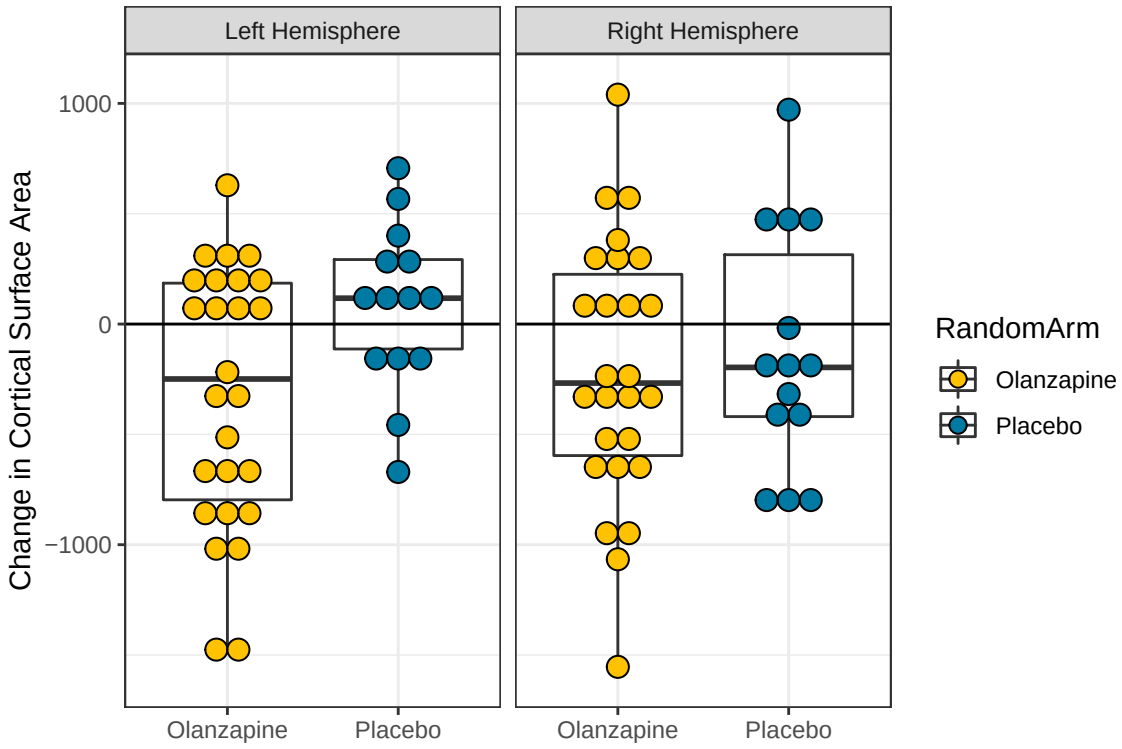
7.1.1 Right Surface Area Model

```
fit_all <- lmer(RSurfArea_change ~ RandomArm + sex + age + (1|site), data= RCT_SA)
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
```

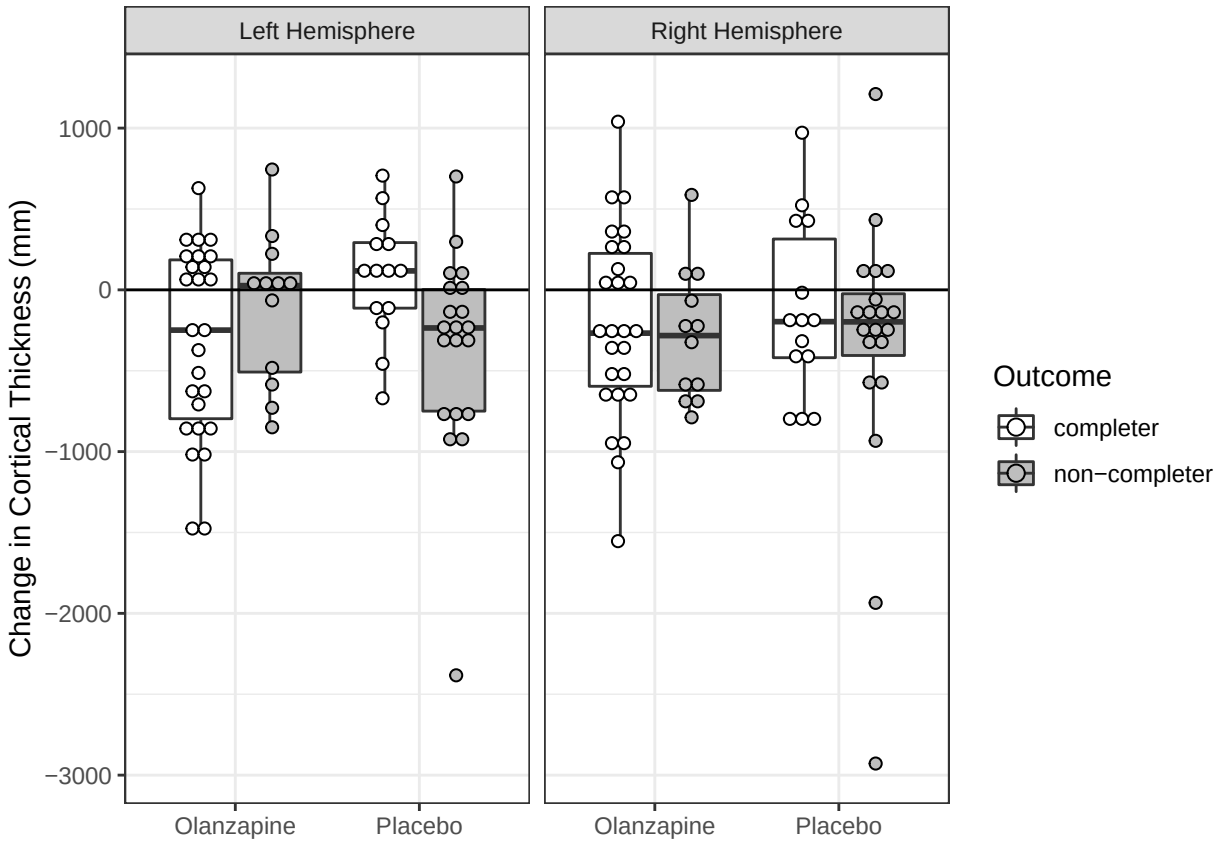
```
## lmerModLmerTest]
## Formula: RSurfArea_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SA
##
## REML criterion at convergence: 576.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0486 -0.7543 -0.1248  0.6839  2.5139
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  site     (Intercept)      0     0.0
##  Residual             325363   570.4
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                   Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)       205.920    358.036  36.000   0.575    0.569
## RandomArmPlacebo  143.174    192.356  36.000   0.744    0.462
## sexM              153.063    185.168  36.000   0.827    0.414
## age                -9.417      6.641  36.000  -1.418    0.165
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

```
#boxplot of difference in thickness (y axis) by randomization group (x axis)
RCT_SA %>%
  gather(TCT, mm, LSurfArea_change, RSurfArea_change) %>%
  mutate(ThickChange = factor(TCT, levels = c("LSurfArea_change", "RSurfArea_change"),
                              labels = c("Left Hemisphere", "Right Hemisphere"))) %>%
ggplot(aes(x= RandomArm, y = mm, fill = RandomArm)) +
     geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
     geom_dotplot(binaxis = 'y', stackdir = 'center', binwidth = 100) +
     geom_hline(yintercept = 0) +
     labs(x = NULL, y = "Change in Cortical Surface Area") +
     scale_fill_manual(values = RandomArmColors) +
     scale_shape_manual(values = c(21)) +
     facet_wrap(~ ThickChange) +
     theme_bw()
```



7.2 Plots and one sampled ttest in all participants

```
df %>%
  gather(TCT, mm, LSurfArea_change, RSurfArea_change) %>%
  mutate(SurfAreaChange = factor(TCT, levels = c("LSurfArea_change", "RSurfArea_change"),
                                 labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  ggplot(aes(x= RandomArm, y = mm, fill = Outcome)) +
     geom_boxplot(outlier.shape = NA) +
     geom_dotplot(binaxis = 'y', stackdir = 'center',
                  position=position_dodge(0.8), binwidth = 75) +
     geom_hline(yintercept = 0) +
     xlab(NULL) +
     ylab("Change in Cortical Thickness (mm)") +
     theme_bw() +
     scale_fill_manual(values = c('white','grey')) +
     facet_wrap(~SurfAreaChange)
```




```
df %>%
  gather(TCT, mm, LSurfArea_change, RSurfArea_change) %>%
  mutate(SurfAreaChange = factor(TCT, levels = c("LSurfArea_change", "RSurfArea_change"),
                                 labels = c("Left Hemisphere", "Right Hemisphere")),
         Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  group_by(SurfAreaChange, category) %>%
  do(tidy(t.test(.$mm, mu = 0, alternative = "two.sided"))) %>%
  knitr::kable()
```

SurfAreaChange	category	estimate	statistic	p.value	parameter	conf.low	conf.high	method
Left Hemisphere	Off protocol	-4.3400	-0.0194916	0.9853825	4	-622.5438	613.8637597	One Sam
Left Hemisphere	RCT	-176.8300	-2.0199766	0.0502934	39	-353.8976	0.2376122	One Sam
Left Hemisphere	Relapse	-310.8519	-2.7173890	0.0115513	26	-545.9912	-75.7124877	One Sam
Right Hemisphere	Off protocol	-206.6400	-0.8364687	0.4499511	4	-892.5289	479.2489135	One Sam
Right Hemisphere	RCT	-183.5925	-2.0476152	0.0473750	39	-364.9502	-2.2347509	One Sam
Right Hemisphere	Relapse	-343.8444	-2.4151973	0.0230569	26	-636.4840	-51.2048411	One Sam

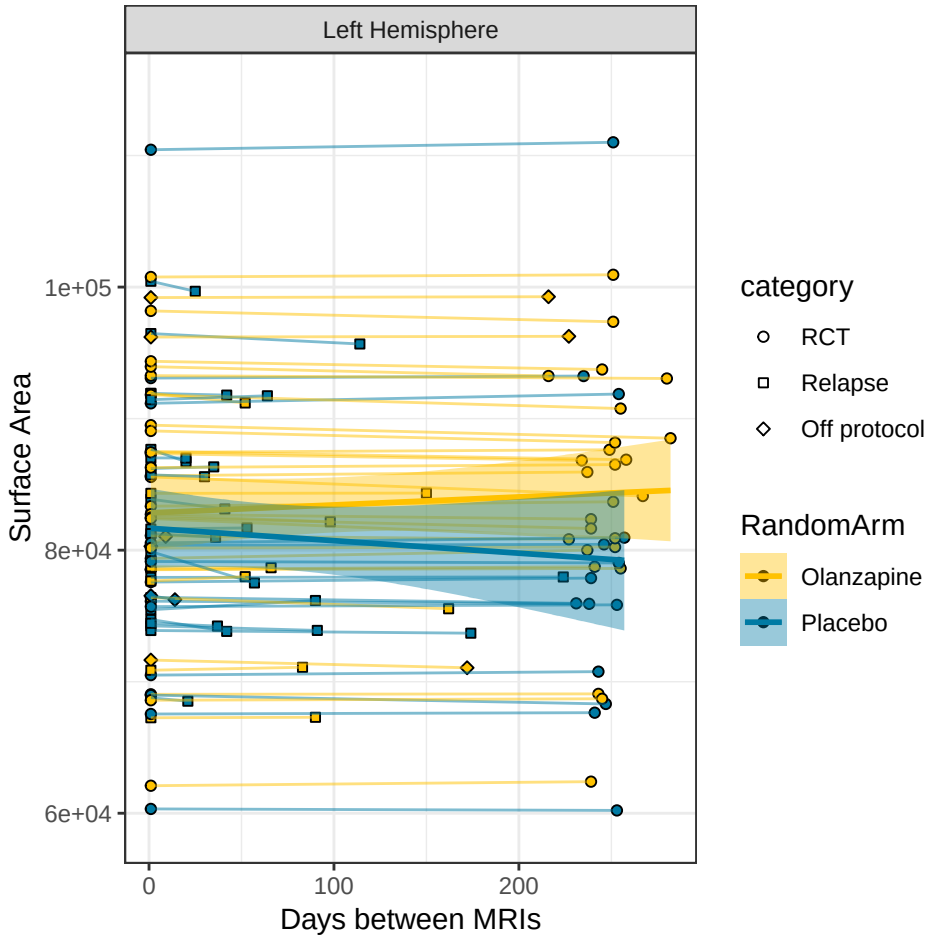
7.3 RCT & Relapse (with time as factor)

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_LSA <- df %>%
    gather(oldcolname, SurfArea, LSurfArea_01, LSurfArea_02) %>%
    mutate(model_days = if_else(oldcolname == "LSurfArea_01", 1, dateDiff)) %>%
mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")),
```

```
            hemi = "Left Hemisphere")

RCTRelapse_LSA %>% filter(model_days == 1) %>% count(RandomArm) %>% knitr::kable()
```

RandomArm	n
Olanzapine	38
Placebo	34

```
#plot all data, including outlier (participant 210030)
RCTRelapse_LSA %>%
  mutate(hemi = "Left Hemisphere") %>%
  ggplot(aes(x=model_days, y=SurfArea, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  xlab("Days between MRIs") +
  ylab("Surface Area") +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  scale_y_continuous(limits = c(59000,115000)) +
  theme_bw()  +
  facet_wrap(~hemi)
```

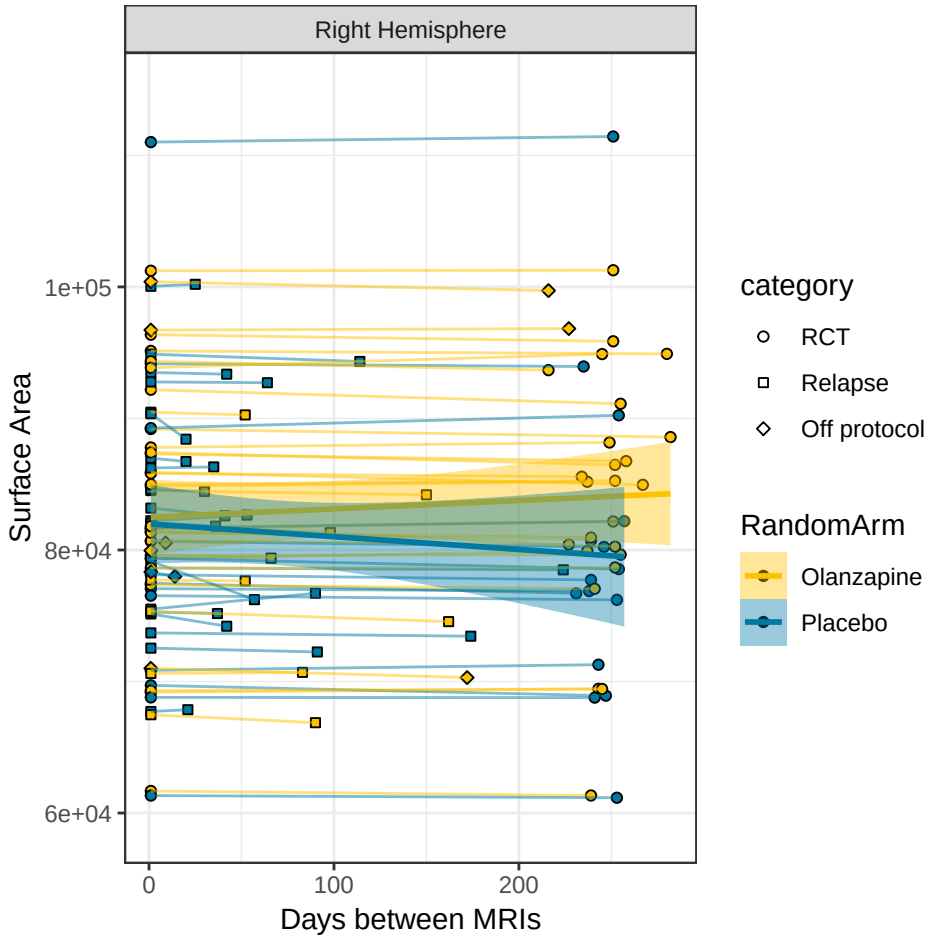


```r
#run mixed linear model, with covariates
  fit_all <- lmer(SurfArea ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: SurfArea ~ RandomArm * model_days + sex + age + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_LSA
##
## REML criterion at convergence: 2542.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.94832 -0.46341 -0.00847  0.45821  2.93101
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 52815526 7267.4
##  site     (Intercept) 12067533 3473.8
##  Residual               163892  404.8
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 85136.6299  3810.0342    29.2781  22.345
## RandomArmPlacebo            -2027.5553  1741.6216    65.5546  -1.164
## model_days                     -1.2901     0.4267    70.0217  -3.023
## sexM                        11090.3955  1745.4287    65.4625   6.354
## age                          -114.7242    56.7060    65.1570  -2.023
## RandomArmPlacebo:model_days     1.2395     0.7210    70.0556   1.719
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo             0.24857
## model_days                   0.00349 **
## sexM                        2.31e-08 ***
## age                          0.04717 *
## RandomArmPlacebo:model_days  0.09002 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.182
## model_days  -0.013  0.025
## sexM        -0.168  0.053  0.000
## age         -0.796 -0.064  0.002 -0.077
## RndmArmPl:_  0.007 -0.034 -0.592  0.001 -0.001
```

7.3.1 Running the right hemisphere RCTRelapse

```r
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_RSA <- df %>%
    gather(oldcolname, SurfArea, RSurfArea_01, RSurfArea_02) %>%
    mutate(model_days = if_else(oldcolname == "RSurfArea_01", 1, dateDiff)) %>%
    mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")),
           hemi = "Right Hemisphere")


#plot all data, including outlier (participant 210030)
RCTRelapse_RSA %>%
  ggplot(aes(x=model_days, y=SurfArea, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm", formula=y~poly(x,1)) +
  xlab("Days between MRIs") +
  ylab("Surface Area") +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  scale_y_continuous(limits = c(59000,115000)) +
  theme_bw()  +
  facet_wrap(~hemi)
```



```r
#run mixed linear model, with covariates
  fit_all <- lmer(SurfArea ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelaps
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: SurfArea ~ RandomArm * model_days + sex + age + (1 | site) +
##     (1 | STUDYID)
##    Data: RCTRelapse_RSA
##
## REML criterion at convergence: 2563.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.12738 -0.35173  0.00122  0.37935  3.09897
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 54382971 7374.5
##  site     (Intercept) 11216454 3349.1
##  Residual               214731  463.4
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 84685.7396  3826.6624    31.4280  22.130
## RandomArmPlacebo            -1348.4060  1767.5776    65.6422  -0.763
## model_days                     -1.0748     0.4884    70.0280  -2.201
## sexM                        11087.7051  1771.1232    65.5082   6.260
## age                          -112.9806    57.5465    65.1767  -1.963
## RandomArmPlacebo:model_days     0.3040     0.8253    70.0711   0.368
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo              0.4483
## model_days                    0.0311 *
## sexM                        3.35e-08 ***
## age                           0.0539 .
## RandomArmPlacebo:model_days   0.7137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.184
## model_days  -0.015  0.028
## sexM        -0.170  0.052  0.000
## age         -0.804 -0.064  0.002 -0.077
## RndmArmPl:_  0.008 -0.038 -0.592  0.001 -0.001
```

8 Whole Skeleton Fractional Anisotropy

```
#load libraries
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.2.0
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(broom)
library(lmerTest)
```

```
## Loading required package: lme4
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
## Loading required package: methods
##
## Attaching package: 'lmerTest'
## The following object is masked from 'package:lme4':
##
##     lmer
## The following object is masked from 'package:stats':
##
##     step
```

```
#bring in subject info (generated by 03_STOPPD_masterDF.Rmd)
# then take only the subjects who completed (n= 72 - note two were excluded for IF)
df <- read_csv('../generated_csvs/STOPPD_masterDF_2018-11-05.csv') %>%
  mutate(STUDYID = as.character(STUDYID)) %>%
  filter(second_complete == "Yes", MR_exclusion == "No")

#rename timepoint variable for clarity
colnames(df)[colnames(df)=="second_timepoint"] <- "category"

#make a datediff column for time between scans
df$dateDiff <- as.numeric(round(difftime(df$second_date, df$first_date, units = "days"), 0))

RandomArmColors = c( "#FFC200", "#007aa3")
```

8.1 Known exclusion reasons

8.1.0.1 known DWI issues

subject 410012 timepoint 02 -> scan was blacklisted "aborted" for system failure..no DWI for this participant

subject 220009_timepoint 01 -> scan was also incomplete (this participant was only able complete the T1w)

So we will filter the data table to exclude these 2 participants (final n=71)

```
df <- filter(df, !(STUDYID %in% c("410012", "220009")))
```

8.2 mangling the Mean Diffusivity cata data

Erin reran the enigma DTI pipeline for only PMC using a different skull stripping parameter (-fa 0.7 to BET). We will use these numbers instead of the others in the archive here..

```
#bring in FA data (from the filesystem)
FA_most <- read_csv('../data/enigma-DTI_archive_201811/enigmaDTI-FA-results.csv')
FA_PMC <- read_csv('../data/enigma-DTI_PMCredo_201809/enigmaDTI-FA-results.csv')

# separate id into it's parts and then drop old PMC data
FA_most <- FA_most %>%
  separate(id, into = c("study", "site", "STUDYID", "timepoint")) %>%
  filter(site != "PMC")

# separate the PMC subject id into it's parts and then bind to the data from the other sites
FA <- FA_PMC %>%
  separate(id, into = c("study", "site", "STUDYID", "timepoint")) %>%
  bind_rows(FA_most)

# drop acute ("00") and other ("03") timepoints from the analysis
FA <- FA %>%
  filter(!(timepoint %in% c("00", "03"))) %>%
  gather(tract, FA, ends_with("FA")) %>%
  spread(timepoint, FA) %>%
  mutate(change = `02` - `01`) %>%
  gather(timepoint, FA, `01`, `02`, change) %>%
  unite(tract_timepoint, tract, timepoint) %>%
  spread(tract_timepoint, FA)

rm(FA_most, FA_PMC)
```

8.3 check for missing FA data

```
# filter the master spreadsheet for the list of completers (no output means we are ok)
df %>%
  anti_join(FA, by = "STUDYID") %>%
  summarise(`Number of missing FA values` = n()) %>%
  knitr::kable()
```

Number of missing FA values
0

8.4 merge (i.e. join) the FA data with the clinical scores

```
all_FA <- df %>%
  select(STUDYID, sex, age, randomization, category, dateDiff) %>%
  mutate(RandomArm = factor(randomization,
                        levels = c("O", "P"),
                        labels = c("Olanzapine", "Placebo"))) %>%
  left_join(FA, by = "STUDYID")

all_FA %>%
  filter(is.na(AverageFA_FA_01)) %>%
  summarise(`Number of missing timepoint 1 FA values` = n()) %>%
  knitr::kable()
```

Number of missing timepoint 1 FA values
0

```
all_FA %>%
  filter(is.na(AverageFA_FA_02)) %>%
  summarise(`Number of missing timepoint 2 FA values` = n()) %>%
  knitr::kable()
```

Number of missing timepoint 2 FA values
0

```
#write out clean FA speadsheet (required for subsequent FA analyses)
write.csv(all_FA, '../generated_csvs/STOPPD_FAclean.csv', row.names = FALSE)
```

8.5 Running Table One to get baseline values

```
library(tableone)
CreateTableOne(data = all_FA,
               strata = c("RandomArm"),
               vars = c("category", "AverageFA_FA_01"))
```

```
##                               Stratified by RandomArm
##                                Olanzapine   Placebo      p      test
##   n                              37           34
##   category (%)                                                 0.004
##      Off protocol                 4 (10.8)     1 ( 2.9)
##      RCT                         26 (70.3)    14 (41.2)
##      Relapse                      7 (18.9)    19 (55.9)
##   AverageFA_FA_01 (mean (sd)) 0.39 (0.03)  0.38 (0.03)   0.130
```

8.5.0.1 baseline collapsed value

```
CreateTableOne(data = all_FA,
               vars = c("category", "AverageFA_FA_01"))
```

```
##
```

Table 7: t.test for baseline group differences

brain	t	df	p.value	method
AverageFA_FA_01	1.54	68.8	0.13	Welch Two Sample t-test

```
##                                Overall
##   n                              71
##   category (%)
##      Off protocol                 5 ( 7.0)
##      RCT                         40 (56.3)
##      Relapse                     26 (36.6)
##   AverageFA_FA_01 (mean (sd)) 0.38 (0.03)
```

8.5.0.2 baseline t.test

```
all_FA %>%
  select(RandomArm, AverageFA_FA_01) %>%
  gather(brain, mm, -RandomArm) %>%
  group_by(brain) %>%
  do(tidy(t.test(mm~RandomArm, data = .))) %>%
  select(brain, statistic, parameter, p.value, method) %>%
  rename(t = statistic, df = parameter) %>%
  knitr::kable(caption = "t.test for baseline group differences", digits = 2)
```

Note that there are very strong scanner effects so it is probably better to consider these by site..

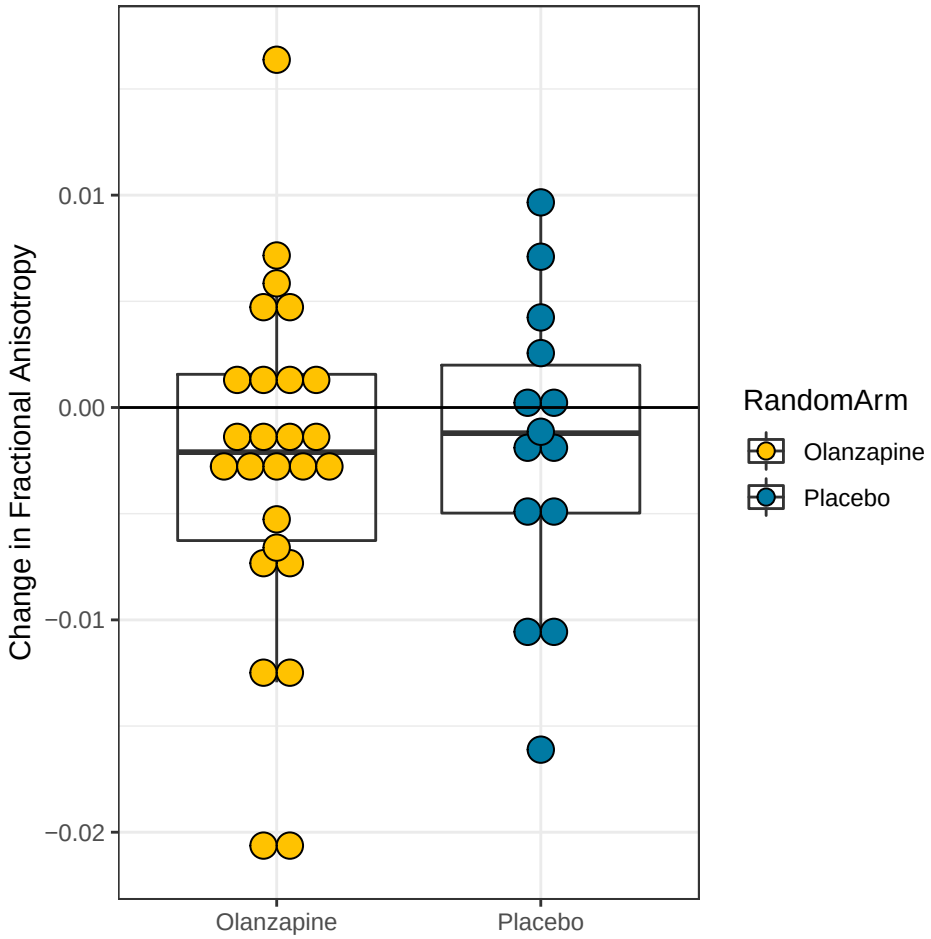
```
CreateTableOne(data = all_FA,
               strata = c("RandomArm","site"),
               vars = c("category", "AverageFA_FA_01"))
```

```
##                               Stratified by RandomArm:site
##                                Olanzapine:CMH Placebo:CMH  Olanzapine:MAS
##   n                              18             11            7
##   category (%)
##      Off protocol                 3 (16.7)       1 ( 9.1)     0 ( 0.0)
##      RCT                         12 (66.7)       5 (45.5)     4 (57.1)
##      Relapse                      3 (16.7)       5 (45.5)     3 (42.9)
##   AverageFA_FA_01 (mean (sd)) 0.42 (0.02)    0.41 (0.02)  0.37 (0.01)
##                               Stratified by RandomArm:site
##                                Placebo:MAS  Olanzapine:NKI Placebo:NKI
##   n                              10            6              7
##   category (%)
##      Off protocol                 0 ( 0.0)      0 (  0.0)      0 ( 0.0)
##      RCT                          4 (40.0)      6 (100.0)      3 (42.9)
##      Relapse                      6 (60.0)      0 (  0.0)      4 (57.1)
##   AverageFA_FA_01 (mean (sd)) 0.37 (0.02)  0.36 (0.03)    0.35 (0.02)
##                               Stratified by RandomArm:site
##                                Olanzapine:PMC Placebo:PMC  p      test
##   n                              6              6
##   category (%)                                              0.180
##      Off protocol                 1 (16.7)       0 ( 0.0)
##      RCT                          4 (66.7)       2 (33.3)
##      Relapse                      1 (16.7)       4 (66.7)
##   AverageFA_FA_01 (mean (sd)) 0.35 (0.01)    0.36 (0.02)  <0.001
```

8.6 RCT only

```
#boxplot of difference in FA in whole skeleton (y axis) by randomization group (x axis)
ggplot(RCT_FA, aes(x= RandomArm, y = diffAverageSkel_FA, fill = RandomArm)) +
   geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
   geom_dotplot(binaxis = 'y', stackdir = 'center') +
   geom_hline(yintercept = 0) +
   xlab(NULL) +
   ylab("Change in Fractional Anisotropy") +
   scale_fill_manual(values = RandomArmColors) +
   scale_shape_manual(values = c(21)) +
   theme_bw()
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#run linear model with covariates of sex, age and site
fit_rct <- lmer(diffAverageSkel_FA ~ RandomArm + sex + age + (1|site), data= RCT_FA)
summary(fit_rct)
```

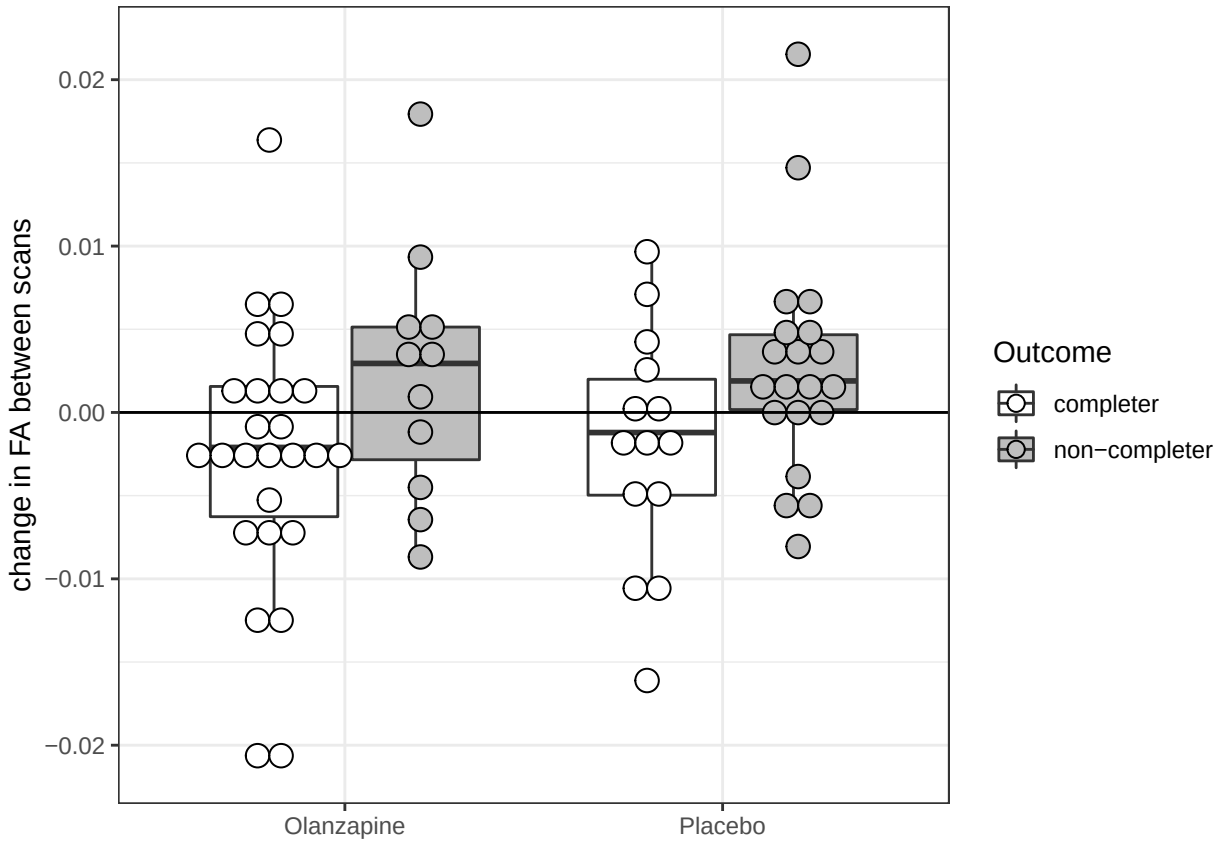
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: diffAverageSkel_FA ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_FA
##
```

```
## REML criterion at convergence: -234.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.41668 -0.51246  0.05053  0.61252  2.31072
##
## Random effects:
##  Groups   Name        Variance                        Std.Dev.
##  site     (Intercept) 0.00000000000000000000001416 0.00000000000119
##  Residual             0.00005358314841593728714876 1 0.00732005112113
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                     Estimate  Std. Error          df t value Pr(>|t|)
## (Intercept)       0.00666412  0.00459470 36.00000000   1.450   0.1556
## RandomArmPlacebo  0.00177238  0.00246851 36.00000000   0.718   0.4774
## sexM              0.00304198  0.00237628 36.00000000   1.280   0.2087
## age              -0.00020489  0.00008522 36.00000000  -2.404   0.0215 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

8.7 adding the non-RCT people to the plot

```
all_FA %>%
  mutate(Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  ggplot(aes(x= RandomArm, y = diffAverageSkel_FA, fill = Outcome)) +
     geom_boxplot(outlier.shape = NA) +
     geom_dotplot(binaxis = 'y', stackdir = 'center',
                  position=position_dodge(0.8)) +
     geom_hline(yintercept = 0) +
     xlab(NULL) +
     ylab("change in FA between scans") +
     theme_bw() +
     scale_fill_manual(values = c('white','grey'))
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



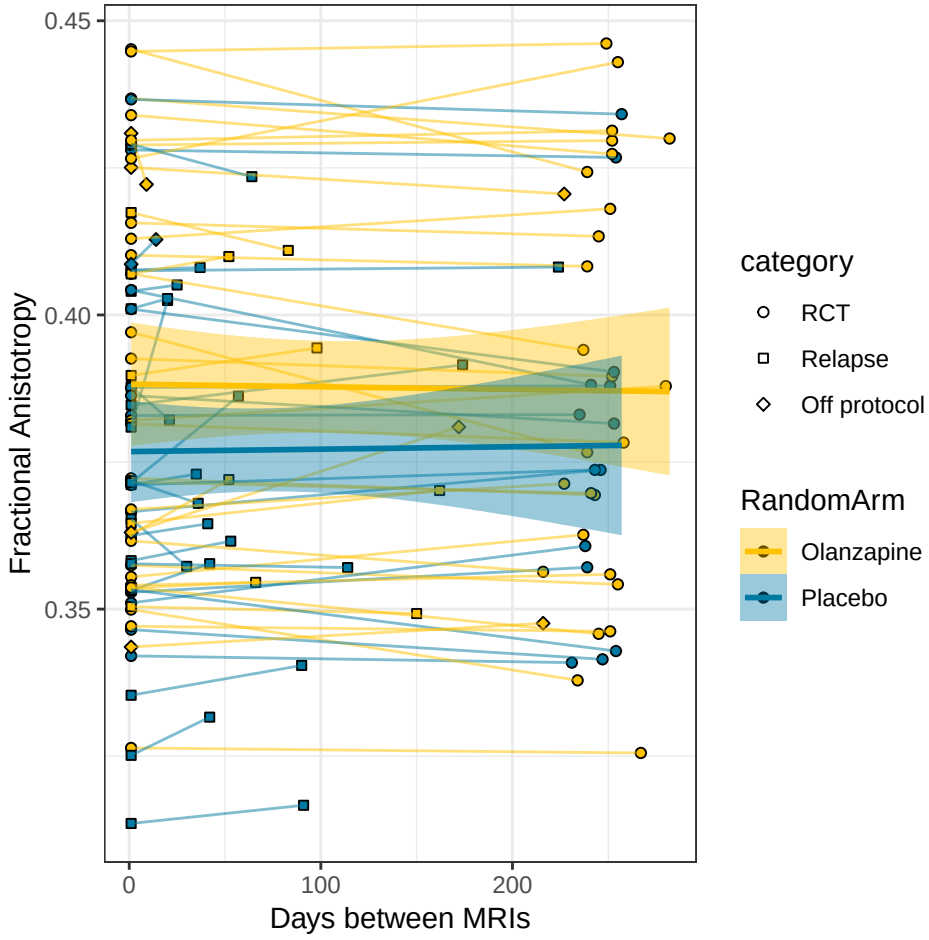

```
all_FA %>%
  mutate(Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  filter(category != "Off protocol") %>%
  group_by(RandomArm, category) %>%
  do(tidy(t.test(.$diffAverageSkel_FA, mu = 0, alternative = "two.sided"))) %>%
  knitr::kable(digits = 3)
```

RandomArm	category	estimate	statistic	p.value	parameter	conf.low	conf.high	method	alternat
Olanzapine	RCT	-0.003	-1.667	0.108	25	-0.006	0.001	One Sample t-test	two.side
Olanzapine	Relapse	0.002	1.174	0.285	6	-0.002	0.007	One Sample t-test	two.side
Placebo	RCT	-0.002	-1.055	0.311	13	-0.006	0.002	One Sample t-test	two.side
Placebo	Relapse	0.003	1.642	0.118	18	-0.001	0.006	One Sample t-test	two.side

8.8 RCT & Relapse (with time as factor)

```
RCTRelapse_wholeskelFA <- RCTRelapse_FA %>%
  filter(Tract == "AverageFA") %>%
  mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")))
#plot
RCTRelapse_wholeskelFA %>%
  ggplot(aes(x=model_days, y=FA, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm", formula=y~poly(x,1)) +
```

```
  xlab("Days between MRIs") +
  ylab("Fractional Anistotropy") +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  theme_bw()
```




```
RCTRelapse_wholeskelFA <- RCTRelapse_FA %>%
  filter(Tract == "AverageFA")
```

```
#run mixed linear model, with covariates
fit_all <- lmer(FA ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_wholesk
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## FA ~ RandomArm * model_days + sex + age + (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_wholeskelFA
##
## REML criterion at convergence: -775.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.21982 -0.35896  0.00547  0.39988  1.97798
##
## Random effects:
##  Groups   Name        Variance   Std.Dev.
##  STUDYID  (Intercept) 0.00026598 0.016309
##  site     (Intercept) 0.00070535 0.026558
##  Residual             0.00002933 0.005416
## Number of obs: 142, groups:  STUDYID, 71; site, 4
##
## Fixed effects:
##                                Estimate   Std. Error           df t value
## (Intercept)                 0.407631799  0.015434155  5.236419521  26.411
## RandomArmPlacebo           -0.001841260  0.004141270 69.083333079  -0.445
## model_days                 -0.000006488  0.000005707 69.766731348  -1.137
## sexM                        0.007365517  0.004061284 64.109601146   1.814
## age                        -0.000643375  0.000131280 64.097168345  -4.901
## RandomArmPlacebo:model_days  0.000002427  0.000009590 70.910670616   0.253
##                                Pr(>|t|)
## (Intercept)                0.000000895 ***
## RandomArmPlacebo                0.6580
## model_days                      0.2594
## sexM                            0.0744 .
## age                        0.000006817 ***
## RandomArmPlacebo:model_days     0.8009
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.109
## model_days  -0.044  0.144
## sexM        -0.099  0.071  0.004
## age         -0.451 -0.076  0.008 -0.086
## RndmArmPl:_  0.024 -0.190 -0.595  0.002 -0.002
```

```
#run mixed linear model, with covariates
RCTRelapse_wholeskelFA_sense <- RCTRelapse_wholeskelFA %>% filter(category != "Off protocol")

fit_all <- lmer(FA ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_wholesk
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## FA ~ RandomArm * model_days + sex + age + (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_wholeskelFA_sense
##
## REML criterion at convergence: -716.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.25703 -0.35536  0.00513  0.38574  1.96192
##
## Random effects:
##  Groups   Name        Variance   Std.Dev.
```

```
##  STUDYID  (Intercept) 0.00027717 0.016648
##  site     (Intercept) 0.00069513 0.026365
##  Residual             0.00002769 0.005262
## Number of obs: 132, groups:  STUDYID, 66; site, 4
##
## Fixed effects:
##                                Estimate   Std. Error           df t value
## (Intercept)                 0.407253648  0.015507242  5.452825329  26.262
## RandomArmPlacebo           -0.003422358  0.004344248 63.413298187  -0.788
## model_days                 -0.000008731  0.000005756 64.532321497  -1.517
## sexM                        0.007097355  0.004309842 59.104600046   1.647
## age                        -0.000612219  0.000139000 59.089659708  -4.404
## RandomArmPlacebo:model_days  0.000004781  0.000009453 65.474883036   0.506
##                                Pr(>|t|)
## (Intercept)                0.000000595 ***
## RandomArmPlacebo                0.434
## model_days                      0.134
## sexM                            0.105
## age                        0.000045244 ***
## RandomArmPlacebo:model_days     0.615
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.114
## model_days  -0.045  0.142
## sexM        -0.071  0.029 -0.001
## age         -0.466 -0.071  0.011 -0.134
## RndmArmPl:_  0.028 -0.185 -0.609  0.007 -0.007
```

8.9 running exploratory Tractwise analysis

No significant effects found

```
RCT_Tractwise <- RCT_FA %>%
  gather(elabel, change_FA, ends_with('_FA_change')) %>%
  filter(!str_detect(elabel, '-L'),
         !str_detect(elabel, '-R'),
         !str_detect(elabel, 'Average')) %>%
  group_by(elabel) %>%
  do(tidy(lm(change_FA ~ RandomArm + sex + age + site, data= .))) %>%
  ungroup() %>% group_by(term) %>%
  mutate(p_FDR = p.adjust(p.value, method = 'fdr'))

RCT_Tractwise %>%
  filter(p_FDR < 0.1) %>%
  arrange(p.value) %>%
  knitr::kable()
```

elabel	term	estimate	std.error	statistic	p.value	p_FDR

```
#cleanup
rm('df', 'fit_all', 'fit_rct', 'FA', 'plot', 'RCT_FA', 'RCTRelapse_FA')
```

```
## Warning in rm("df", "fit_all", "fit_rct", "FA", "plot", "RCT_FA",
## "RCTRelapse_FA"): object 'plot' not found
```

9 Whole Skeleton Mean Diffusivity

```
#load libraries
library(tidyverse)
library(lme4)
library(lmerTest)
library(growthmodels)
library(broom)
```

```
#bring in subject info (generated by 03_STOPPD_masterDF.Rmd)
# then take only the subjects who completed (n= 72 - note two were excluded for IF)
df <- read_csv('../generated_csvs/STOPPD_masterDF_2018-11-05.csv') %>%
  mutate(STUDYID = as.character(STUDYID)) %>%
  filter(second_complete == "Yes", MR_exclusion == "No")

#rename timepoint variable for clarity
colnames(df)[colnames(df)=="second_timepoint"] <- "category"

#make a datediff column for time between scans
df$dateDiff <- as.numeric(round(difftime(df$second_date, df$first_date, units = "days"), 0))

RandomArmColors = c( "#FFC200", "#007aa3")
```

9.1 Known exclusion reasons

9.1.0.1 known DWI issues

subject 410012 timepoint 02 -> scan was blacklisted "aborted" for system failure..no DWI for this participant

subject 220009_timepoint 01 -> scan was also incomplete (this participant was only able complete the T1w)

So we will filter the data table to exclude these 2 participants (final n=71)

```
df <- filter(df, !(STUDYID %in% c("410012", "220009")))
```

9.2 mangling the Mean Diffusivity cata data

Erin reran the enigma DTI pipeline for only PMC using a different skull stripping parameter (-fa 0.7 to BET). We will use these numbers instead of the others in the archive here..

```
#bring in MD data (from the filesystem)
MD_most <- read_csv('../data/enigma-DTI_archive_201811/enigmaDTI-MD-results.csv')
MD_PMC <- read_csv('../data/enigma-DTI_PMCredo_201809/enigmaDTI-MD-results.csv')

# separate id into it's parts and then drop old PMC data
MD_most <- MD_most %>%
  separate(id, into = c("study", "site", "STUDYID", "timepoint")) %>%
```

```
  filter(site != "PMC")

# separate the PMC subject id into it's parts and then bind to the data from the other sites
MD <- MD_PMC %>%
  separate(id, into = c("study", "site", "STUDYID", "timepoint")) %>%
  bind_rows(MD_most)

# drop acute ("00") and other ("03") timepoints from the analysis
MD <- MD %>%
  filter(!(timepoint %in% c("00", "03"))) %>%
  gather(tract, MD, ends_with("MD")) %>%
  spread(timepoint, MD) %>%
  mutate(change = `02` - `01`) %>%
  gather(timepoint, MD, `01`, `02`, change) %>%
  unite(tract_timepoint, tract, timepoint) %>%
  spread(tract_timepoint, MD)
```

9.3 check for missing MD data

```
# filter the master spreadsheet for the list of completers (no output means we are ok)
df %>%
  anti_join(MD, by = "STUDYID") %>%
  summarise(`Number of missing MD values` = n()) %>%
  knitr::kable()
```

Number of missing MD values
0

9.4 merge (i.e. join) the MD data with the clinical scores

```
all_MD <- df %>%
  select(STUDYID, sex, age, randomization, category, dateDiff) %>%
  mutate(RandomArm = factor(randomization,
                        levels = c("O", "P"),
                        labels = c("Olanzapine", "Placebo"))) %>%
  left_join(MD, by = "STUDYID")

all_MD %>%
  filter(is.na(AverageFA_MD_01)) %>%
  summarise(`Number of missing timepoint 1 MD values` = n()) %>%
  knitr::kable()
```

Number of missing timepoint 1 MD values
0

```
all_MD %>%
  filter(is.na(AverageFA_MD_02)) %>%
  summarise(`Number of missing timepoint 2 MD values` = n()) %>%
  knitr::kable()
```

Number of missing timepoint 2 MD values
0

```
#write out clean MD speadsheet (required for subsequent MD analyses)
write.csv(all_MD, '../generated_csvs/STOPPD_MDclean.csv', row.names = FALSE)
```

9.5 Running Table One to get baseline values

```
library(tableone)
print(CreateTableOne(data = all_MD,
               strata = c("RandomArm"),
               vars = c("category", "AverageFA_MD_01")), contDigits = 5)
```

```
##                               Stratified by RandomArm
##                                Olanzapine        Placebo           p
##   n                                 37                34
##   category (%)                                                    0.004
##      Off protocol                    4 (10.8)          1 ( 2.9)
##      RCT                            26 (70.3)         14 (41.2)
##      Relapse                         7 (18.9)         19 (55.9)
##   AverageFA_MD_01 (mean (sd)) 0.00142 (0.00017) 0.00136 (0.00019)  0.165
##                               Stratified by RandomArm
##                                test
##   n
##   category (%)
##      Off protocol
##      RCT
##      Relapse
##   AverageFA_MD_01 (mean (sd))
```

```
print(CreateTableOne(data = all_MD,
               vars = c("category", "AverageFA_MD_01")), contDigits = 5)
```

```
##
##                               Overall
##   n                                71
##   category (%)
##      Off protocol                  5 ( 7.0)
##      RCT                          40 (56.3)
##      Relapse                      26 (36.6)
##   AverageFA_MD_01 (mean (sd)) 0.00140 (0.00018)
```

9.5.0.1 baseline t.test

```
all_MD %>%
  select(RandomArm, AverageFA_MD_01) %>%
  gather(brain, mm, -RandomArm) %>%
  group_by(brain) %>%
  do(tidy(t.test(mm~RandomArm, data = .))) %>%
  select(brain, statistic, parameter, p.value, method) %>%
  rename(t = statistic, df = parameter) %>%
  knitr::kable(caption = "t.test for baseline group differences", digits = 2)
```

Note that there are very strong scanner effects so it is probably better to consider these by site..

Table 8: t.test for baseline group differences

brain	t	df	p.value	method
AverageFA_MD_01	1.4	67.41	0.17	Welch Two Sample t-test

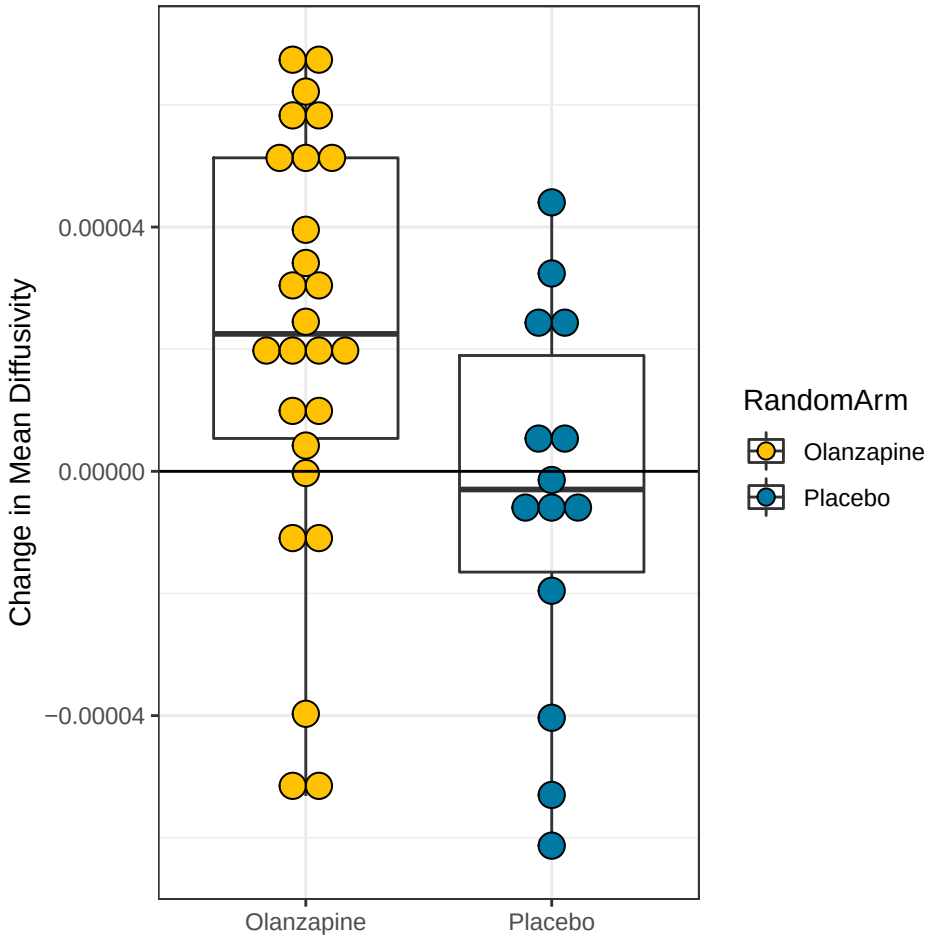
```
print(CreateTableOne(data = all_MD,
               strata = c("RandomArm","site"),
               vars = c("category", "AverageFA_MD_01")), contDigits = 5)
```

```
##                               Stratified by RandomArm:site
##                                Olanzapine:CMH    Placebo:CMH
##   n                                 18                11
##   category (%)
##      Off protocol                    3 (16.7)          1 ( 9.1)
##      RCT                            12 (66.7)          5 (45.5)
##      Relapse                         3 (16.7)          5 (45.5)
##   AverageFA_MD_01 (mean (sd)) 0.00151 (0.00018) 0.00151 (0.00013)
##                               Stratified by RandomArm:site
##                                Olanzapine:MAS    Placebo:MAS
##   n                                  7                10
##   category (%)
##      Off protocol                    0 ( 0.0)          0 ( 0.0)
##      RCT                             4 (57.1)          4 (40.0)
##      Relapse                         3 (42.9)          6 (60.0)
##   AverageFA_MD_01 (mean (sd)) 0.00131 (0.00012) 0.00126 (0.00017)
##                               Stratified by RandomArm:site
##                                Olanzapine:NKI    Placebo:NKI
##   n                                  6                 7
##   category (%)
##      Off protocol                    0 (  0.0)         0 ( 0.0)
##      RCT                             6 (100.0)         3 (42.9)
##      Relapse                         0 (  0.0)         4 (57.1)
##   AverageFA_MD_01 (mean (sd)) 0.00128 (0.00011) 0.00137 (0.00017)
##                               Stratified by RandomArm:site
##                                Olanzapine:PMC    Placebo:PMC       p
##   n                                  6                 6
##   category (%)                                                    0.180
##      Off protocol                    1 (16.7)          0 ( 0.0)
##      RCT                             4 (66.7)          2 (33.3)
##      Relapse                         1 (16.7)          4 (66.7)
##   AverageFA_MD_01 (mean (sd)) 0.00143 (0.00012) 0.00125 (0.00015) <0.001
##                               Stratified by RandomArm:site
##                                test
##   n
##   category (%)
##      Off protocol
##      RCT
##      Relapse
##   AverageFA_MD_01 (mean (sd))
```

9.6 RCT only

```
#boxplot of difference in MD in whole skeleton (y axis) by randomization group (x axis)
ggplot(RCT_MD, aes(x= RandomArm, y = diffAverageSkel_MD, fill = RandomArm)) +
   geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
   geom_dotplot(binaxis = 'y', stackdir = 'center') +
   geom_hline(yintercept = 0) +
   xlab(NULL) +
   ylab("Change in Mean Diffusivity") +
   scale_fill_manual(values = RandomArmColors) +
   scale_shape_manual(values = c(21)) +
   theme_bw()
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



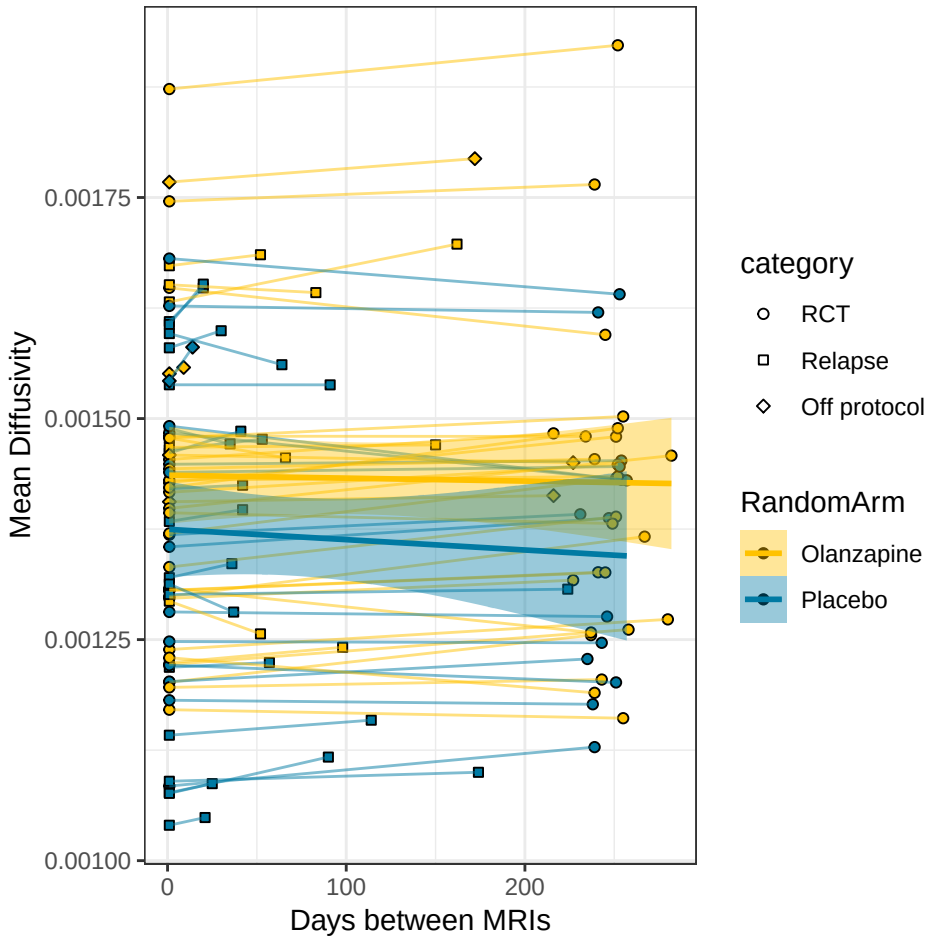

```
#run linear model with covariates of sex, age and site
fit_rct <- lmer(diffAverageSkel_MD ~ RandomArm + sex + age + (1|site), data= RCT_MD)
summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: diffAverageSkel_MD ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_MD
##
```

```
## REML criterion at convergence: -621.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3065 -0.4491  0.0422  0.7902  1.4955
##
## Random effects:
##  Groups   Name        Variance                          Std.Dev.
##  site     (Intercept) 0.000000000000000000000001198 0.000000000001094
##  Residual             0.000000001167888941144929037 0.000034174390136
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                        Estimate     Std. Error             df t value
## (Intercept)       0.00001593673  0.00002145081 35.99999999947   0.743
## RandomArmPlacebo -0.00002622481  0.00001152451 35.99999999950  -2.276
## sexM              0.00000257258  0.00001109389 35.99999999972   0.232
## age               0.00000008944  0.00000039787 35.99999999925   0.225
##                  Pr(>|t|)
## (Intercept)        0.4623
## RandomArmPlacebo   0.0289 *
## sexM               0.8179
## age                0.8234
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

9.7 RCT & Relapse (with time as factor)

```
RCTRelapse_wholeskelMD <- RCTRelapse_MD %>%
  filter(Tract == "AverageFA") %>%
  mutate(category = factor(category, levels = c("RCT","Relapse", "Off protocol")))
#plot
RCTRelapse_wholeskelMD %>%
  ggplot(aes(x=model_days, y=MD, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm", formula=y~poly(x,1)) +
  xlab("Days between MRIs") +
  ylab("Mean Diffusivity") +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  theme_bw()
```




```
#run mixed linear model, with covariates
fit_all <- lmer(MD ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_wholesk
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## MD ~ RandomArm * model_days + sex + age + (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_wholeskelMD
##
## REML criterion at convergence: -2246.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.75931 -0.44531 -0.00936  0.39581  1.91032
##
## Random effects:
##  Groups   Name        Variance        Std.Dev.
##  STUDYID  (Intercept) 0.0000000077174 0.00008785
##  site     (Intercept) 0.0000000087245 0.00009340
##  Residual             0.0000000004336 0.00002082
## Number of obs: 142, groups:  STUDYID, 71; site, 4
##
## Fixed effects:
```

```
##                                 Estimate     Std. Error             df
## (Intercept)                  0.00093603329  0.00006263043  8.68938506644
## RandomArmPlacebo            -0.00003682637  0.00002181291 66.85102591970
## model_days                   0.00000008623  0.00000002197 69.34886239368
## sexM                         0.00005832529  0.00002158631 64.19751680155
## age                          0.00000762514  0.00000069783 64.16140897910
## RandomArmPlacebo:model_days -0.00000009581  0.00000003699 69.94946322992
##                             t value             Pr(>|t|)
## (Intercept)                  14.945 0.000000169265448769 ***
## RandomArmPlacebo             -1.688             0.096017 .
## model_days                    3.925             0.000202 ***
## sexM                          2.702             0.008808 **
## age                          10.927 0.000000000000000276 ***
## RandomArmPlacebo:model_days  -2.590             0.011664 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.140
## model_days  -0.041  0.106
## sexM        -0.129  0.071  0.003
## age         -0.590 -0.076  0.006 -0.086
## RndmArmPl:_  0.023 -0.139 -0.594  0.001 -0.002
```

```
RCTRelapse_wholeskelMD_sense <- RCTRelapse_wholeskelMD %>% filter(category != "Off protocol")
#run mixed linear model, with covariates
fit_all <- lmer(MD ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_wholesk
summary(fit_all)
```

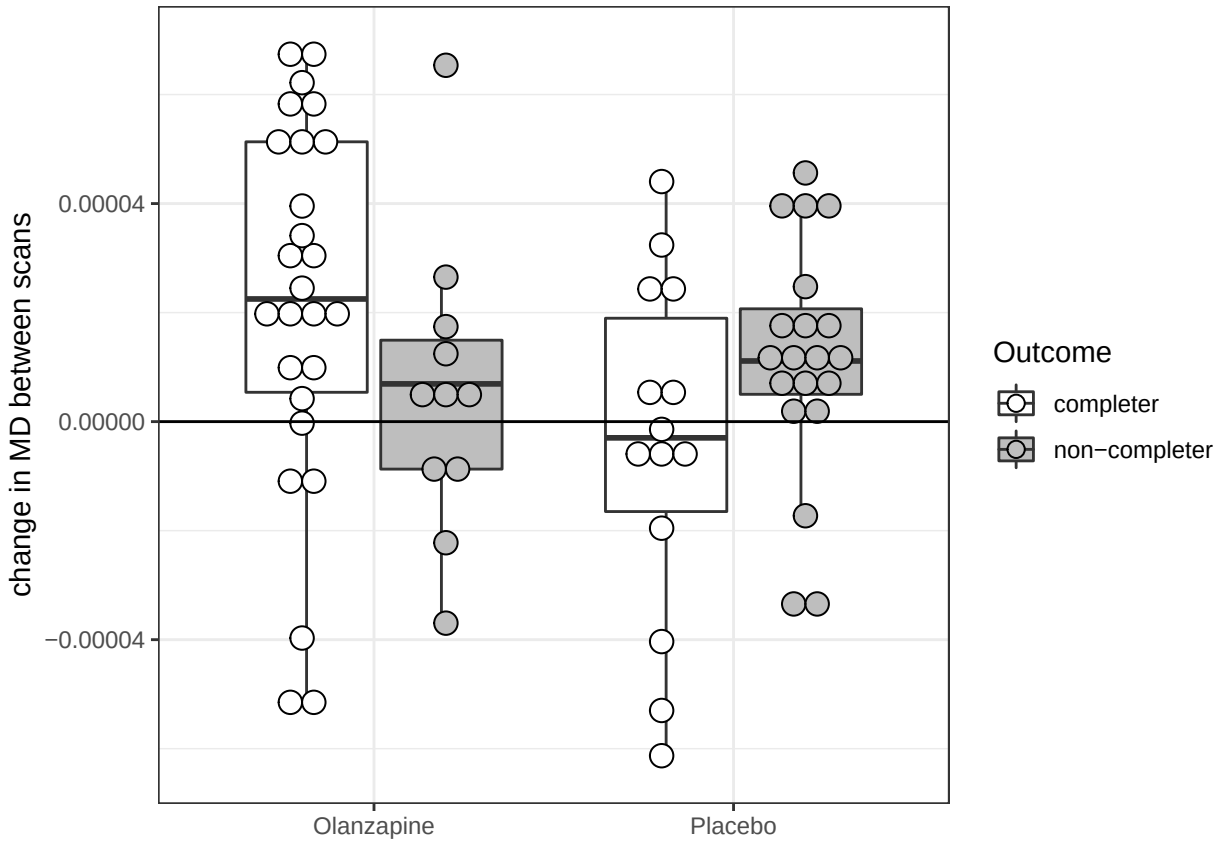
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## MD ~ RandomArm * model_days + sex + age + (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_wholeskelMD_sense
##
## REML criterion at convergence: -2074.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.75438 -0.43253 -0.00483  0.39766  1.90966
##
## Random effects:
##  Groups   Name        Variance       Std.Dev.
##  STUDYID  (Intercept) 0.000000007944 0.00008913
##  site     (Intercept) 0.000000008402 0.00009166
##  Residual             0.000000000447 0.00002114
## Number of obs: 132, groups:  STUDYID, 66; site, 4
##
## Fixed effects:
##                                 Estimate     Std. Error             df
## (Intercept)                  0.00093650663  0.00006296200  9.36164315607
## RandomArmPlacebo            -0.00003402152  0.00002283248 61.70059785288
## model_days                   0.00000009109  0.00000002314 64.26215950200
## sexM                         0.00006069042  0.00002282329 59.19528186801
```

```
## age                           0.00000756727  0.00000073617 59.14985441694
## RandomArmPlacebo:model_days -0.00000010128  0.00000003807 64.80607378513
##                              t value            Pr(>|t|)
## (Intercept)                   14.874 0.00000007918918823 ***
## RandomArmPlacebo              -1.490            0.141306
## model_days                     3.936            0.000206 ***
## sexM                           2.659            0.010062 *
## age                           10.279 0.00000000000000881 ***
## RandomArmPlacebo:model_days   -2.661            0.009824 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.147
## model_days  -0.045  0.108
## sexM        -0.092  0.029  0.000
## age         -0.608 -0.072  0.008 -0.134
## RndmArmPl:_  0.027 -0.142 -0.608  0.005 -0.006
```

9.8 adding the non-RCT people to the boxplot

```
all_MD %>%
  mutate(Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  ggplot(aes(x= RandomArm, y = diffAverageSkel_MD, fill = Outcome)) +
     geom_boxplot(outlier.shape = NA) +
     geom_dotplot(binaxis = 'y', stackdir = 'center',
                  position=position_dodge(0.8)) +
     geom_hline(yintercept = 0) +
     xlab(NULL) +
     ylab("change in MD between scans") +
     theme_bw() +
     scale_fill_manual(values = c('white','grey'))
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



9.8.1 post-hoc look at subgroups against 0 change null

```
all_MD %>%
  mutate(Outcome = case_when(category == "Off protocol" ~ "non-completer",
                             category == "Relapse" ~ "non-completer",
                             category == "RCT"~ "completer")) %>%
  filter(category != "Off protocol") %>%
  group_by(RandomArm, category) %>%
  do(tidy(t.test(.$diffAverageSkel_MD, mu = 0, alternative = "two.sided"))) %>%
  knitr::kable(digits = 3)
```

RandomArm	category	estimate	statistic	p.value	parameter	conf.low	conf.high	method	alternat
Olanzapine	RCT	0	3.235	0.003	25	0	0	One Sample t-test	two.side
Olanzapine	Relapse	0	0.344	0.743	6	0	0	One Sample t-test	two.side
Placebo	RCT	0	-0.487	0.635	13	0	0	One Sample t-test	two.side
Placebo	Relapse	0	2.021	0.058	18	0	0	One Sample t-test	two.side

9.9 running exploratory Tractwise analysis

No significant effects found

```
RCT_Tractwise <- RCT_MD %>%
  gather(elabel, change_MD, ends_with('_MD_change')) %>%
  filter(!str_detect(elabel, '-L'),
         !str_detect(elabel, '-R'),
```

```
        !str_detect(elabel, 'Average')) %>%
  group_by(elabel) %>%
  do(tidy(lm(change_MD ~ RandomArm + sex + age + site, data= .))) %>%
  ungroup() %>% group_by(term) %>%
  mutate(p_FDR = p.adjust(p.value, method = 'fdr'))

RCT_Tractwise_suppltable <- RCT_Tractwise %>%
  filter(p_FDR < 0.06) %>%
  arrange(p.value) %>%
  ungroup() %>% select(-term, -p.value)
RCT_Tractwise_suppltable %>% write_csv('../generated_csvs/suppltable4b_MDtractwise.csv')
RCT_Tractwise_suppltable %>%
  knitr::kable()
```

elabel	estimate	std.error	statistic	p_FDR
SS_MD_change	-0.0000541	0.0000167	-3.247844	0.0418925
FXST_MD_change	-0.0000564	0.0000186	-3.034776	0.0418925
EC_MD_change	-0.0000526	0.0000176	-2.990254	0.0418925
SLF_MD_change	-0.0000284	0.0000102	-2.788888	0.0474893
RLIC_MD_change	-0.0000537	0.0000196	-2.737613	0.0474893

```
#cleanup
rm('df', 'fit_all', 'fit_rct', 'MD', 'plot', 'RCT_FA', 'RCTRelapse_FA')
```

10 Freesurfer Derived Subcortical Volumes

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.2.0
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lme4)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
## Loading required package: methods
```

```
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'
```

```
## The following object is masked from 'package:lme4':
##
##     lmer
```

```
## The following object is masked from 'package:stats':
##
##     step
```

```
library(broom)

df <- read_csv("../generated_csvs/STOPPD_masterDF_2018-11-05.csv",na = "empty") #spreadsheet created by
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   STUDYID = col_integer()
## )
```

```
## See spec(...) for full column specifications.
```

```
FS <- read_csv('../data/fs-enigma-long_201811/LandRvolumes.csv') #bring in subcortical data, from pipel
```

```
## Parsed with column specification:
## cols(
##   SubjID = col_character(),
##   LLatVent = col_double(),
##   RLatVent = col_double(),
##   Lthal = col_double(),
##   Rthal = col_double(),
##   Lcaud = col_double(),
##   Rcaud = col_double(),
##   Lput = col_double(),
##   Rput = col_double(),
##   Lpal = col_double(),
##   Rpal = col_double(),
##   Lhippo = col_double(),
##   Rhippo = col_double(),
##   Lamyg = col_double(),
##   Ramyg = col_double(),
##   Laccumb = col_double(),
##   Raccumb = col_double(),
##   ICV = col_double()
## )
```

```
RandomArmColors = c( "#FFC200", "#007aa3")
```

```
# remove participants that did not complete first and second scan (n=74)
# then add offlabel and dateDiff (in days columns)
# + a scan is by definition offlabel if it is the third scan
# then select the cols for analysis
df <- df %>%
  filter(first_complete == "Yes",
         second_complete == "Yes",
         MR_exclusion == "No") %>%
  mutate(offLabel  = if_else(third_complete == "Yes", "Yes", ''),
         dateDiff = round(difftime(second_date, first_date, units = "days"), 0),
         STUDYID = parse_character(STUDYID),
```

```
         age = parse_number(age),
         category = factor(second_timepoint, levels = c("RCT","Relapse", "Off protocol"))) %>%
  select(STUDYID, randomization, sex, age, category, offLabel, dateDiff)
```

10.1 cleaning the CT data

```
# separating the subject id and anything afterwards to identify the longtudinal pipeline participants
# separating the subject id into site, "STUDYID" and timepoint columns
# filtering (two steps) to only include the longitudinal pipeline data
FS_long <- FS %>%
  separate(SubjID, into = c("subid", "longitudinal_pipe"), sep = '\\.', extra = "drop", fill = "right") %>%
  separate(subid, into = c("study", "site", "STUDYID", "timepoint"), fill = "right") %>%
  filter(longitudinal_pipe == "long") %>%
  filter(timepoint != "00", timepoint != "03", timepoint != "")

# adding columns that combine L and R
FS_long_plus <- FS_long %>%
  mutate(Thalamus = Lthal + Rthal,
         Hippocampus = Lhippo + Rhippo,
         Striatum = Lcaud + Rcaud + Lput + Rput)


# move CT from long to wide format
FS_wide <- FS_long_plus %>%
  gather(region, volume, -study, -site, -timepoint, -STUDYID, -longitudinal_pipe) %>%
  spread(timepoint, volume) %>%
  mutate(change = `02` - `01`,
         percchange = (`02`-`01`)/`01`) %>%
  gather(timepoint, volume, `01`, `02`, change, percchange) %>%
  unite(newcolnames, region, timepoint) %>%
  spread(newcolnames, volume)
```

```
# merge CT values with df
ana_df <- inner_join(df, FS_wide, by='STUDYID') %>%
    mutate(STUDYID = as.character(STUDYID),
         dateDiff = as.numeric(dateDiff),
         RandomArm = factor(randomization,
                        levels = c("O", "P"),
                        labels = c("Olanzapine", "Placebo")))

# write.csv
write_csv(ana_df, '../generated_csvs/STOPPD_participants_LandRVolumes_20181116.csv')
```

10.2 report any mising values from clinical trial sample

```
anti_join(df, FS_wide, by='STUDYID') %>%
  summarise(`Number of participants missing` = n()) %>%
  knitr::kable()
```

Number of participants missing
0

```r
ana_df %>%
  filter(is.na(ICV_01)) %>%
  summarise(`Number of participants missing timepoint 01` = n()) %>%
  knitr::kable()
```

Number of participants missing timepoint 01
0

```r
ana_df %>%
  filter(is.na(ICV_02)) %>%
  summarise(`Number of participants missing timepoint 02` = n()) %>%
  knitr::kable()
```

Number of participants missing timepoint 02
0

```r
library(tableone)

print(CreateTableOne(data = ana_df,
               strata = c("RandomArm"),
               vars = c("category", "Hippocampus_01", "Striatum_01", 'Thalamus_01')))
```

```
##                             Stratified by RandomArm
##                              Olanzapine         Placebo            p
##   n                                38                34
##   category (%)                                                    0.008
##      RCT                           26 (68.4)         14 (41.2)
##      Relapse                        8 (21.1)         19 (55.9)
##      Off protocol                   4 (10.5)          1 ( 2.9)
##   Hippocampus_01 (mean (sd))  7538.32 (871.37)   7390.00 (1099.58)  0.526
##   Striatum_01 (mean (sd))    16931.60 (1825.84) 16610.18 (2077.11)  0.487
##   Thalamus_01 (mean (sd))    13326.38 (1834.30) 12989.71 (1916.03)  0.449
##                             Stratified by RandomArm
##                              test
##   n
##   category (%)
##      RCT
##      Relapse
##      Off protocol
##   Hippocampus_01 (mean (sd))
##   Striatum_01 (mean (sd))
##   Thalamus_01 (mean (sd))
```

```r
print(CreateTableOne(data = ana_df,
               vars = c("category", "Hippocampus_01", "Striatum_01", 'Thalamus_01')))
```

```
##
##                              Overall
##   n                                72
##   category (%)
##      RCT                           40 (55.6)
##      Relapse                       27 (37.5)
##      Off protocol                   5 ( 6.9)
##   Hippocampus_01 (mean (sd))  7468.28 (981.44)
##   Striatum_01 (mean (sd))    16779.82 (1941.31)
##   Thalamus_01 (mean (sd))    13167.39 (1867.72)
```

Table 9: t.test for baseline group differences

brain	t	df	p.value	method
Hippocampus_01	0.63	62.82	0.53	Welch Two Sample t-test
Striatum_01	0.69	66.19	0.49	Welch Two Sample t-test
Thalamus_01	0.76	68.33	0.45	Welch Two Sample t-test

```
ana_df %>%
  select(RandomArm, Hippocampus_01, Striatum_01, Thalamus_01) %>%
  gather(brain, mm, -RandomArm) %>%
  group_by(brain) %>%
  do(tidy(t.test(mm~RandomArm, data = .))) %>%
  select(brain, statistic, parameter, p.value, method) %>%
  rename(t = statistic, df = parameter) %>%
  knitr::kable(caption = "t.test for baseline group differences", digits = 2)
```

10.3 creating an control error term calculating data frame

```
## identify the repeat control in a column and mangle the STUDYID to match in a new column
FS_long1 <- FS_long_plus %>%
  mutate(repeat_run = if_else(str_sub(STUDYID,1,1)=="R", "02", "01"),
         STUDYID = str_replace(STUDYID, 'R',""))

## extra the repeat study ids as a character vector
repeat_ids <- filter(FS_long1, repeat_run == "02")$STUDYID

## filter for only the subjects who are in the repeats list then switch to wide format
FS_wide_controls <- FS_long1 %>%
  filter(STUDYID %in% repeat_ids) %>%
  gather(region, volume, -study, -site, -timepoint, -STUDYID, -longitudinal_pipe, -repeat_run) %>%
  unite(newcolnames, region, repeat_run) %>%
  spread(newcolnames, volume)

#write.csv
  write.csv(FS_wide_controls, '../generated_csvs/STOPPD_errorControls_LandRVolumes_2018-11-05.csv', row

rm(FS_long1, repeat_ids)
```

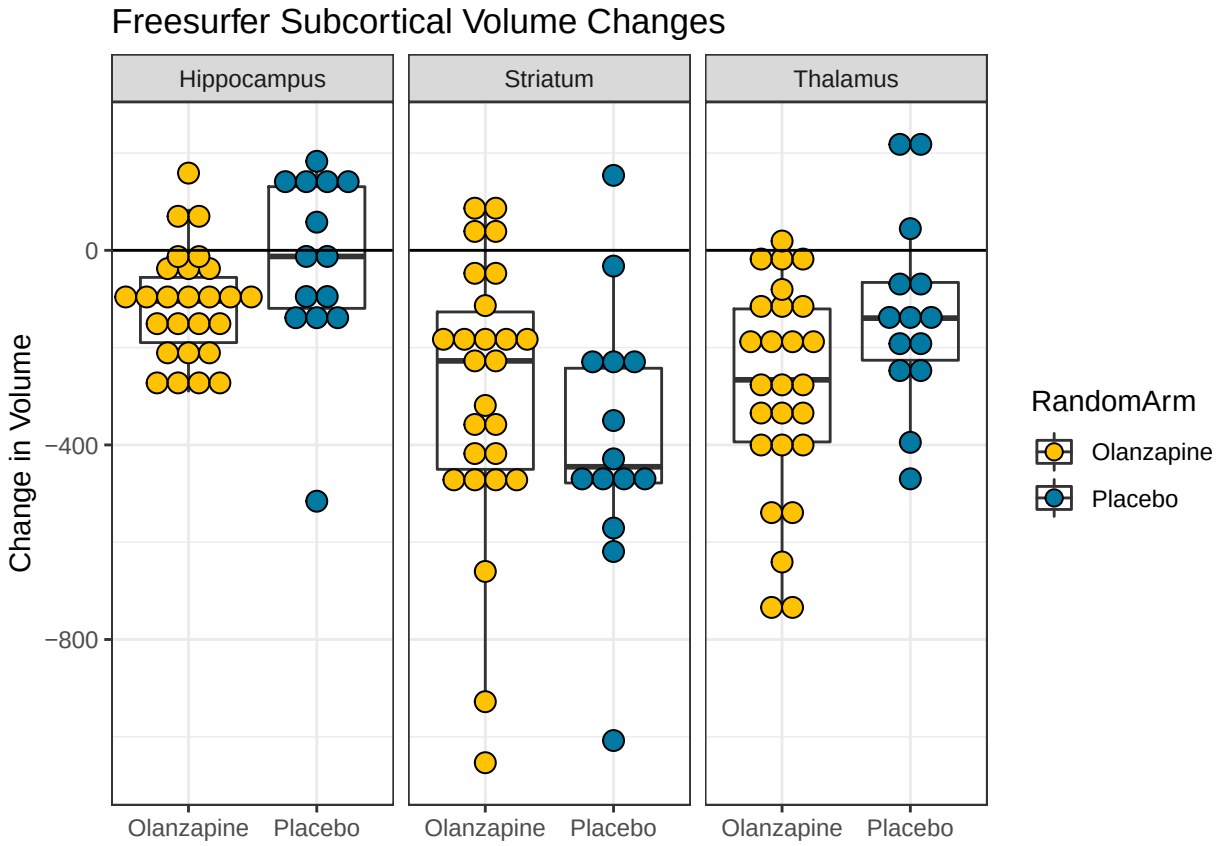
10.4 run RCT analysis (because it's simpler across volumes)

```
# make sure that STUDYID is an character not a number
# make sure that dateDiff is a number, not an interger
# label the RandomArm variable
RCT_SubCort <- ana_df %>%

  filter(category == "RCT")
```

```
#boxplot of difference in thickness (y axis) by RandomArm group (x axis)
RCT_SubCort  %>%
  gather(region, volume_change, Thalamus_change, Hippocampus_change, Striatum_change) %>%
```

```
  mutate(Region = str_replace(region, '_change','')) %>%
ggplot(aes(x= RandomArm, y = volume_change, fill = RandomArm)) +
     geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
     geom_dotplot(binaxis = 'y', stackdir = 'center') +
     geom_hline(yintercept = 0) +
     ggtitle("Freesurfer Subcortical Volume Changes") +
     xlab(NULL) +
     ylab("Change in Volume") +
     scale_fill_manual(values = RandomArmColors) +
     scale_shape_manual(values = c(21)) +
     facet_wrap(~Region) +
     theme_bw()
```

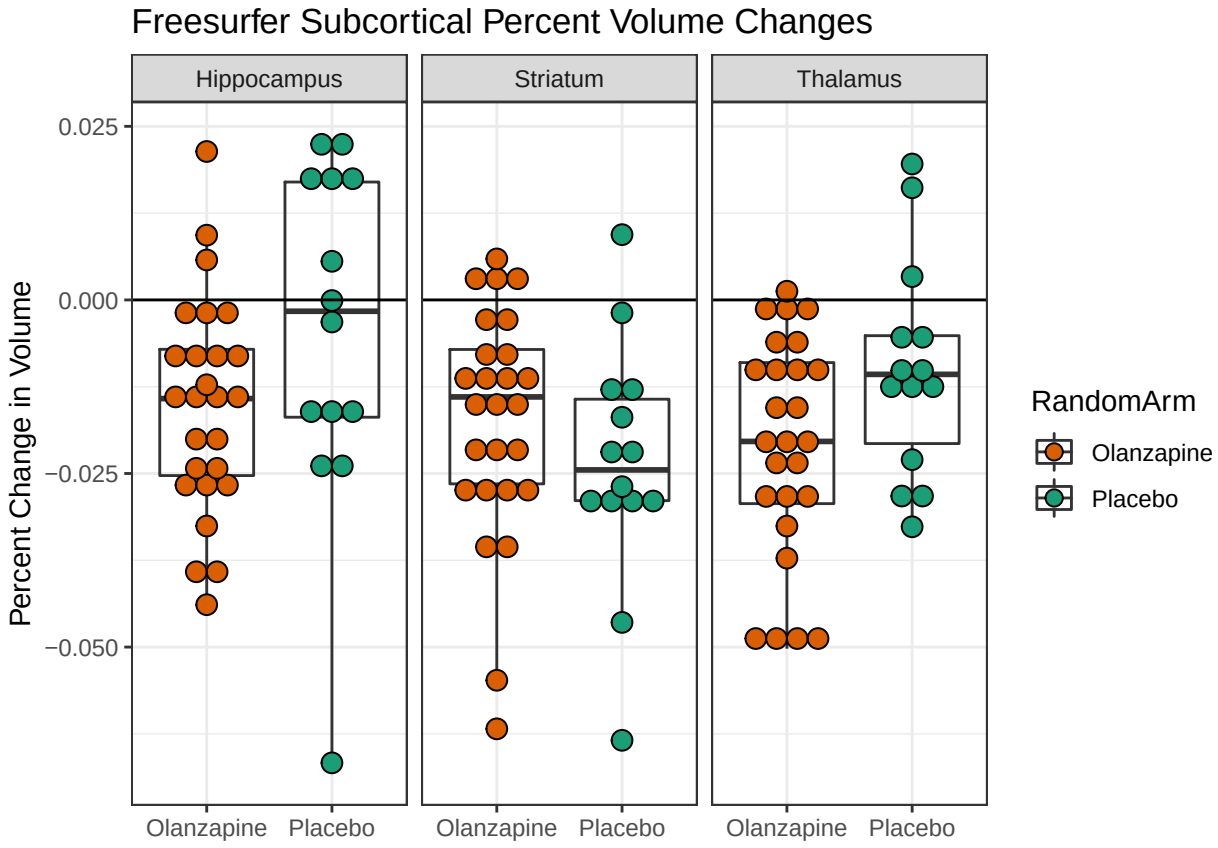
```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#boxplot of difference in thickness (y axis) by RandomArm group (x axis)
RCT_SubCort  %>%
  gather(region, volume_percchange, Thalamus_percchange, Hippocampus_percchange, Striatum_percchange) %
  mutate(Region = str_replace(region, '_percchange','')) %>%
ggplot(aes(x= RandomArm, y = volume_percchange, fill = RandomArm)) +
     geom_boxplot(outlier.shape = NA, alpha = 0.0001) +
     geom_dotplot(binaxis = 'y', stackdir = 'center') +
     geom_hline(yintercept = 0) +
     ggtitle("Freesurfer Subcortical Percent Volume Changes") +
     xlab(NULL) +
     ylab("Percent Change in Volume") +
     scale_fill_brewer(palette = "Dark2", direction = -1) +
```

```
    scale_shape_manual(values = c(21)) +
    facet_wrap(~Region) +
    theme_bw()
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



10.4.1 Running RCT Linear Models

10.4.1.1 Thalamus

```
#run linear model with covariates of sex and age
  fit_rct <- lmer(Thalamus_change ~ RandomArm + sex + age + (1|site), data= RCT_SubCort)
  summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Thalamus_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SubCort
##
## REML criterion at convergence: 504.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1283 -0.6307  0.1721  0.5984  1.4686
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
```

```
##  site     (Intercept)  4782     69.15
##  Residual             41571    203.89
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                  Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)      -182.712    139.263   29.080  -1.312   0.1998
## RandomArmPlacebo  156.222     69.401   34.664   2.251   0.0308 *
## sexM               16.157     66.777   34.519   0.242   0.8102
## age                -1.784      2.470   35.941  -0.722   0.4748
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.011
## sexM        -0.063  0.075
## age         -0.892 -0.192 -0.183
```

```
#run linear model with covariates of sex and age and site intercept
  fit_rct <- lmer(Thalamus_percchange ~ RandomArm + sex + age + (1|site), data= RCT_SubCort)
  summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Thalamus_percchange ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SubCort
##
## REML criterion at convergence: -182.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9666 -0.6349  0.1623  0.6530  1.5828
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  site     (Intercept) 2.737e-05 0.005232
##  Residual             2.181e-04 0.014767
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                    Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      -0.0071511  0.0101398 29.1678166  -0.705   0.4862
## RandomArmPlacebo  0.0116258  0.0050286 34.6583454   2.312   0.0268 *
## sexM              0.0035373  0.0048381 34.5125265   0.731   0.4696
## age              -0.0002695  0.0001793 35.9629119  -1.503   0.1415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.010
## sexM        -0.064  0.075
## age         -0.890 -0.192 -0.182
```

10.4.1.2 Striatum

```
#run linear model with covariates of sex and age
  fit_rct <- lmer(Striatum_change ~ RandomArm + sex + age + (1|site), data= RCT_SubCort)
  summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Striatum_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SubCort
##
## REML criterion at convergence: 526.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.68942 -0.49396  0.03391  0.53963  1.92558
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  site     (Intercept)     0      0.0
##  Residual             82431    287.1
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                  Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)      -169.875    180.214   36.000  -0.943    0.352
## RandomArmPlacebo  -80.518     96.821   36.000  -0.832    0.411
## sexM              -12.233     93.203   36.000  -0.131    0.896
## age                -2.318      3.343   36.000  -0.693    0.492
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

```
#run linear model with covariates of sex and age
  fit_rct <- lmer(Striatum_percchange ~ RandomArm + sex + age + (1|site), data= RCT_SubCort)
  summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Striatum_percchange ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SubCort
##
## REML criterion at convergence: -172.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.55786 -0.37884  0.03767  0.58473  2.04779
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  site     (Intercept) 0.0000000 0.00000
##  Residual             0.0002991 0.01729
```

```
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                    Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      -0.0066369  0.0108551 36.0000000  -0.611    0.545
## RandomArmPlacebo -0.0048423  0.0058319 36.0000000  -0.830    0.412
## sexM              0.0018895  0.0056140 36.0000000   0.337    0.738
## age              -0.0002271  0.0002013 36.0000000  -1.128    0.267
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

10.4.1.3 Hippocampus

```
#run linear model with covariates of sex and age
  fit_rct <- lmer(Hippocampus_change ~ RandomArm + sex + age + (1|site), data= RCT_SubCort)
  summary(fit_rct)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Hippocampus_change ~ RandomArm + sex + age + (1 | site)
##    Data: RCT_SubCort
##
## REML criterion at convergence: 476.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4916 -0.6315 -0.0032  0.5931  1.8058
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  site     (Intercept)     0      0.0
##  Residual             20379    142.8
## Number of obs: 40, groups:  site, 4
##
## Fixed effects:
##                   Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)          5.198     89.606 36.000   0.058   0.9541
## RandomArmPlacebo    97.320     48.141 36.000   2.022   0.0507 .
## sexM                -6.910     46.342 36.000  -0.149   0.8823
## age                 -2.171      1.662 36.000  -1.306   0.1998
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP sexM
## RndmArmPlcb -0.022
## sexM        -0.044  0.067
## age         -0.921 -0.181 -0.201
```

```r
#run linear model with covariates of sex and age
  fit_rct <- lm(Hippocampus_percchange ~ RandomArm + sex + age, data= RCT_SubCort)
  summary(fit_rct)
```

```
##
## Call:
## lm(formula = Hippocampus_percchange ~ RandomArm + sex + age,
##     data = RCT_SubCort)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.06398 -0.01338  0.00013  0.01082  0.03488
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.0039836  0.0119300   0.334   0.7404
## RandomArmPlacebo  0.0124023  0.0064094   1.935   0.0609 .
## sexM             -0.0001382  0.0061699  -0.022   0.9823
## age              -0.0003643  0.0002213  -1.646   0.1084
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01901 on 36 degrees of freedom
## Multiple R-squared:  0.1357, Adjusted R-squared:  0.0637
## F-statistic: 1.884 on 3 and 36 DF,  p-value: 0.1497
```

```r
#run linear model with covariates of sex and age
  fit_rct <- lm(Hippocampus_percchange ~ RandomArm + sex + age + site, data= RCT_SubCort)
  summary(fit_rct)
```

```
##
## Call:
## lm(formula = Hippocampus_percchange ~ RandomArm + sex + age +
##     site, data = RCT_SubCort)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.062098 -0.012941  0.001371  0.014436  0.035558
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.0073077  0.0137456   0.532   0.5985
## RandomArmPlacebo  0.0123147  0.0067210   1.832   0.0760 .
## sexM             -0.0009176  0.0064477  -0.142   0.8877
## age              -0.0004477  0.0002510  -1.784   0.0836 .
## siteMAS           0.0040824  0.0085714   0.476   0.6370
## siteNKI          -0.0020478  0.0081914  -0.250   0.8041
## sitePMC           0.0080461  0.0100633   0.800   0.4297
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01955 on 33 degrees of freedom
## Multiple R-squared:  0.1618, Adjusted R-squared:  0.009442
## F-statistic: 1.062 on 6 and 33 DF,  p-value: 0.4047
```

10.5 RCT & Relapse (with time as factor)

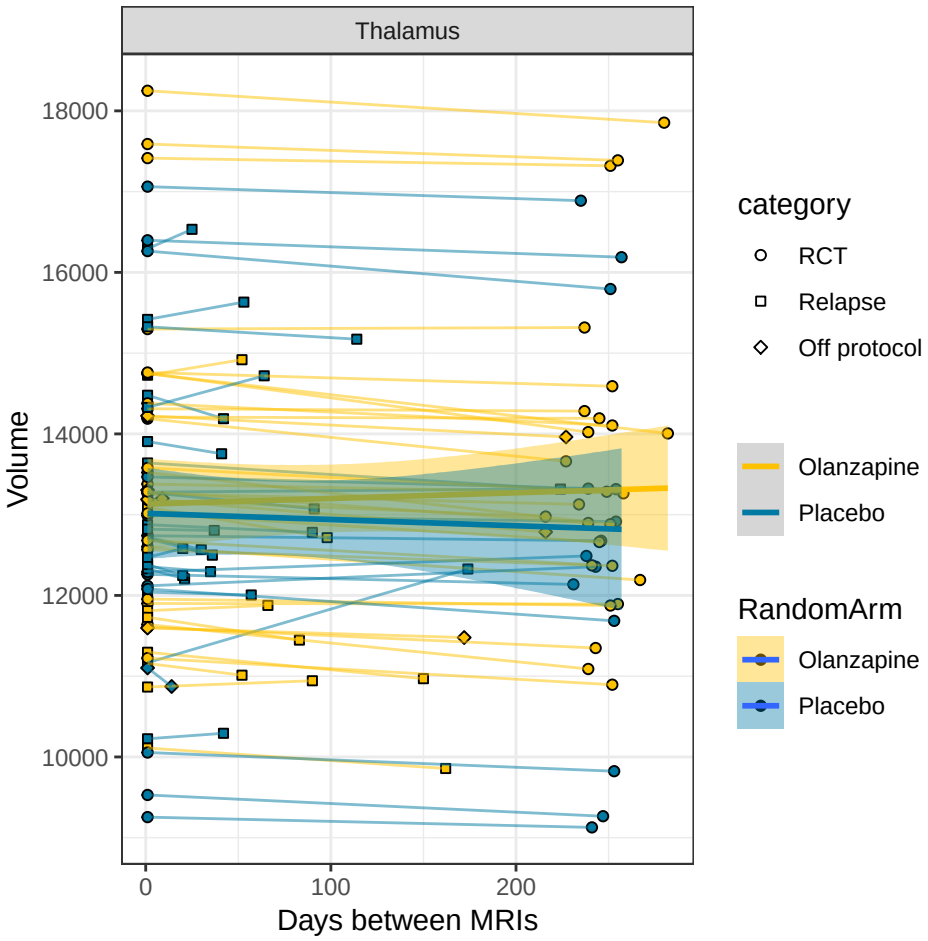
10.5.1 Thalamus

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_Thalamus <- ana_df %>%
    gather(oldcolname, volume, Thalamus_01, Thalamus_02) %>%
    mutate(model_days = if_else(oldcolname == "Thalamus_01", 1, dateDiff))

RCTRelapse_Thalamus %>% filter(model_days == 1) %>% count(RandomArm) %>% knitr::kable()
```

RandomArm	n
Olanzapine	38
Placebo	34

```
RCTRelapse_Thalamus_sense <- RCTRelapse_Thalamus %>% filter(category != "Off protocol")
```

```
 RCTRelapse_Thalamus %>%
  mutate(roi = "Thalamus") %>%
  ggplot(aes(x=model_days, y=volume, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  labs(x = "Days between MRIs", y = "Volume", colour = NULL) +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  theme_bw()  +
  facet_wrap(~roi)
```




```
#run mixed linear model, with covariates
  fit_all <- lmer(volume ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days + sex + age + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Thalamus
##
## REML criterion at convergence: 2189.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5900 -0.4277  0.0075  0.3954  3.5698
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 1725112  1313.4
##  site     (Intercept)  329065   573.6
##  Residual               29304   171.2
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
```

```
##                               Estimate Std. Error         df t value
## (Intercept)                  15820.7399   678.5347    34.2363  23.316
## RandomArmPlacebo              -154.1418   316.5539    66.4583  -0.487
## model_days                      -1.1867     0.1804    70.1254  -6.580
## sexM                          1860.4791   316.4113    65.6780   5.880
## age                            -61.0926    10.2816    65.3437  -5.942
## RandomArmPlacebo:model_days      0.8101     0.3046    70.3093   2.659
##                              Pr(>|t|)
## (Intercept)                   < 2e-16 ***
## RandomArmPlacebo              0.62791
## model_days                   7.18e-09 ***
## sexM                         1.51e-07 ***
## age                          1.20e-07 ***
## RandomArmPlacebo:model_days  0.00969 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.187
## model_days  -0.031  0.059
## sexM        -0.171  0.052  0.001
## age         -0.810 -0.064  0.004 -0.078
## RndmArmPl:_  0.017 -0.078 -0.592  0.001 -0.001
```

```
#run mixed linear model, with covariates
  fit_all <- lmer(volume ~ RandomArm*model_days + sex + age + (1|site) + (1|STUDYID), data= RCTRelapse_
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days + sex + age + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Thalamus_sense
##
## REML criterion at convergence: 2039.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4937 -0.4158  0.0095  0.3908  3.4791
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 1786865  1336.7
##  site     (Intercept)  334546   578.4
##  Residual               30876   175.7
## Number of obs: 134, groups:  STUDYID, 67; site, 4
##
## Fixed effects:
##                               Estimate Std. Error         df t value
## (Intercept)                  15947.4510   700.6484    35.0741  22.761
## RandomArmPlacebo              -196.6842   332.4836    61.3571  -0.592
## model_days                      -1.1802     0.1920    65.1012  -6.146
## sexM                          1979.3127   335.5814    60.6584   5.898
## age                            -63.4241    10.8760    60.3263  -5.832
```

```
## RandomArmPlacebo:model_days     0.8061     0.3168    65.2702   2.544
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo              0.5563
## model_days                  5.40e-08 ***
## sexM                        1.76e-07 ***
## age                         2.31e-07 ***
## RandomArmPlacebo:model_days   0.0133 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy sexM   age
## RndmArmPlcb -0.191
## model_days  -0.033  0.061
## sexM        -0.119  0.008 -0.001
## age         -0.815 -0.059  0.005 -0.125
## RndmArmPl:_  0.020 -0.080 -0.606  0.004 -0.004
```

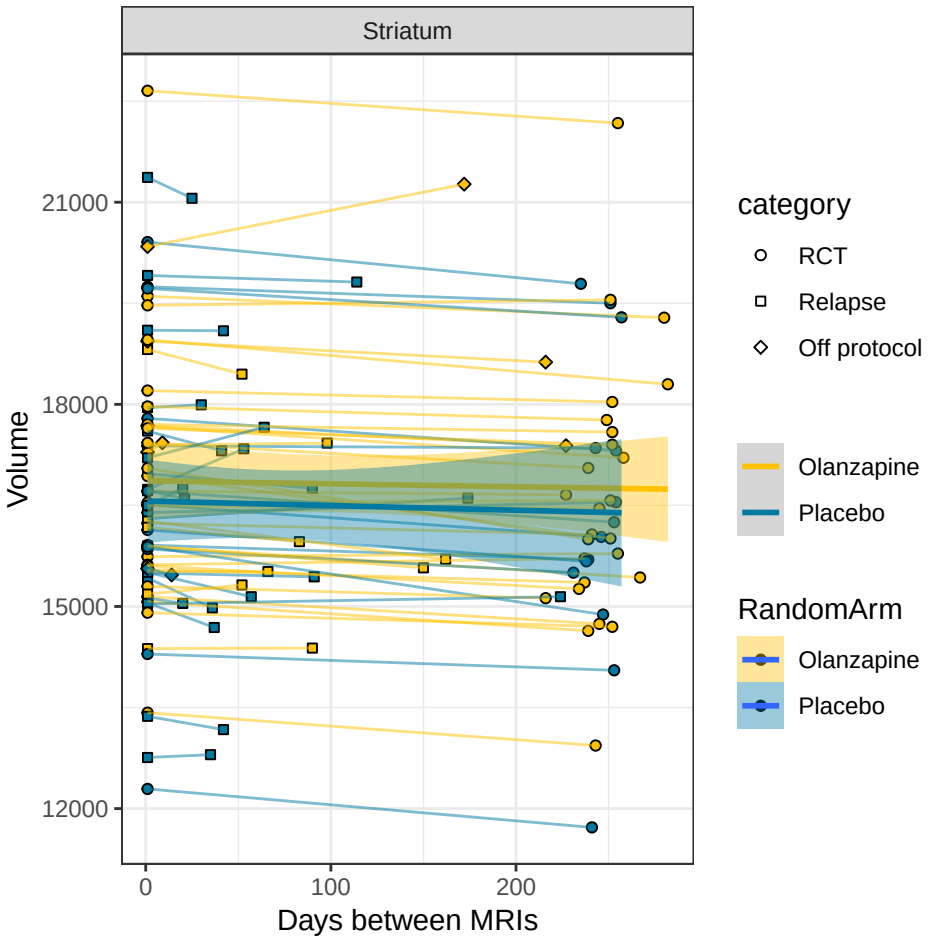
10.5.2 Striatum

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_Striatum <- ana_df %>%
    gather(oldcolname, volume, Striatum_01, Striatum_02) %>%
    mutate(model_days = if_else(oldcolname == "Striatum_01", 1, dateDiff)) %>%
  mutate(age_centered = age - mean(age),
         model_days_centered = model_days - mean(model_days))

RCTRelapse_Striatum %>% filter(model_days == 1) %>% count(RandomArm) %>% knitr::kable()
```

RandomArm	n
Olanzapine	38
Placebo	34

```
RCTRelapse_Striatum_sense <- RCTRelapse_Striatum %>% filter(category != "Off protocol")
```

```
 RCTRelapse_Striatum %>%
  mutate(roi = "Striatum") %>%
  ggplot(aes(x=model_days, y=volume, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  labs(x = "Days between MRIs", y = "Volume", colour = NULL) +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  theme_bw()  +
  facet_wrap(~roi)
```




```
  fit_all <- lmer(volume ~ RandomArm*model_days + age + sex + (1|site) + (1|STUDYID), data= RCTRelapse_S
  print(fit_all)
```

```
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: volume ~ RandomArm * model_days + age + sex + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Striatum
## REML criterion at convergence: 2250.494
## Random effects:
##  Groups   Name        Std.Dev.
##  STUDYID  (Intercept) 1673.98
##  site     (Intercept)   50.84
##  Residual              215.57
## Number of obs: 144, groups:  STUDYID, 72; site, 4
## Fixed Effects:
##                  (Intercept)             RandomArmPlacebo
##                   18785.0451                    -197.5670
##                   model_days                           age
##                      -1.1427                     -47.6647
##                         sexM  RandomArmPlacebo:model_days
##                    1604.6240                      -0.1886
```

```
  summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
```

```
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days + age + sex + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Striatum
##
## REML criterion at convergence: 2250.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.37598 -0.36232  0.03756  0.33443  2.82794
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 2802195  1673.98
##  site     (Intercept)    2585    50.84
##  Residual               46470   215.57
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                              Estimate Std. Error         df t value
## (Intercept)                 18785.0451   774.3552    64.4702  24.259
## RandomArmPlacebo             -197.5670   398.9866    68.8067  -0.495
## model_days                     -1.1427     0.2271    70.1337  -5.031
## age                           -47.6647    13.0310    67.1314  -3.658
## sexM                         1604.6240   399.1139    67.7920   4.020
## RandomArmPlacebo:model_days    -0.1886     0.3836    70.3100  -0.492
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo            0.622056
## model_days                  3.61e-06 ***
## age                         0.000501 ***
## sexM                        0.000148 ***
## RandomArmPlacebo:model_days 0.624398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy age    sexM
## RndmArmPlcb -0.202
## model_days  -0.034  0.059
## age         -0.902 -0.055  0.003
## sexM        -0.172  0.037  0.000 -0.079
## RndmArmPl:_  0.019 -0.078 -0.592 -0.001  0.002
```

```
    fit_all <- lmer(volume ~ RandomArm*model_days + age + sex + (1|site) + (1|STUDYID), data= RCTRelaps
  print(fit_all)
```

```
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: volume ~ RandomArm * model_days + age + sex + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Striatum_sense
## REML criterion at convergence: 2069.416
## Random effects:
##  Groups   Name        Std.Dev.
##  STUDYID  (Intercept) 1.534e+03
```

```
##  site     (Intercept) 5.355e-04
##  Residual             1.999e+02
## Number of obs: 134, groups:  STUDYID, 67; site, 4
## Fixed Effects:
##                  (Intercept)             RandomArmPlacebo
##                    1.921e+04                   -1.834e+01
##                   model_days                          age
##                   -1.245e+00                   -6.041e+01
##                         sexM  RandomArmPlacebo:model_days
##                    1.809e+03                   -8.351e-02
```

```
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days + age + sex + (1 | site) + (1 |
##     STUDYID)
##    Data: RCTRelapse_Striatum_sense
##
## REML criterion at convergence: 2069.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.87831 -0.38523  0.02011  0.35407  1.90856
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  STUDYID  (Intercept) 2.353e+06 1.534e+03
##  site     (Intercept) 2.868e-07 5.355e-04
##  Residual             3.996e+04 1.999e+02
## Number of obs: 134, groups:  STUDYID, 67; site, 4
##
## Fixed effects:
##                               Estimate Std. Error         df t value
## (Intercept)                  1.921e+04  7.235e+02  6.316e+01  26.558
## RandomArmPlacebo            -1.834e+01  3.783e+02  6.384e+01  -0.048
## model_days                  -1.245e+00  2.184e-01  6.511e+01  -5.700
## age                         -6.041e+01  1.243e+01  6.300e+01  -4.862
## sexM                         1.809e+03  3.815e+02  6.300e+01   4.742
## RandomArmPlacebo:model_days -8.351e-02  3.604e-01  6.527e+01  -0.232
##                             Pr(>|t|)
## (Intercept)                  < 2e-16 ***
## RandomArmPlacebo               0.961
## model_days                  3.14e-07 ***
## age                         8.09e-06 ***
## sexM                        1.25e-05 ***
## RandomArmPlacebo:model_days    0.817
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_dy age    sexM
## RndmArmPlcb -0.205
## model_days  -0.036  0.061
## age         -0.901 -0.054  0.005
```

```
## sexM        -0.115 -0.007 -0.001 -0.126
## RndmArmPl:_  0.022 -0.080 -0.606 -0.003  0.004
```

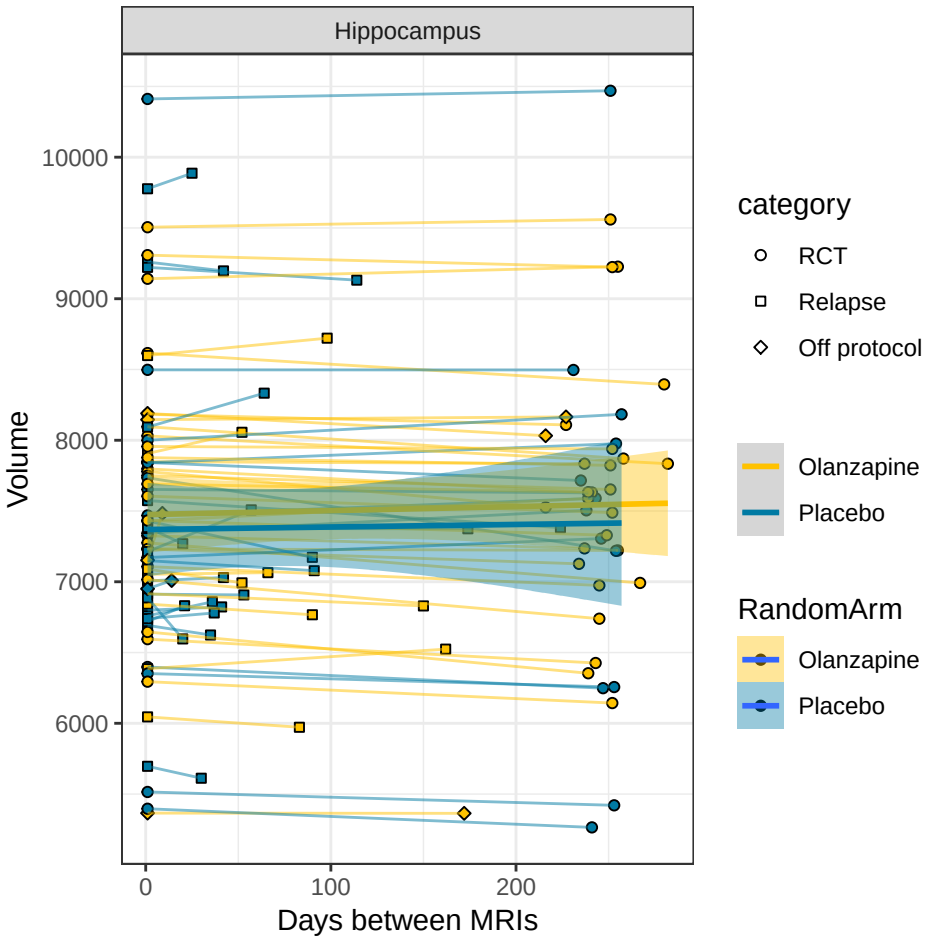
10.5.3 Hippocampus

```
#restructure data for RCT & Relapse participants (N=72)
  RCTRelapse_Hippocampus <- ana_df %>%
    gather(oldcolname, volume, Hippocampus_01, Hippocampus_02) %>%
    mutate(model_days = if_else(oldcolname == "Hippocampus_01", 1, dateDiff)) %>%
  mutate(age_centered = age - mean(age),
         model_days_centered = model_days - mean(model_days))

RCTRelapse_Hippocampus %>% filter(model_days == 1) %>% count(RandomArm) %>% knitr::kable()
```

RandomArm	n
Olanzapine	38
Placebo	34

```
RCTRelapse_Hippocampus_sense <- RCTRelapse_Hippocampus %>% filter(category != "Off protocol")
```

```
#plot all data, including outlier (participant 210030)
 RCTRelapse_Hippocampus %>%
  mutate(roi = "Hippocampus") %>%
  ggplot(aes(x=model_days, y=volume, fill = RandomArm)) +
  geom_point(aes(shape = category)) +
  geom_line(aes(group=STUDYID, color = RandomArm), alpha = 0.5) +
  geom_smooth(aes(color = RandomArm), method="lm") +
  labs(x = "Days between MRIs", y = "Volume", colour = NULL) +
  scale_colour_manual(values = RandomArmColors) +
  scale_fill_manual(values = RandomArmColors) +
  scale_shape_manual(values = c(21:23)) +
  theme_bw()  +
  facet_wrap(~roi)
```




```
#run mixed linear model, with covariates
  fit_all <- lmer(volume ~ RandomArm*model_days_centered + sex + age_centered + (1|site) + (1|STUDYID),
  print(fit_all)
```

```
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: volume ~ RandomArm * model_days_centered + sex + age_centered +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus
## REML criterion at convergence: 2049.592
## Random effects:
##  Groups   Name        Std.Dev.
##  STUDYID  (Intercept) 829.3
##  site     (Intercept) 164.5
##  Residual             100.6
## Number of obs: 144, groups:  STUDYID, 72; site, 4
## Fixed Effects:
##                          (Intercept)
##                            7238.7852
##                       RandomArmPlacebo
##                              -72.6709
##                    model_days_centered
##                               -0.4047
##                                  sexM
##                              584.5259
```

```
##                          age_centered
##                              -31.7712
## RandomArmPlacebo:model_days_centered
##                               0.2636
```

```
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days_centered + sex + age_centered +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus
##
## REML criterion at convergence: 2049.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.40824 -0.41230 -0.00417  0.40063  2.36527
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 687763   829.3
##  site     (Intercept)  27053   164.5
##  Residual              10111   100.6
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
##                                      Estimate Std. Error        df
## (Intercept)                          7238.7852   188.6898    7.7620
## RandomArmPlacebo                      -72.6709   198.2076   66.6777
## model_days_centered                    -0.4047     0.1060   70.1144
## sexM                                  584.5259   198.7201   66.5371
## age_centered                          -31.7712     6.4714   65.6248
## RandomArmPlacebo:model_days_centered    0.2636     0.1790   70.2723
##                                      t value Pr(>|t|)
## (Intercept)                           38.363 3.93e-10 ***
## RandomArmPlacebo                      -0.367 0.715048
## model_days_centered                   -3.820 0.000286 ***
## sexM                                   2.941 0.004491 **
## age_centered                          -4.909 6.36e-06 ***
## RandomArmPlacebo:model_days_centered   1.473 0.145272
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_d_ sexM   ag_cnt
## RndmArmPlcb -0.529
## mdl_dys_cnt -0.009  0.008
## sexM        -0.518  0.046  0.000
## age_centerd  0.069 -0.060  0.003 -0.079
## RndmArmP:__  0.005  0.005 -0.592  0.001 -0.001
```

```
#run mixed linear model, with covariates
  fit_all <- lmer(volume ~ RandomArm*model_days_centered*age_centered + sex + (1|site) + (1|STUDYID), da
  print(fit_all)
```

```
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: volume ~ RandomArm * model_days_centered * age_centered + sex +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus
## REML criterion at convergence: 2055.319
## Random effects:
##  Groups   Name        Std.Dev.
##  STUDYID  (Intercept) 831.9
##  site     (Intercept) 189.1
##  Residual             100.5
## Number of obs: 144, groups:  STUDYID, 72; site, 4
## Fixed Effects:
##                                         (Intercept)
##                                           7.239e+03
##                                      RandomArmPlacebo
##                                            -7.478e+01
##                                   model_days_centered
##                                            -4.255e-01
##                                         age_centered
##                                           -3.460e+01
##                                                 sexM
##                                           5.897e+02
##                RandomArmPlacebo:model_days_centered
##                                           2.785e-01
##                       RandomArmPlacebo:age_centered
##                                           6.427e+00
##                    model_days_centered:age_centered
##                                           -8.378e-03
## RandomArmPlacebo:model_days_centered:age_centered
##                                           1.802e-02
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
```

```
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days_centered * age_centered + sex +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus
##
## REML criterion at convergence: 2055.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.35631 -0.44217  0.01185  0.39228  2.31761
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 692108   831.9
##  site     (Intercept)  35772   189.1
##  Residual              10110   100.5
## Number of obs: 144, groups:  STUDYID, 72; site, 4
##
## Fixed effects:
```

```
##                                                    Estimate Std. Error
## (Intercept)                                        7.239e+03  1.958e+02
## RandomArmPlacebo                                  -7.478e+01  1.990e+02
## model_days_centered                               -4.255e-01  1.073e-01
## age_centered                                      -3.460e+01  9.044e+00
## sexM                                               5.897e+02  2.000e+02
## RandomArmPlacebo:model_days_centered               2.785e-01  1.799e-01
## RandomArmPlacebo:age_centered                      6.427e+00  1.328e+01
## model_days_centered:age_centered                  -8.378e-03  6.862e-03
## RandomArmPlacebo:model_days_centered:age_centered  1.802e-02  1.470e-02
##                                                          df t value
## (Intercept)                                        7.246e+00  36.977
## RandomArmPlacebo                                   6.544e+01  -0.376
## model_days_centered                                6.812e+01  -3.966
## age_centered                                       6.636e+01  -3.825
## sexM                                               6.530e+01   2.948
## RandomArmPlacebo:model_days_centered               6.827e+01   1.548
## RandomArmPlacebo:age_centered                      6.657e+01   0.484
## model_days_centered:age_centered                   6.810e+01  -1.221
## RandomArmPlacebo:model_days_centered:age_centered  6.836e+01   1.226
##                                                   Pr(>|t|)
## (Intercept)                                        1.6e-09 ***
## RandomArmPlacebo                                  0.708339
## model_days_centered                               0.000178 ***
## age_centered                                      0.000291 ***
## sexM                                              0.004430 **
## RandomArmPlacebo:model_days_centered              0.126208
## RandomArmPlacebo:age_centered                     0.629926
## model_days_centered:age_centered                  0.226334
## RandomArmPlacebo:model_days_centered:age_centered 0.224457
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_d_ ag_cnt sexM   RnAP:__ RnAP:_ md__:_
## RndmArmPlcb -0.514
## mdl_dys_cnt -0.008  0.008
## age_centerd  0.082 -0.036  0.002
## sexM        -0.507  0.046 -0.001 -0.104
## RndmArmP:__  0.004  0.006 -0.596 -0.002  0.002
## RndmArmPl:_ -0.049 -0.011 -0.001 -0.695  0.069  0.000
## mdl_dys_c:_  0.005 -0.002  0.157 -0.012 -0.007 -0.094   0.008
## RndmAP:__:_ -0.004  0.001 -0.074  0.006  0.008  0.005   0.031 -0.467
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
```

```
#run mixed linear model, with covariates
  fit_all <- lmer(volume ~ RandomArm*model_days_centered + sex + age_centered + (1|site) + (1|STUDYID),
  print(fit_all)
```

```
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: volume ~ RandomArm * model_days_centered + sex + age_centered +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus_sense
## REML criterion at convergence: 1903.747
```

```
## Random effects:
##  Groups   Name        Std.Dev.
##  STUDYID  (Intercept) 846.51
##  site     (Intercept)  92.00
##  Residual              99.13
## Number of obs: 134, groups:  STUDYID, 67; site, 4
## Fixed Effects:
##                           (Intercept)
##                             7254.3186
##                        RandomArmPlacebo
##                               -110.9660
##                     model_days_centered
##                                 -0.4211
##                                    sexM
##                               583.4918
##                           age_centered
##                               -31.2064
## RandomArmPlacebo:model_days_centered
##                                  0.2795
```

```
summary(fit_all)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: volume ~ RandomArm * model_days_centered + sex + age_centered +
##     (1 | site) + (1 | STUDYID)
##    Data: RCTRelapse_Hippocampus_sense
##
## REML criterion at convergence: 1903.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4428 -0.4317 -0.0118  0.4159  2.3980
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  STUDYID  (Intercept) 716581   846.51
##  site     (Intercept)   8465    92.00
##  Residual               9827    99.13
## Number of obs: 134, groups:  STUDYID, 67; site, 4
##
## Fixed effects:
##                                      Estimate Std. Error        df
## (Intercept)                         7254.3186   181.3540    7.4248
## RandomArmPlacebo                    -110.9660   208.3516   61.6101
## model_days_centered                   -0.4211     0.1083   65.0858
## sexM                                 583.4918   210.8283   61.6355
## age_centered                         -31.2064     6.8588   60.0991
## RandomArmPlacebo:model_days_centered   0.2795     0.1788   65.2177
##                                     t value Pr(>|t|)
## (Intercept)                          40.001 6.06e-10 ***
## RandomArmPlacebo                     -0.533  0.59623
## model_days_centered                  -3.887  0.00024 ***
## sexM                                  2.768  0.00745 **
## age_centered                         -4.550 2.66e-05 ***
```

```
## RandomArmPlacebo:model_days_centered   1.563  0.12278
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) RndmAP mdl_d_ sexM   ag_cnt
## RndmArmPlcb -0.569
## mdl_dys_cnt -0.011  0.010
## sexM        -0.526 -0.003 -0.001
## age_centerd  0.116 -0.056  0.004 -0.126
## RndmArmP:__  0.005  0.003 -0.606  0.003 -0.003
```