Chapter 7 Analyses and Output

RSF

December 1, 2019

#######################################################################################  
# Chapter 7 Data Analyses  
# Author: Ryan Falck  
# Last edited: 2019-11-27  
#######################################################################################  
  
# Data preparation --------------------------------------------------------  
  
if (!require("pacman")) install.packages("pacman")

## Loading required package: pacman

## Warning: package 'pacman' was built under R version 3.5.3

pacman::p\_load(readxl, mice, psych, tidyverse,nlme,lsmeans,lme4,lmerTest,dplyr)  
setwd("C:/Users/rfalck/Desktop/UBC-PhD/Thesis Material/Thesis defense materials/Theis Data and Analyses")#Directory where you put the spreadsheet  
subset.v4 <- read\_excel("Chapter 7 Data.xlsx")  
  
library(plyr)

## -------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

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

## The following object is masked from 'package:purrr':  
##   
## compact

Test<-rename(subset.v4,c("PSQI\_Sleep\_Latency\_(Time)"="PSQI\_Sleep\_Latency\_Time","PSQI\_Category:\_Sleep\_Quality"="PSQI\_Category\_Sleep\_Quality",  
 "PSQI\_Category:\_Sleep\_Latency"="PSQI\_Category\_Sleep\_Latency","PSQI\_Category:\_Sleep\_Duration"="PSQI\_Category\_Sleep\_Duration",  
 "PSQI\_Category:\_Sleep\_Efficiency"="PSQI\_Category\_Sleep\_Efficiency","PSQI\_Category:\_Sleep\_Disturbances"="PSQI\_Category\_Sleep\_Disturbances",  
 "PSQI\_Category:\_Medications"="PSQI\_Category\_Medications", "PSQI\_Category:\_Dysfunction"="PSQI\_Category\_Dysfunction",  
 "Retirement-Age"="Retirement\_Age","CHAMPS\_kcal/week\_Exercise"="CHAMPS\_kcalperweek\_Exercise",  
 "CHAMPS\_MVPA\_Hours/Week"="CHAMPS\_Exercise\_HoursperWeek", "Employed-Other"="Employed\_other"))  
detach("package:plyr", unload= TRUE)  
  
  
  
Test<-Test[c(1:5,7:40,42,46,51:79,81:84,94:109,136)]  
Test.1<-Test[c(1:23,25:91)]  
Test.Final<-Test.1[c(1:5,16:23,34:36,38,41:44,63,65:71,80:84,88)]  
Test.Final$Rest\_HR<-as.numeric(Test.Final$Rest\_HR)  
Test.Final$Retirement\_Age<-as.numeric(Test.Final$Retirement\_Age)  
  
varying <- Test.Final[c(1:2,6:14,19:35:length(Test.Final))]

## Warning in 19:35:length(Test.Final): numerical expression has 17 elements:  
## only the first used

baseline <- subset(Test.Final[-c(6:14,19:35:length(Test.Final))],Time==1)

## Warning in 19:35:length(Test.Final): numerical expression has 17 elements:  
## only the first used

varying.1 <- subset(varying,Time==1)  
varying.2 <- subset(varying,Time==2)  
varying.3 <- subset(varying,Time==3)  
  
colnames(varying.1) <- paste(colnames(varying.1),"1",sep=".")  
colnames(varying.2) <- paste(colnames(varying.2),"2",sep=".")  
colnames(varying.3) <- paste(colnames(varying.3),"3",sep=".")  
  
wide.data <- left\_join(baseline,varying.1,by=c("ID"="ID.1")) %>%   
 left\_join(.,varying.2,by=c("ID"="ID.2")) %>%   
 left\_join(.,varying.3,by=c("ID"="ID.3"))  
  
Timevars <- grep("Time",colnames(wide.data),value=TRUE)  
wide.data <- wide.data[,!(colnames(wide.data)%in%Timevars)]  
wide.data.abb <- wide.data[c(1:3,5:6,17,21,22,25:34,43,47,48,51:60,69,73,74,77:86)]  
wide.data.abb <- wide.data.abb %>%   
 mutate(female=ifelse(Sex=="F",1,0),  
 tx=ifelse(Group=="INT",1,0)) %>%   
 select(-Sex,-Group)  
  
colnames(wide.data.abb) <- (gsub("\_","",colnames(wide.data.abb)))  
colnames(wide.data.abb) <- (gsub(".1","baseline",colnames(wide.data.abb)))  
  
long.data.abb <- reshape(as.data.frame(wide.data.abb),idvar="ID",varying=17:42,direction="long",sep=".") #reshape to long data frame with T2 and T3 repeated outcome and baseline as separate time-invariant  
  
# Multiple Imputation -----------------------------------------------------  
set.seed(1234)  
ini <- mice(wide.data.abb,maxit=0,pred=quickpred(wide.data.abb,exclude="ID"))

## Warning in data.matrix(data): NAs introduced by coercion

## Warning: Number of logged events: 4

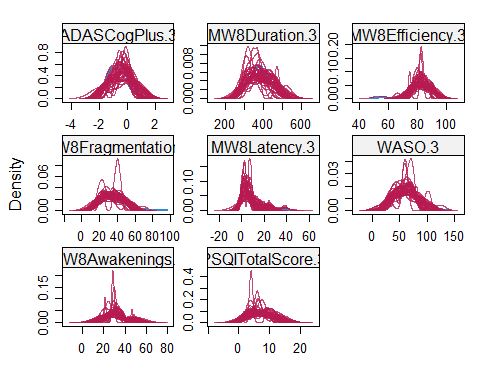
meth <- ini$meth  
meth

## ID Age Education   
## "" "" ""   
## PSQITotalScorebaseline PcntSBbaseline PcntMVPAbaseline   
## "" "pmm" "pmm"   
## ADASCogPlusbaseline ADASCogPlusSEbaseline MMSEbaseline   
## "" "" ""   
## MoCAbaseline MW8Durationbaseline MW8Efficiencybaseline   
## "" "pmm" "pmm"   
## MW8Fragmentationbaseline MW8Latencybaseline MW8Awakeningsbaseline   
## "pmm" "pmm" "pmm"   
## WASObaseline PSQITotalScore.2 PcntSB.2   
## "pmm" "pmm" "pmm"   
## PcntMVPA.2 ADASCogPlus.2 ADASCogPlusSE.2   
## "pmm" "pmm" "pmm"   
## MMSE.2 MoCA.2 MW8Duration.2   
## "" "" "pmm"   
## MW8Efficiency.2 MW8Fragmentation.2 MW8Latency.2   
## "pmm" "pmm" "pmm"   
## MW8Awakenings.2 WASO.2 PSQITotalScore.3   
## "pmm" "pmm" "pmm"   
## PcntSB.3 PcntMVPA.3 ADASCogPlus.3   
## "pmm" "pmm" "pmm"   
## ADASCogPlusSE.3 MMSE.3 MoCA.3   
## "pmm" "" ""   
## MW8Duration.3 MW8Efficiency.3 MW8Fragmentation.3   
## "pmm" "pmm" "pmm"   
## MW8Latency.3 MW8Awakenings.3 WASO.3   
## "pmm" "pmm" "pmm"   
## female tx   
## "" ""

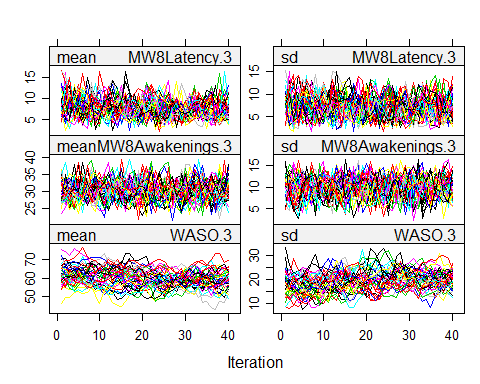
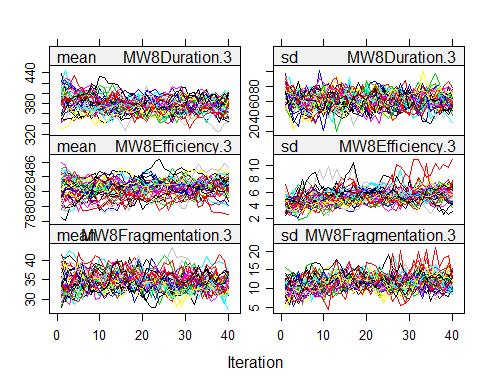
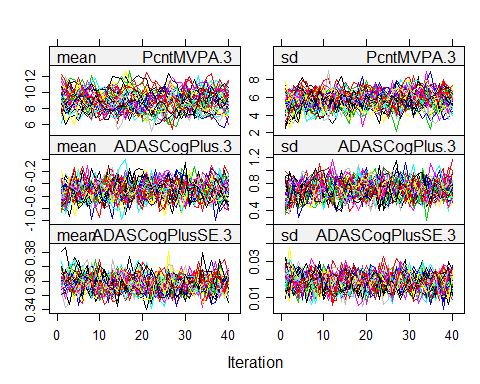
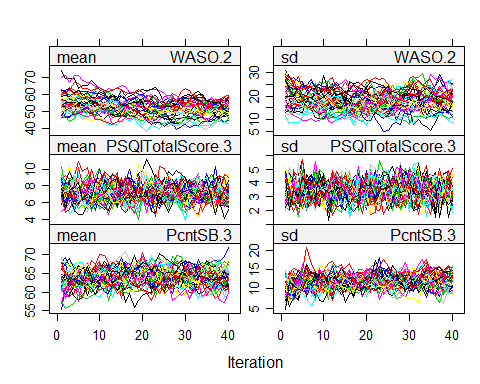
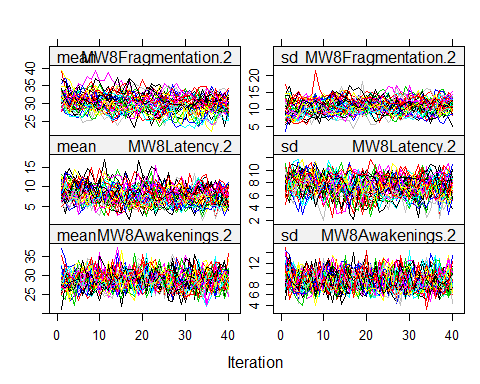
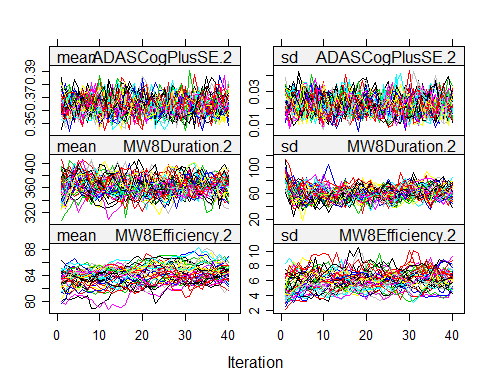
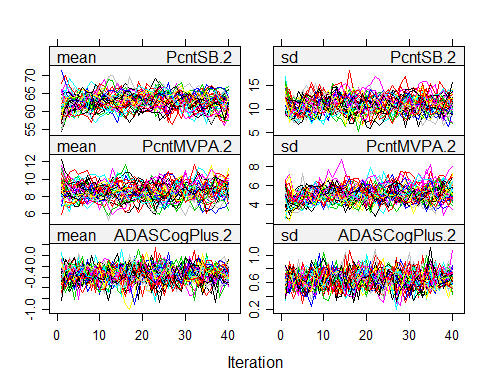
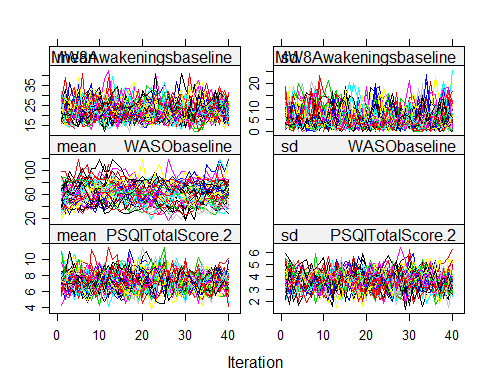
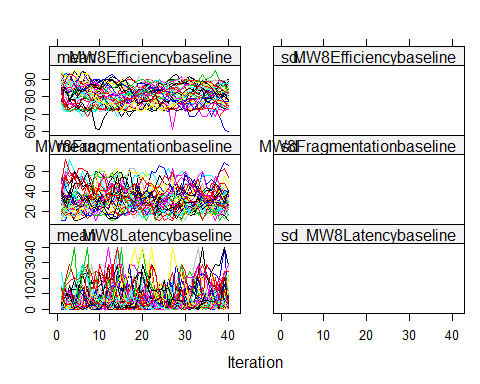
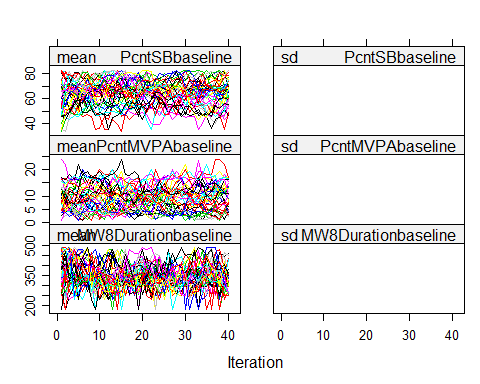
m <- 40  
wide.data.imp <- mice(wide.data.abb,m=m,maxit=40,pred=quickpred(wide.data.abb,exclude="ID"),print=FALSE,seed=1234)

## Warning in data.matrix(data): NAs introduced by coercion  
  
## Warning in data.matrix(data): Number of logged events: 4

densityplot(wide.data.imp, ~ADASCogPlus.3+MW8Duration.3+MW8Efficiency.3+MW8Fragmentation.3+MW8Latency.3+WASO.3+MW8Awakenings.3+PSQITotalScore.3)



plot(wide.data.imp)



wide <- as.list(1:m)  
for (i in 1:m){  
 wide[[i]] <- mice::complete(wide.data.imp,action=i)  
}  
  
names <- colnames(wide[[1]])  
names <- gsub("\_","",names)  
names <- gsub(".1","baseline",names)  
wide <- lapply(wide,setNames,names)  
  
long <- lapply(wide,reshape,idvar="ID",varying=17:42,direction="long",sep=".") #reshape to long data frame with T2 and T3 repeated outcome and baseline as separate time-invariant   
  
#imputed baseline values for outcomes of interest  
describe(wide.data.abb$MW8Efficiencybaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 95 82.58 7.16 83.85 83.31 6.16 60.12 95.41 35.29 -1.09 1.45  
## se  
## X1 0.73

describeBy(wide.data.abb$MW8Efficiencybaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 83.49 5.7 84.26 83.6 5.43 68.45 95.41 26.96 -0.27 -0.15  
## se  
## X1 0.82  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 47 81.65 8.35 83.29 82.78 7.54 60.12 93.53 33.41 -1.13 0.65  
## se  
## X1 1.22

describe(wide.data.abb$MW8Durationbaseline)

## vars n mean sd median trimmed mad min max range skew  
## X1 1 95 367.67 58.33 364.07 368.74 61.63 183.5 486.79 303.29 -0.27  
## kurtosis se  
## X1 0.15 5.98

describeBy(wide.data.abb$MW8Durationbaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 48 376.14 56.16 378.11 377.63 50.46 183.5 480.77 297.27 -0.53  
## kurtosis se  
## X1 1.18 8.11  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 47 359.02 59.82 356.21 359.07 66.29 232.07 486.79 254.71 -0.01  
## kurtosis se  
## X1 -0.55 8.73

describe(wide.data.abb$MW8Fragmentationbaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 95 33.25 12.86 30.21 31.85 9.82 10.82 72.57 61.74 1.02 0.73  
## se  
## X1 1.32

describeBy(wide.data.abb$MW8Fragmentationbaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 31.52 9.79 29.71 30.98 9.27 10.82 58.73 47.91 0.59 0.41  
## se  
## X1 1.41  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 47 35.01 15.29 30.54 33.57 12.95 12.39 72.57 60.17 0.87  
## kurtosis se  
## X1 -0.25 2.23

describe(wide.data.abb$MW8Latencybaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 95 6.56 7.61 4.21 5.16 4.55 0 39.71 39.71 1.85 3.65  
## se  
## X1 0.78

describeBy(wide.data.abb$MW8Latencybaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 5.47 6.03 4.21 4.52 4.71 0 24.46 24.46 1.49 1.72  
## se  
## X1 0.87  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 47 7.67 8.88 4.5 6.14 4.87 0 39.71 39.71 1.68 2.5  
## se  
## X1 1.29

describe(wide.data.abb$WASObaseline)

## vars n mean sd median trimmed mad min max range skew  
## X1 1 95 61.24 21.08 60.4 60.43 20 17.29 119.84 102.55 0.44  
## kurtosis se  
## X1 0.09 2.16

describeBy(wide.data.abb$WASObaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 48 59.74 18.38 60.47 60.14 16.31 17.29 109.99 92.69 -0.05  
## kurtosis se  
## X1 0.24 2.65  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 47 62.78 23.62 59.7 61.26 24.69 26.39 119.84 93.45 0.57  
## kurtosis se  
## X1 -0.44 3.45

describe(wide.data.abb$PSQITotalScorebaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 96 10.17 2.77 10 10.13 2.97 6 16 10 0.01 -1.07 0.28

describeBy(wide.data.abb$PSQITotalScorebaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 10.48 2.93 11 10.47 3.71 6 16 10 -0.08 -1.19  
## se  
## X1 0.42  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 48 9.85 2.59 10 9.8 2.97 6 15 9 0.06 -1.02 0.37

describe(wide.data.abb$ADASCogPlusbaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 96 -0.41 0.55 -0.43 -0.41 0.57 -1.91 0.92 2.83 0.02 -0.19  
## se  
## X1 0.06

describeBy(wide.data.abb$ADASCogPlusbaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 -0.42 0.53 -0.42 -0.44 0.6 -1.38 0.92 2.31 0.34 -0.48  
## se  
## X1 0.08  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 -0.39 0.57 -0.44 -0.39 0.38 -1.91 0.86 2.76 -0.24 -0.06  
## se  
## X1 0.08

describe(wide.data.abb$PcntMVPAbaseline)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 95 9.74 6.66 8.58 8.98 6.59 0.93 35.96 35.03 1.19 1.76  
## se  
## X1 0.68

describeBy(wide.data.abb$PcntMVPAbaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 10.27 6.83 9.86 9.67 7.8 0.93 29.92 28.99 0.71 -0.12  
## se  
## X1 0.99  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 47 9.19 6.5 8.07 8.37 5.37 1.23 35.96 34.73 1.72 4.22  
## se  
## X1 0.95

describe(wide.data.abb$PcntSBbaseline)

## vars n mean sd median trimmed mad min max range skew  
## X1 1 95 62.24 12.19 63.77 62.85 10.93 24.63 82.6 57.97 -0.51  
## kurtosis se  
## X1 0.02 1.25

describeBy(wide.data.abb$PcntSBbaseline, group= wide.data.abb$tx)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 48 61.22 12.5 61.91 61.38 12.95 34.12 82.6 48.48 -0.08 -0.77  
## se  
## X1 1.8  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 47 63.28 11.9 64.9 64.37 8.24 24.63 80.79 56.16 -0.98 1.16  
## se  
## X1 1.74

#create function to run lmm on various outcomes in multiply-imputed data  
lmm.mi.mdl <- function(y,cov,data){  
 fit <- lapply(data,FUN=function(x){  
 lme(as.formula(paste0(y, "~ factor(tx)\*factor(time)+",cov)),  
 random=~1|ID,weights=varIdent(form=~1|tx\*time),data=x)  
 })  
 coef <- sapply(fit,fixef)  
 means <- sapply(fit,lsmeans,~tx\*time)  
 contrasts <- sapply(means,contrast,"pairwise",by="time",adj="none")  
   
 mean\_control\_t2 <- mean(sapply(means,function(x){summary(x)[1,3]})) #pooled mean value for control at t2  
 mean\_tx\_t2 <- mean(sapply(means,function(x){summary(x)[2,3]}))  
 mean\_control\_t3 <- mean(sapply(means,function(x){summary(x)[3,3]}))  
 mean\_tx\_t3 <- mean(sapply(means,function(x){summary(x)[4,3]}))  
   
 bwvar\_control\_t2 <- mean(sapply(means,function(x){summary(x)[1,4]^2})) #average variance over imputations for control at t2  
 bwvar\_tx\_t2 <- mean(sapply(means,function(x){summary(x)[2,4]^2}))  
 bwvar\_control\_t3 <- mean(sapply(means,function(x){summary(x)[3,4]^2}))  
 bwvar\_tx\_t3 <- mean(sapply(means,function(x){summary(x)[4,4]^2}))  
   
 withinvar\_control\_t2 <- sd(sapply(means,function(x){summary(x)[1,3]}))^2 #variance in mean estimates across imputations for control at t2  
 withinvar\_tx\_t2 <- sd(sapply(means,function(x){summary(x)[2,3]}))^2  
 withinvar\_control\_t3 <- sd(sapply(means,function(x){summary(x)[3,3]}))^2  
 withinvar\_tx\_t3 <- sd(sapply(means,function(x){summary(x)[4,3]}))^2  
   
 dfCorrection <- (m+1)/m  
 totVar\_control\_t2 <- bwvar\_control\_t2+withinvar\_control\_t2\*dfCorrection #total variance   
 totVar\_tx\_t2 <- bwvar\_tx\_t2+withinvar\_tx\_t2\*dfCorrection  
 totVar\_control\_t3 <- bwvar\_control\_t3+withinvar\_control\_t3\*dfCorrection  
 totVar\_tx\_t3 <- bwvar\_tx\_t3+withinvar\_tx\_t3\*dfCorrection  
   
 SE\_control\_t2 <- sqrt(totVar\_control\_t2) #square root of variance  
 SE\_tx\_t2 <- sqrt(totVar\_tx\_t2)  
 SE\_control\_t3 <- sqrt(totVar\_control\_t3)  
 SE\_tx\_t3 <- sqrt(totVar\_tx\_t3)  
   
 mean\_diff\_t2 <- mean(sapply(contrasts,function(x){summary(x)[1,3]})) #mean difference at t2 with t1 adjusted  
 mean\_diff\_t3 <- mean(sapply(contrasts,function(x){summary(x)[2,3]}))  
   
 t\_ratio\_t2 <-mean(sapply(contrasts, function (x){summary(x)[1,6]}))  
 t\_ratio\_t3 <-mean(sapply(contrasts, function (x){summary(x)[2,6]}))  
 df\_t2 <-mean(sapply(contrasts, function (x){summary(x)[1,5]}))  
 df\_t3 <-mean(sapply(contrasts, function (x){summary(x)[2,5]}))  
   
 bwvar\_diff\_t2 <- mean(sapply(contrasts,function(x){summary(x)[1,4]^2})) #avarage varaince   
 bwvar\_diff\_t3 <- mean(sapply(contrasts,function(x){summary(x)[2,4]^2}))  
   
 withinvar\_diff\_t2 <- sd(sapply(contrasts,function(x){summary(x)[1,3]}))^2 #variance in mean difference estimates  
 withinvar\_diff\_t3 <- sd(sapply(contrasts,function(x){summary(x)[2,3]}))^2  
   
 totVar\_diff\_t2 <- bwvar\_diff\_t2+withinvar\_diff\_t2\*dfCorrection  
 totVar\_diff\_t3 <- bwvar\_diff\_t3+withinvar\_diff\_t3\*dfCorrection  
   
 lambda\_t2 <- (withinvar\_diff\_t2+(withinvar\_diff\_t2/m))/totVar\_diff\_t2  
 n <- nrow(wide.data.imp$data) #sample size  
 k <- 5 #number of fixed effects estimates in model  
 nu\_old\_t2 <- (m-1)/lambda\_t2^2   
 nu\_com <- n-k #degrees of freedom  
 nu\_obs\_t2 <- (nu\_com+1)/(nu\_com+3)\*nu\_com\*(1-lambda\_t2)  
 nu\_BR\_t2 <- (nu\_old\_t2\*nu\_obs\_t2)/(nu\_old\_t2+nu\_obs\_t2) #barnard-rubin degrees of freedom for T2 contrast  
   
 pvalue\_diff\_t2 <- pt(q=abs(mean\_diff\_t2)/sqrt(totVar\_diff\_t2),df=nu\_BR\_t2,lower.tail=FALSE)\*2   
 diff\_t2\_LL <- mean\_diff\_t2-sqrt(totVar\_diff\_t2)\*qt(.975,nu\_BR\_t2)  
 diff\_t2\_UL <- mean\_diff\_t2+sqrt(totVar\_diff\_t2)\*qt(.975,nu\_BR\_t2)  
   
   
   
 lambda\_t3 <- (withinvar\_diff\_t3+(withinvar\_diff\_t3/m))/totVar\_diff\_t3  
 nu\_old\_t3 <- (m-1)/lambda\_t3^2   
 nu\_obs\_t3 <- (nu\_com+1)/(nu\_com+3)\*nu\_com\*(1-lambda\_t3)  
 nu\_BR\_t3 <- (nu\_old\_t3\*nu\_obs\_t3)/(nu\_old\_t3+nu\_obs\_t3) #barnard-rubin degrees of freedom for T3 contrast  
   
 pvalue\_diff\_t3<- pt(q=abs(mean\_diff\_t3)/sqrt(totVar\_diff\_t3),df=nu\_BR\_t3,lower.tail=FALSE)\*2  
 diff\_t3\_LL <- mean\_diff\_t3-sqrt(totVar\_diff\_t3)\*qt(.975,nu\_BR\_t3)  
 diff\_t3\_UL <- mean\_diff\_t3+sqrt(totVar\_diff\_t3)\*qt(.975,nu\_BR\_t3)  
   
 return(list(Control\_adjmean\_T2=mean\_control\_t2,Control\_SE\_T2=SE\_control\_t2,  
 TX\_adjmean\_T2=mean\_tx\_t2,TX\_SE\_T2=SE\_tx\_t2,  
 Control\_adjmean\_T3=mean\_control\_t3,Control\_SE\_T3=SE\_control\_t3,  
 TX\_adjmean\_T3=mean\_tx\_t3,TX\_SE\_T3=SE\_tx\_t3,  
 Diff\_adjmean\_T2=mean\_diff\_t2, Diff\_SE\_T2=sqrt(totVar\_diff\_t2),Diff\_pvalue\_T2=pvalue\_diff\_t2, LL\_CI\_T2=diff\_t2\_LL, UL\_CI\_T2=diff\_t2\_UL,  
 Diff\_adjmean\_T3=mean\_diff\_t3,Diff\_SE\_T3=sqrt(totVar\_diff\_t3),Diff\_pvalue\_T3=pvalue\_diff\_t3, LL\_CI\_T3=diff\_t3\_LL, UL\_CI\_T3=diff\_t3\_UL))  
}  
  
#run above function on each outcome -- add as needed  
  
lmm.mi.mdl("MW8Efficiency","MW8Efficiencybaseline",long) #Efficiency

## $Control\_adjmean\_T2  
## [1] 82.53655  
##   
## $Control\_SE\_T2  
## [1] 0.6853367  
##   
## $TX\_adjmean\_T2  
## [1] 83.50567  
##   
## $TX\_SE\_T2  
## [1] 0.8348408  
##   
## $Control\_adjmean\_T3  
## [1] 81.51749  
##   
## $Control\_SE\_T3  
## [1] 0.6676786  
##   
## $TX\_adjmean\_T3  
## [1] 82.79142  
##   
## $TX\_SE\_T3  
## [1] 0.6972999  
##   
## $Diff\_adjmean\_T2  
## [1] -0.9691212  
##   
## $Diff\_SE\_T2  
## [1] 1.082859  
##   
## $Diff\_pvalue\_T2  
## [1] 0.3734414  
##   
## $LL\_CI\_T2  
## [1] -3.123472  
##   
## $UL\_CI\_T2  
## [1] 1.18523  
##   
## $Diff\_adjmean\_T3  
## [1] -1.273923  
##   
## $Diff\_SE\_T3  
## [1] 0.9679943  
##   
## $Diff\_pvalue\_T3  
## [1] 0.1917969  
##   
## $LL\_CI\_T3  
## [1] -3.199337  
##   
## $UL\_CI\_T3  
## [1] 0.6514919

lmm.mi.mdl("MW8Duration","MW8Durationbaseline",long) #Duration

## $Control\_adjmean\_T2  
## [1] 346.5117  
##   
## $Control\_SE\_T2  
## [1] 7.846513  
##   
## $TX\_adjmean\_T2  
## [1] 366.0236  
##   
## $TX\_SE\_T2  
## [1] 7.006658  
##   
## $Control\_adjmean\_T3  
## [1] 376.0639  
##   
## $Control\_SE\_T3  
## [1] 6.011652  
##   
## $TX\_adjmean\_T3  
## [1] 383.3626  
##   
## $TX\_SE\_T3  
## [1] 6.939301  
##   
## $Diff\_adjmean\_T2  
## [1] -19.5119  
##   
## $Diff\_SE\_T2  
## [1] 10.37048  
##   
## $Diff\_pvalue\_T2  
## [1] 0.06355897  
##   
## $LL\_CI\_T2  
## [1] -40.15116  
##   
## $UL\_CI\_T2  
## [1] 1.12736  
##   
## $Diff\_adjmean\_T3  
## [1] -7.298709  
##   
## $Diff\_SE\_T3  
## [1] 9.115629  
##   
## $Diff\_pvalue\_T3  
## [1] 0.4256686  
##   
## $LL\_CI\_T3  
## [1] -25.43711  
##   
## $UL\_CI\_T3  
## [1] 10.83969

lmm.mi.mdl("MW8Fragmentation","MW8Fragmentationbaseline",long) #Fragmentation

## $Control\_adjmean\_T2  
## [1] 32.54822  
##   
## $Control\_SE\_T2  
## [1] 1.007887  
##   
## $TX\_adjmean\_T2  
## [1] 31.48828  
##   
## $TX\_SE\_T2  
## [1] 1.5022  
##   
## $Control\_adjmean\_T3  
## [1] 35.5028  
##   
## $Control\_SE\_T3  
## [1] 1.288072  
##   
## $TX\_adjmean\_T3  
## [1] 32.40422  
##   
## $TX\_SE\_T3  
## [1] 1.096583  
##   
## $Diff\_adjmean\_T2  
## [1] 1.059948  
##   
## $Diff\_SE\_T2  
## [1] 1.797675  
##   
## $Diff\_pvalue\_T2  
## [1] 0.5570577  
##   
## $LL\_CI\_T2  
## [1] -2.51594  
##   
## $UL\_CI\_T2  
## [1] 4.635836  
##   
## $Diff\_adjmean\_T3  
## [1] 3.098574  
##   
## $Diff\_SE\_T3  
## [1] 1.711189  
##   
## $Diff\_pvalue\_T3  
## [1] 0.07392438  
##   
## $LL\_CI\_T3  
## [1] -0.3067195  
##   
## $UL\_CI\_T3  
## [1] 6.503868

lmm.mi.mdl("MW8Latency","MW8Latencybaseline",long) #Latency

## $Control\_adjmean\_T2  
## [1] 7.266777  
##   
## $Control\_SE\_T2  
## [1] 1.124491  
##   
## $TX\_adjmean\_T2  
## [1] 5.359127  
##   
## $TX\_SE\_T2  
## [1] 0.907026  
##   
## $Control\_adjmean\_T3  
## [1] 7.278417  
##   
## $Control\_SE\_T3  
## [1] 0.9984197  
##   
## $TX\_adjmean\_T3  
## [1] 6.6916  
##   
## $TX\_SE\_T3  
## [1] 1.124827  
##   
## $Diff\_adjmean\_T2  
## [1] 1.907649  
##   
## $Diff\_SE\_T2  
## [1] 1.447505  
##   
## $Diff\_pvalue\_T2  
## [1] 0.1915222  
##   
## $LL\_CI\_T2  
## [1] -0.9755697  
##   
## $UL\_CI\_T2  
## [1] 4.790868  
##   
## $Diff\_adjmean\_T3  
## [1] 0.5868164  
##   
## $Diff\_SE\_T3  
## [1] 1.505124  
##   
## $Diff\_pvalue\_T3  
## [1] 0.6976692  
##   
## $LL\_CI\_T3  
## [1] -2.408796  
##   
## $UL\_CI\_T3  
## [1] 3.582429

lmm.mi.mdl("WASO","WASObaseline",long) #WASO

## $Control\_adjmean\_T2  
## [1] 56.52093  
##   
## $Control\_SE\_T2  
## [1] 2.529967  
##   
## $TX\_adjmean\_T2  
## [1] 56.93936  
##   
## $TX\_SE\_T2  
## [1] 2.616893  
##   
## $Control\_adjmean\_T3  
## [1] 65.84516  
##   
## $Control\_SE\_T3  
## [1] 2.43142  
##   
## $TX\_adjmean\_T3  
## [1] 61.44057  
##   
## $TX\_SE\_T3  
## [1] 2.360335  
##   
## $Diff\_adjmean\_T2  
## [1] -0.4184341  
##   
## $Diff\_SE\_T2  
## [1] 3.64106  
##   
## $Diff\_pvalue\_T2  
## [1] 0.9087899  
##   
## $LL\_CI\_T2  
## [1] -7.662114  
##   
## $UL\_CI\_T2  
## [1] 6.825246  
##   
## $Diff\_adjmean\_T3  
## [1] 4.404588  
##   
## $Diff\_SE\_T3  
## [1] 3.414673  
##   
## $Diff\_pvalue\_T3  
## [1] 0.2008294  
##   
## $LL\_CI\_T3  
## [1] -2.39161  
##   
## $UL\_CI\_T3  
## [1] 11.20079

lmm.mi.mdl("PSQITotalScore","PSQITotalScorebaseline",long) #PSQI

## $Control\_adjmean\_T2  
## [1] 8.488765  
##   
## $Control\_SE\_T2  
## [1] 0.4665303  
##   
## $TX\_adjmean\_T2  
## [1] 7.173214  
##   
## $TX\_SE\_T2  
## [1] 0.4675999  
##   
## $Control\_adjmean\_T3  
## [1] 7.88512  
##   
## $Control\_SE\_T3  
## [1] 0.4385058  
##   
## $TX\_adjmean\_T3  
## [1] 6.529464  
##   
## $TX\_SE\_T3  
## [1] 0.4730331  
##   
## $Diff\_adjmean\_T2  
## [1] 1.315552  
##   
## $Diff\_SE\_T2  
## [1] 0.6546442  
##   
## $Diff\_pvalue\_T2  
## [1] 0.04775835  
##   
## $LL\_CI\_T2  
## [1] 0.01330057  
##   
## $UL\_CI\_T2  
## [1] 2.617802  
##   
## $Diff\_adjmean\_T3  
## [1] 1.355656  
##   
## $Diff\_SE\_T3  
## [1] 0.642477  
##   
## $Diff\_pvalue\_T3  
## [1] 0.03792045  
##   
## $LL\_CI\_T3  
## [1] 0.07745342  
##   
## $UL\_CI\_T3  
## [1] 2.633858

lmm.mi.mdl("ADASCogPlus","ADASCogPlusbaseline",long) #ADAS-cog-plus

## $Control\_adjmean\_T2  
## [1] -0.5563165  
##   
## $Control\_SE\_T2  
## [1] 0.05519709  
##   
## $TX\_adjmean\_T2  
## [1] -0.5632247  
##   
## $TX\_SE\_T2  
## [1] 0.05849618  
##   
## $Control\_adjmean\_T3  
## [1] -0.7256596  
##   
## $Control\_SE\_T3  
## [1] 0.05485462  
##   
## $TX\_adjmean\_T3  
## [1] -0.5813806  
##   
## $TX\_SE\_T3  
## [1] 0.06067236  
##   
## $Diff\_adjmean\_T2  
## [1] 0.006908173  
##   
## $Diff\_SE\_T2  
## [1] 0.0806132  
##   
## $Diff\_pvalue\_T2  
## [1] 0.9319189  
##   
## $LL\_CI\_T2  
## [1] -0.1534727  
##   
## $UL\_CI\_T2  
## [1] 0.167289  
##   
## $Diff\_adjmean\_T3  
## [1] -0.1442789  
##   
## $Diff\_SE\_T3  
## [1] 0.08274934  
##   
## $Diff\_pvalue\_T3  
## [1] 0.08559578  
##   
## $LL\_CI\_T3  
## [1] -0.3092979  
##   
## $UL\_CI\_T3  
## [1] 0.02073998

lmm.mi.mdl("PcntMVPA","PcntMVPAbaseline",long) #MVPA

## $Control\_adjmean\_T2  
## [1] 9.537228  
##   
## $Control\_SE\_T2  
## [1] 0.3851672  
##   
## $TX\_adjmean\_T2  
## [1] 9.185523  
##   
## $TX\_SE\_T2  
## [1] 0.4012403  
##   
## $Control\_adjmean\_T3  
## [1] 9.30945  
##   
## $Control\_SE\_T3  
## [1] 0.50105  
##   
## $TX\_adjmean\_T3  
## [1] 9.377648  
##   
## $TX\_SE\_T3  
## [1] 0.478305  
##   
## $Diff\_adjmean\_T2  
## [1] 0.3517042  
##   
## $Diff\_SE\_T2  
## [1] 0.552066  
##   
## $Diff\_pvalue\_T2  
## [1] 0.5260477  
##   
## $LL\_CI\_T2  
## [1] -0.7483286  
##   
## $UL\_CI\_T2  
## [1] 1.451737  
##   
## $Diff\_adjmean\_T3  
## [1] -0.06819785  
##   
## $Diff\_SE\_T3  
## [1] 0.683561  
##   
## $Diff\_pvalue\_T3  
## [1] 0.920771  
##   
## $LL\_CI\_T3  
## [1] -1.427968  
##   
## $UL\_CI\_T3  
## [1] 1.291572

lmm.mi.mdl("PcntSB","PcntSBbaseline",long) #SB

## $Control\_adjmean\_T2  
## [1] 61.07397  
##   
## $Control\_SE\_T2  
## [1] 0.9553696  
##   
## $TX\_adjmean\_T2  
## [1] 61.11533  
##   
## $TX\_SE\_T2  
## [1] 0.8439177  
##   
## $Control\_adjmean\_T3  
## [1] 62.78393  
##   
## $Control\_SE\_T3  
## [1] 0.9779479  
##   
## $TX\_adjmean\_T3  
## [1] 62.27288  
##   
## $TX\_SE\_T3  
## [1] 1.127799  
##   
## $Diff\_adjmean\_T2  
## [1] -0.04135993  
##   
## $Diff\_SE\_T2  
## [1] 1.264557  
##   
## $Diff\_pvalue\_T2  
## [1] 0.9739923  
##   
## $LL\_CI\_T2  
## [1] -2.559167  
##   
## $UL\_CI\_T2  
## [1] 2.476447  
##   
## $Diff\_adjmean\_T3  
## [1] 0.5110463  
##   
## $Diff\_SE\_T3  
## [1] 1.482171  
##   
## $Diff\_pvalue\_T3  
## [1] 0.7311283  
##   
## $LL\_CI\_T3  
## [1] -2.437346  
##   
## $UL\_CI\_T3  
## [1] 3.459438

#~~~~~~~~~~~~~~~~~~~~~Change Score Analyses~~~~~~~~~~~~~~~~~~#  
tester<-mice::complete(wide.data.imp, action="long", include = TRUE)  
  
#CREATE NEW CHANGE SCORE VARIABLES (Higher scores = better)  
tester$ADASCogPlus.Change <- tester$ADASCogPlusbaseline - tester$ADASCogPlus.3  
tester$PSQI.Change <- tester$PSQITotalScorebaseline - tester$PSQITotalScore.3  
tester$MW8Efficiency.Change <- tester$MW8Efficiency.3 - tester$MW8Efficiencybaseline  
tester$MW8Duration.Change <- tester$MW8Duration.3 - tester$MW8Durationbaseline  
tester$MW8Fragmentation.Change <- tester$MW8Fragmentationbaseline - tester$MW8Fragmentation.3  
tester$MW8Latency.Change <- tester$MW8Latencybaseline - tester$MW8Latency.3  
tester$WASO.Change <- tester$WASObaseline - tester$WASO.3  
tester$SB.Change <- tester$PcntSBbaseline - tester$PcntSB.3  
tester$MVPA.Change <- tester$PcntMVPA.3 - tester$PcntMVPAbaseline  
  
change.scores<-as.mids(tester)

## Warning in data.matrix(x): NAs introduced by coercion

#Change Score Analyses: Relationship between Changes in Sleep and Changes in Cognitive function  
lm.efficiency <- with(change.scores, lm(ADASCogPlus.Change~MW8Efficiency.Change + Age + female + MW8Efficiencybaseline + tx))  
summary(pool(lm.efficiency))

## estimate std.error statistic df  
## (Intercept) 1.544887065 0.676185747 2.2847081 82.67892  
## MW8Efficiency.Change 0.006174505 0.009791397 0.6306051 62.24098  
## Age -0.012720285 0.007356014 -1.7292361 80.85440  
## female 0.024078990 0.085671217 0.2810628 73.52429  
## MW8Efficiencybaseline -0.003590444 0.006055544 -0.5929184 73.65101  
## tx -0.175886750 0.082422875 -2.1339555 78.14530  
## p.value  
## (Intercept) 0.02489213  
## MW8Efficiency.Change 0.53060748  
## Age 0.08758318  
## female 0.77945170  
## MW8Efficiencybaseline 0.55505206  
## tx 0.03598258

lm.efficiency.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + MW8Efficiencybaseline + tx))  
pool.r.squared(lm.efficiency, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.1023215 0.01470263 0.2439082 NaN

pool.r.squared(lm.efficiency.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.09432647 0.01162906 0.232985 NaN

lm.fragmentation <- with(change.scores, lm(ADASCogPlus.Change~MW8Fragmentation.Change + Age + female + MW8Fragmentationbaseline + tx))  
summary(pool(lm.fragmentation))

## estimate std.error statistic df  
## (Intercept) 1.1633325545 0.555038274 2.09595015 81.65943  
## MW8Fragmentation.Change 0.0005064997 0.005765116 0.08785594 59.16150  
## Age -0.0135602768 0.007253248 -1.86954539 81.51848  
## female 0.0395521934 0.087639782 0.45130410 70.13105  
## MW8Fragmentationbaseline 0.0040383260 0.003320245 1.21627339 73.81532  
## tx -0.1738103916 0.082719347 -2.10120603 79.36847  
## p.value  
## (Intercept) 0.03918310  
## MW8Fragmentation.Change 0.93028789  
## Age 0.06513748  
## female 0.65316287  
## MW8Fragmentationbaseline 0.22775587  
## tx 0.03879596

lm.fragmentation.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + MW8Fragmentationbaseline + tx))  
pool.r.squared(lm.fragmentation, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.1082754 0.01679634 0.2529206 NaN

pool.r.squared(lm.fragmentation.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.105047 0.01562867 0.248116 NaN

lm.duration <- with(change.scores, lm(ADASCogPlus.Change~MW8Duration.Change + Age + female + MW8Durationbaseline + tx))  
summary(pool(lm.duration))

## estimate std.error statistic df  
## (Intercept) 1.315767e+00 0.6117175368 2.15093960 78.99761  
## MW8Duration.Change -8.499568e-04 0.0010449957 -0.81335913 59.52136  
## Age -1.362423e-02 0.0072461590 -1.88020063 81.18515  
## female 2.983568e-02 0.0822179966 0.36288499 75.72192  
## MW8Durationbaseline -2.935009e-05 0.0007992292 -0.03672299 71.74156  
## tx -1.474973e-01 0.0829395065 -1.77837244 77.56614  
## p.value  
## (Intercept) 0.03453744  
## MW8Duration.Change 0.41925395  
## Age 0.06366727  
## female 0.71770180  
## MW8Durationbaseline 0.97080787  
## tx 0.07926130

lm.duration.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + MW8Durationbaseline + tx))  
pool.r.squared(lm.duration, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.09899036 0.01110051 0.2472188 NaN

pool.r.squared(lm.duration.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.08708226 0.00770696 0.2283926 NaN

lm.latency <- with(change.scores, lm(ADASCogPlus.Change~MW8Latency.Change + Age + female + MW8Latencybaseline + tx))  
summary(pool(lm.latency))

## estimate std.error statistic df p.value  
## (Intercept) 1.365443966 0.534567115 2.5542985 82.88537 0.01246960  
## MW8Latency.Change 0.003690661 0.006229992 0.5924023 60.70334 0.55578155  
## Age -0.014729083 0.007162098 -2.0565317 82.99213 0.04287278  
## female 0.040132338 0.081621479 0.4916884 78.04855 0.62431962  
## MW8Latencybaseline 0.002806406 0.007267652 0.3861504 68.71671 0.70057833  
## tx -0.174811409 0.082292452 -2.1242703 76.56796 0.03687546

lm.latency.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + MW8Latencybaseline + tx))  
pool.r.squared(lm.latency, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.1060707 0.0154568 0.2512822 NaN

pool.r.squared(lm.latency.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.0988114 0.0134041 0.2388765 NaN

lm.WASO <- with(change.scores, lm(ADASCogPlus.Change~WASO.Change + Age + female + WASObaseline + tx))  
summary(pool(lm.WASO))

## estimate std.error statistic df p.value  
## (Intercept) 1.073706304 0.590738801 1.8175652 80.18661 0.07286588  
## WASO.Change 0.002787025 0.002775466 1.0041646 67.56423 0.31888213  
## Age -0.011216428 0.007425375 -1.5105537 81.21905 0.13478317  
## female 0.017125876 0.085401798 0.2005330 75.84221 0.84160056  
## WASObaseline 0.001202124 0.002130808 0.5641636 76.61469 0.57429067  
## tx -0.176919960 0.082391771 -2.1473013 76.47458 0.03493780

lm.WASO.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + WASObaseline + tx))  
pool.r.squared(lm.WASO, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.1139126 0.01946014 0.2596572 NaN

pool.r.squared(lm.WASO.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.09939967 0.01384502 0.2389951 NaN

lm.PSQI <- with(change.scores, lm(ADASCogPlus.Change~PSQI.Change + Age + female + PSQITotalScorebaseline + tx))  
summary(pool(lm.PSQI))

## estimate std.error statistic df  
## (Intercept) 1.4833222975 0.577704986 2.56761208 77.80023  
## PSQI.Change -0.0009382077 0.014590177 -0.06430406 63.06617  
## Age -0.0145300441 0.007329947 -1.98228499 80.81533  
## female 0.0291331452 0.082491312 0.35316622 75.17737  
## PSQITotalScorebaseline -0.0108094467 0.015200452 -0.71112666 76.04946  
## tx -0.1671863904 0.085464016 -1.95621967 73.51358  
## p.value  
## (Intercept) 0.01215659  
## PSQI.Change 0.94893151  
## Age 0.05084595  
## female 0.72495221  
## PSQITotalScorebaseline 0.47918123  
## tx 0.05423860

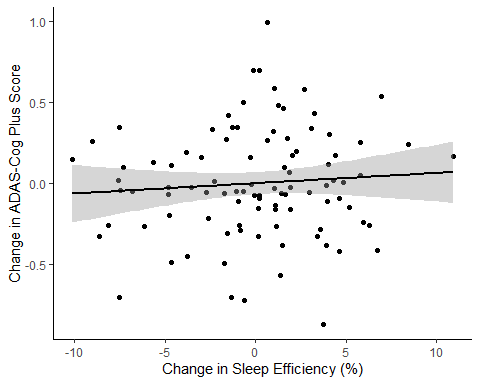
lm.PSQI.1 <- with(change.scores, lm(ADASCogPlus.Change~ Age + female + PSQITotalScorebaseline + tx))  
pool.r.squared(lm.PSQI, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.09423262 0.009943502 0.2388278 NaN

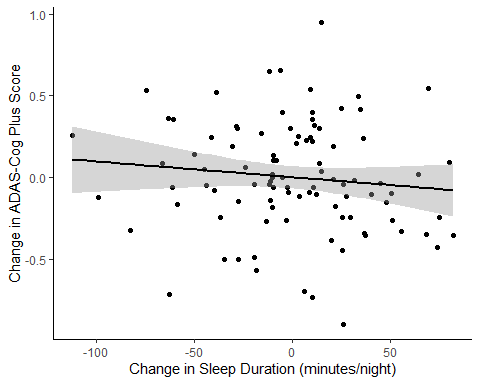
pool.r.squared(lm.PSQI.1, adjusted = FALSE)

## est lo 95 hi 95 fmi  
## R^2 0.09132425 0.008766946 0.2355716 NaN

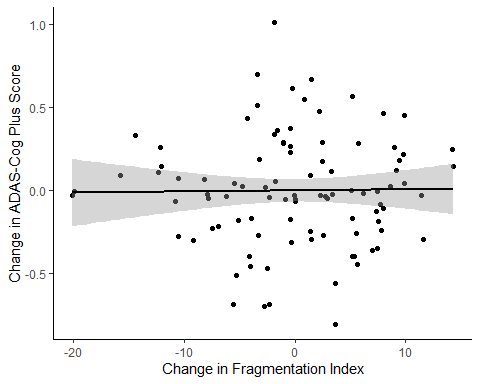
#GRAPH RESIDUAL CHANGE SCORES#  
  
#Efficiency  
lm.efficiency.1<-with(change.scores, lm(ADASCogPlus.Change~Age+ female + MW8Efficiencybaseline + tx))  
lm.efficiency.2<-with(change.scores, lm(MW8Efficiency.Change~Age + female + MW8Efficiencybaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.efficiency.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.efficiency.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
efficiency.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(efficiency.adas.graph)<-c("ADASCog","Efficiency")  
ggplot(data = efficiency.adas.graph, aes(x = efficiency.adas.graph$Efficiency, y = efficiency.adas.graph$ADASCog)) +   
 labs(x="Change in Sleep Efficiency (%)", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))



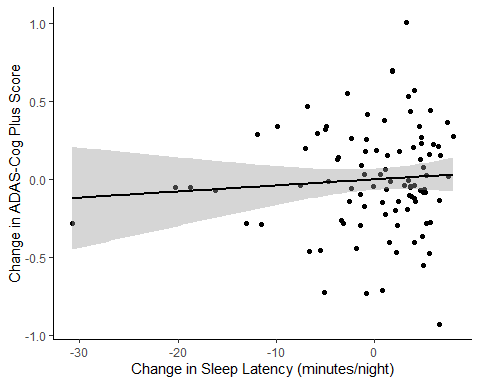
#Duration  
lm.duration.1<-with(change.scores, lm(ADASCogPlus.Change~Age + female + MW8Durationbaseline + tx))  
lm.duration.2<-with(change.scores, lm(MW8Duration.Change~Age + female + MW8Durationbaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.duration.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.duration.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
duration.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(duration.adas.graph)<-c("ADASCog","Duration")  
ggplot(data = duration.adas.graph, aes(x = duration.adas.graph$Duration, y = duration.adas.graph$ADASCog)) +   
 labs(x="Change in Sleep Duration (minutes/night)", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))



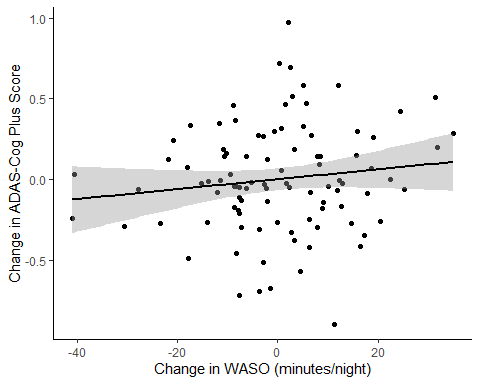
#Fragmentation  
lm.fragmentation.1<-with(change.scores, lm(ADASCogPlus.Change~Age+ female + MW8Fragmentationbaseline + tx))  
lm.fragmentation.2<-with(change.scores, lm(MW8Fragmentation.Change~Age + female + MW8Fragmentationbaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.fragmentation.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.fragmentation.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
fragmentation.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(fragmentation.adas.graph)<-c("ADASCog","Fragmentation")  
ggplot(data = fragmentation.adas.graph, aes(x = fragmentation.adas.graph$Fragmentation, y = fragmentation.adas.graph$ADASCog)) +   
 labs(x="Change in Fragmentation Index", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))



#Latency  
lm.latency.1<-with(change.scores, lm(ADASCogPlus.Change~Age + female + MW8Latencybaseline + tx))  
lm.latency.2<-with(change.scores, lm(MW8Latency.Change~Age + female + MW8Latencybaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.latency.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.latency.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
latency.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(latency.adas.graph)<-c("ADASCog","Latency")  
ggplot(data = latency.adas.graph, aes(x = latency.adas.graph$Latency, y = latency.adas.graph$ADASCog)) +   
 labs(x="Change in Sleep Latency (minutes/night)", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))



#WASO  
lm.WASO.1<-with(change.scores, lm(ADASCogPlus.Change~Age + female + WASObaseline + tx))  
lm.WASO.2<-with(change.scores, lm(WASO.Change~Age + female + WASObaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.WASO.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.WASO.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
WASO.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(WASO.adas.graph)<-c("ADASCog","WASO")  
ggplot(data = WASO.adas.graph, aes(x = WASO.adas.graph$WASO, y = WASO.adas.graph$ADASCog)) +   
 labs(x="Change in WASO (minutes/night)", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))



#PSQI  
lm.PSQI.1<-with(change.scores, lm(ADASCogPlus.Change~Age + female + PSQITotalScorebaseline + tx))  
lm.PSQI.2<-with(change.scores, lm(PSQI.Change~Age + female + PSQITotalScorebaseline + tx))  
  
  
RS1=NULL  
RS2=NULL  
for (i in 1:40){  
 RS1=rbind(RS1,residuals(lm.PSQI.1$analyses[[i]]))  
 RS1mean=colMeans(RS1)  
 RS2=rbind(RS2,residuals(lm.PSQI.2$analyses[[i]]))  
 RS2mean=colMeans(RS2)  
}  
  
  
PSQI.adas.graph<-as.data.frame(cbind(RS1mean,RS2mean))  
colnames(PSQI.adas.graph)<-c("ADASCog","PSQI")  
ggplot(data = PSQI.adas.graph, aes(x = PSQI.adas.graph$PSQI, y = PSQI.adas.graph$ADASCog)) +   
 labs(x="Change in PSQI Total Score", y="Change in ADAS-Cog Plus Score") +geom\_point(color='black') + geom\_smooth(method= 'lm', se = TRUE, color= 'black') +  
 theme\_bw() + theme(panel.border = element\_blank(), panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(), axis.line = element\_line(colour = "black"))

