```
###################################################
##12/29/15 - BAG
##Simulation Code for Informed Presence Analysis
##With Figures
####################################################


library(RColorBrewer)
library(foreach)
library(doParallel)
library(doRNG)
cl <- makeCluster(10)  ###Make appropriate for computing environment
registerDoParallel(cl)

############################
##Simulation 1
##Confounding and M-Bias
############################


###No Association
ps <- c(.1,.25,.5,.75,.9,1)
CFS.ALL <- CFS <- vector("list",6)
Bs <- c(.1,0)  ##No Association
n <- 2000
set.seed(99)

for(k in 1:6){
    for(j in 1:6){
        cfs <- foreach(i = c(1:500), .combine = 'rbind') %dorng% {
            DM <- sample(rep(c(0,1),n/2))
            DEP <- rbinom(n,1,1/(1 + exp(-cbind(1,DM) %*% Bs)))
            dx <- DM+DEP
            nvis <- sapply(dx, function(x)trunc(rlnorm(1,.5*(x+1)))) #People with more conditions have 
more visits
            dat <- data.frame(cbind(DEP,DM,nvis))
            dat$DEPo <- apply(dat,1,function(x)ifelse(ifelse(x[1] == 1 & x[3] > 0, 
max(rbinom(x[3],1,ps[k])),0) ==0,0,1))
            dat$DMo <- apply(dat,1,function(x)ifelse(ifelse(x[2] == 1 & x[3] > 0, 
max(rbinom(x[3],1,ps[j])),0) ==0,0,1))

            c(coef(glm(DEP ~ DM, data = dat, family = binomial))[2],
              coef(glm(DEP ~ DM, data = subset(dat,nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo, data = subset(dat, nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo + nvis, data = subset(dat, nvis > 0), family = binomial))[2],
        }
        colnames(cfs) <- c("True", "True.Observed","NoAdjust","Adjust")
        CFS[[j]] <- cfs
    }
    names(CFS) <- paste("j",ps)
    CFS.ALL[[k]] <- CFS
    print(k)

}
names(CFS.ALL) <- paste("k",ps)

Bias <- lapply(CFS.ALL, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,mean))))
Bias.sd <- lapply(CFS.ALL, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,quantile,p = c(.025,.975)))))

do.call("rbind",Bias)


####Association
ps <- c(.1,.25,.5,.75,.9,1)
CFS.ALL.Assoc <- CFS <- vector("list",6)
n <- 2000
Bs <- c(.1,log(1.25)) ###Assocition


set.seed(99)

for(k in 1:6){
```

```
    for(j in 1:6){
        cfs <- foreach(i = c(1:500), .combine = 'rbind') %dorng% {
            DM <- sample(rep(c(0,1),n/2))
            DEP <- rbinom(n,1,1/(1 + exp(-cbind(1,DM) %*% Bs)))
            dx <- DM+DEP
            nvis <- sapply(dx, function(x)trunc(rlnorm(1,.5*(x+1))))
            dat <- data.frame(cbind(DEP,DM,nvis))
            dat$DEPo <- apply(dat,1,function(x)ifelse(ifelse(x[1] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[k])),0) ==0,0,1))
            dat$DMo <- apply(dat,1,function(x)ifelse(ifelse(x[2] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[j])),0) ==0,0,1))
            c(coef(glm(DEP ~ DM, data = dat, family = binomial))[2],
              coef(glm(DEP ~ DM, data = subset(dat,nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo, data = subset(dat, nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo + nvis, data = subset(dat, nvis > 0), family = binomial))[2],
           }
        colnames(cfs) <- c("True", "True.Observed","NoAdjust","Adjust")
        CFS[[j]] <- cfs
    }
    names(CFS) <- paste("j",ps)
    CFS.ALL.Assoc[[k]] <- CFS
    print(k)

}
names(CFS.ALL.Assoc) <- paste("k",ps)

Bias.Assoc <- lapply(CFS.ALL.Assoc, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,mean))))
Bias.Assoc.sd <- lapply(CFS.ALL.Assoc, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,quantile,p = c(.025,.975)))))


###No Association, No Confounding
ps <- c(.1,.25,.5,.75,.9,1)
CFS.ALL.nc <- CFS <- vector("list",6)
set.seed(99)
Bs <- c(.1,0) ##No Association
n <- 2000

for(k in 1:6){
    for(j in 1:6){
        cfs <- foreach(i = c(1:500), .combine = 'rbind') %dorng% {
            DM <- sample(rep(c(0,1),n/2))
            DEP <- rbinom(n,1,1/(1 + exp(-cbind(1,DM) %*% Bs)))
            dx <- DM+DEP
            nvis <-  trunc(rlnorm(n,1)) ###Equal number Visits
            dat <- data.frame(cbind(DEP,DM,nvis))
            dat$DEPo <- apply(dat,1,function(x)ifelse(ifelse(x[1] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[k])),0) ==0,0,1))
            dat$DMo <- apply(dat,1,function(x)ifelse(ifelse(x[2] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[j])),0) ==0,0,1))
            c(coef(glm(DEP ~ DM, data = dat, family = binomial))[2],
              coef(glm(DEP ~ DM, data = subset(dat,nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo, data = subset(dat, nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo + nvis, data = subset(dat, nvis > 0), family = binomial))[2],
          }
        colnames(cfs) <- c("True", "True.Observed","NoAdjust","Adjust")
        CFS[[j]] <- cfs
    }
    names(CFS) <- paste("j",ps)
    CFS.ALL.nc[[k]] <- CFS
    print(k)

}
names(CFS.ALL.nc) <- paste("k",ps)

Bias.nc <- lapply(CFS.ALL.nc, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,mean))))
Bias.sd.nc <- lapply(CFS.ALL.nc, function(z)t(sapply(z, function(x)apply(apply(x,1,function(y)y-
Bs[2]),1,quantile,p = c(.025,.975)))))

####Association, No Confounding
ps <- c(.1,.25,.5,.75,.9,1)
CFS.ALL.Assoc.nc <- CFS <- vector("list",6)
```

```
n <- 2000
Bs <- c(.1,log(1.25))  ##Create an Association
set.seed(99)

for(k in 1:6){
    for(j in 1:6){
        cfs <- foreach(i = c(1:500), .combine = 'rbind') %dorng% {
            DM <- sample(rep(c(0,1),n/2))
            DEP <- rbinom(n,1,1/(1 + exp(-cbind(1,DM) %*% Bs)))
            nvis <- trunc(rlnorm(n,1))###Make equal
            dat <- data.frame(cbind(DEP,DM,nvis))
            dat$DEPo <- apply(dat,1,function(x)ifelse(ifelse(x[1] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[j])),0) ==0,0,1))
            dat$DMo <- apply(dat,1,function(x)ifelse(ifelse(x[2] == 1 & x[3] > 0,
max(rbinom(x[3],1,ps[j])),0) ==0,0,1))
            c(coef(glm(DEP ~ DM, data = dat, family = binomial))[2],
              coef(glm(DEP ~ DM, data = subset(dat,nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo, data = subset(dat, nvis > 0), family = binomial))[2],
              coef(glm(DEPo ~ DMo + nvis, data = subset(dat, nvis > 0), family = binomial))[2],
          }
        colnames(cfs) <- c("True", "True.Observed","NoAdjust","Adjust")
        CFS[[j]] <- cfs
    }
    names(CFS) <- paste("j",ps)
    CFS.ALL.Assoc.nc[[k]] <- CFS
    print(k)

}
names(CFS.ALL.Assoc.nc) <- paste("k",ps)

Bias.Assoc.nc <- lapply(CFS.ALL.Assoc.nc, function(z)t(sapply(z,
function(x)apply(apply(x,1,function(y)y -Bs[2]),1,mean))))
Bias.sd.Assoc.nc <- lapply(CFS.ALL.Assoc.nc, function(z)t(sapply(z,
function(x)apply(apply(x,1,function(y)y-Bs[2]),1,quantile,p = c(.025,.975)))))

#############Plots
rn <- range(c(range(Bias.sd),range(Bias.Assoc.sd),range(Bias.sd.nc), range(Bias.sd.Assoc.nc)))
cols <- brewer.pal(4,"Dark2")[c(1,3)]

###Figure 3
par(mfrow = c(2,4))
frame()
title("Scenario 1/3\nUnder Alternative", cex.main = 1.5)
legend("center", legend = c("No Adjustment","Adjustment"),col = cols, lwd = 2, cex = 1)
for(i in 1:6){
    plot(ps,Bias.Assoc[[i]][,3], xlim = c(0,1.05), ylim = rn, col = cols[1], ylab = "Bias", xlab =
"Prob Diab", main = paste("Prob Hyp = ", ps[i]))
    segments(ps,Bias.Assoc.sd[[i]][,5], ps, Bias.Assoc.sd[[i]][,6], col = cols[1], lwd = 2)
    points(ps+.05,Bias.Assoc[[i]][,4], col = cols[2])
    segments(ps+.05,Bias.Assoc.sd[[i]][,7], ps+.05, Bias.Assoc.sd[[i]][,8], col = cols[2], lwd = 2)
    abline(h = 0, lty = 2)
}


###Figure 4
par(mfrow = c(2,4))
frame()
title("Scenario 2/4\nUnder Alternative", cex.main = 1.5)
legend("center", legend = c("No Adjustment","Adjustment"),col = cols, lwd = 2, cex = 1)
for(i in 1:6){
    plot(ps,Bias.Assoc.nc[[i]][,3], xlim = c(0,1.05), ylim = rn, col = cols[1], ylab = "Bias", xlab =
"Prob Diab", main = paste("Prob Hyp = ", ps[i]))
    segments(ps,Bias.sd.Assoc.nc[[i]][,5], ps, Bias.sd.Assoc.nc[[i]][,6], col = cols[1], lwd = 2)
    points(ps+.05,Bias.Assoc.nc[[i]][,4], col = cols[2])
    segments(ps+.05,Bias.sd.Assoc.nc[[i]][,7], ps+.05, Bias.sd.Assoc.nc[[i]][,8], col = cols[2], lwd =
2)
    abline(h = 0, lty = 2)
}
dev.off()

#############################################
######Sim 2 - Disease Severity Confounding
###########################
library(MASS)
```

```
rho <- c(0,.1,.25,.5,.75,.9,1)
CFS.ALL <-  vector("list",7)
Bs <- c(.1,.5)
n <- 2000
set.seed(99)


for(j in 1:7){
    cfs <- foreach(i = c(1:500), .combine = 'rbind', .packages = 'MASS') %dorng% {
        dd <- mvrnorm(n, mu = c(0,5),Sigma = matrix(c(1,rho[j],rho[j],1), ncol = 2)) ###Create
Correlation
        DS <- dd[,1]
        nvis <- round(dd[,2])
        DM <- rbinom(n, 1, 1/(1 + exp(-cbind(1,DS) %*% Bs)))
        DEP <- rbinom(n,1,1/(1 + exp(-cbind(1,DS) %*% Bs)))
        dat <- data.frame(cbind(DEP,DM,DS,nvis))
        c(coef(glm(DEP ~ DM, data = dat, family = binomial))[2],
          coef(glm(DEP ~ DM + DS, data = dat, family = binomial))[2],
          coef(glm(DEP ~ DM + nvis, data = dat, family = binomial))[2],
          coef(glm(DEP ~ DM + nvis + DS, data = dat, family = binomial))[2])
    }
    colnames(cfs) <- c("NoAdjust", "DS-Adjusted","NVIS-Adjusted","Double-Adjust")
    CFS.ALL[[j]] <- cfs
    print(j)

}

bias2 <- sapply(CFS.ALL,function(x)apply(x,2,mean))
se2 <- lapply(CFS.ALL, function(x)apply(x,2,quantile, p = c(0.025,.975)))


cols <- brewer.pal(4,"Dark2")

###Figure 5

plot(seq(1,1.6,length = 4),bias2[,1], xlim = c(1,9), ylim = range(se2), xaxt = "n", col = cols, pch =
15, ylab = "Bias", xlab = "Correlation between Disease Severity & Number of Visits")
segments(seq(1,1.6,length = 4),se2[[1]][1,],seq(1,1.6,length = 4),se2[[1]][2,], col = cols, lwd = 2)
for(i in 2:7){
    points(seq(i,i+.6,length = 4),bias2[,i], col = cols, pch = 15)
    segments(seq(i,i + .6,length = 4),se2[[i]][1,],seq(i,i + .6,length = 4),se2[[i]][2,], col = cols,
lwd = 2)
}
abline(h = 0, lty = 2)
abline(v = c(2:7) - .1)
axis(1, at = seq(1.4, 7.4, length = 7), rho)
legend("topright", legend = c("Unadjusted", "Adjusted For General Health", "Adjusted for # Visits",
"Adjusted for Both"), col = cols, lwd = 2, cex = .9)
```