```
## Test if the chisq difference test applys when missing data are available
## Generate data in R and analyze data in Mplus
## The chisq value is read in for density plot.

library(mvtnorm)

gen.model<-function(T){
## generate mplus scripts for data analysis
## Mplus input scripts
mplus.null<-'m1null.inp'
mplus.alt<-'m2alt.inp'

## for null model
cat('TITLE: Null model\n',file=mplus.null);
cat('DATA:\n',file=mplus.null, append=T);
cat('  FILE=data.txt;\n',file=mplus.null, append=T);
cat('VARIABLE: \n',file=mplus.null, append=T);
cat('    NAMES ARE y1-y',T, ' x;\n', sep='', file=mplus.null, append=T);
cat('    USEVARIABLES ARE y1-y',T, ' x;\n', sep='', file=mplus.null, append=T);
cat('    MISSING = ALL(999); \n',file=mplus.null, append=T);

cat('ANALYSIS: \n',file=mplus.null, append=T);
cat('    COVERAGE=.01; \n',file=mplus.null, append=T);

cat('MODEL: \n',file=mplus.null, append=T);
cat('    i s q| ',file=mplus.null, append=T);
for (i in 1:T){
  cat('y',i,'@',i-1,' ',sep='',file=mplus.null,append=T)
}
cat(';\n',file=mplus.null,append=T)

cat('    [i * 50, s *, q * ]; \n',file=mplus.null, append=T);
cat('    i * 100, s *, q * ; \n',file=mplus.null, append=T);
cat('    q on x; \n',file=mplus.null, append=T);
cat('    i with s * 0; \n',file=mplus.null, append=T);
cat('    i with q * 0; \n',file=mplus.null, append=T);
cat('    s with q * 0; \n',file=mplus.null, append=T);
cat('    y1-y',T, ' (10);\n', sep='', file=mplus.null, append=T);
cat('    [y1-y',T, '@0];\n', sep='', file=mplus.null, append=T);

cat('SAVEDATA:\n', sep='', file=mplus.null, append=T);
cat('    results=res.txt;\n', sep='', file=mplus.null, append=T);

## for alternative model
cat('TITLE: Null model\n',file=mplus.alt);
cat('DATA:\n',file=mplus.alt, append=T);
cat('  FILE=data.txt;\n',file=mplus.alt, append=T);
cat('VARIABLE: \n',file=mplus.alt, append=T);
cat('    NAMES ARE y1-y',T, ' x;\n', sep='', file=mplus.alt, append=T);
cat('    USEVARIABLES ARE y1-y',T, ' x;\n', sep='', file=mplus.alt, append=T);
cat('    MISSING = ALL(999); \n',file=mplus.alt, append=T);

```

Function to generate Mplus scripts for M0 and M1
This function require the input of number of measurement occasions

```
cat('ANALYSIS: \n',file=mplus.alt, append=T);
cat('    COVERAGE=.01; \n',file=mplus.alt, append=T);

cat('MODEL: \n',file=mplus.alt, append=T);
cat('    i s q| ',file=mplus.alt, append=T);
for (i in 1:T){
  cat('y',i,'@',i-1,' ',sep='',file=mplus.alt,append=T)
}
cat(';\n',file=mplus.alt,append=T)

cat('    [i * 50, s *, q *]; \n',file=mplus.alt, append=T);
cat('    i * 100, s *, q *; \n',file=mplus.alt, append=T);
cat('    q on x @ 0; \n',file=mplus.alt, append=T);
cat('    i with s * 0; \n',file=mplus.alt, append=T);
cat('    i with q * 0; \n',file=mplus.alt, append=T);
cat('    q with s * 0; \n',file=mplus.alt, append=T);
cat('    y1-y',T, ' (10);\n', sep='', file=mplus.alt, append=T);
cat('    [y1-y',T, '@0];\n', sep='', file=mplus.alt, append=T);

cat('SAVEDATA:\n', sep='', file=mplus.alt, append=T);
cat('    results=res.txt;\n', sep='', file=mplus.alt, append=T);
}

## Data generation

## quadratic model with covariate
grm.quadratic<-function(N, T, R){
## Constants and parameters
mL<-50
vL<-100
mS<-5
vS<-25
vLS<-0
mQ<--1
vQ<-9
vLQ<-0
vSQ<-0

bQ<-0

vE<-25

sigma<-array(c(vL,vLS,vLQ, vLS,vS,vSQ, vLQ,vSQ,vQ), dim=c(3,3))


y<-array(NA, dim=c(N, T))
x<-rep(c(0,1),N/2)

for (i in 1:N){
  mu<-c(mL,mS,mQ+x[i]*bQ)
  LSQ<-rmvnorm(1, mu, sigma)
  for (j in 1:T){
    y[i, j] <- LSQ[1] + LSQ[2]*(j-1)/(T-1) + LSQ[3]*((j-1)/(T-1))^2 + rnorm(1, 0, sqrt(vE))
```

Function to generate complete data for the quadratic growth curve model
The population parameter values

```
  }
}

```

Save generated data into a file

```
data.file<-paste('Q-N-',N,'-T-',T,'-',R,'.txt',sep='')
write.table(cbind(y,x), data.file, row.names=F, col.names=F)

}


```

Generate 10000 sets of complete data

```
for (i in 1:10000){
  grm.quadratic(500,5,i)
}

## process data
```

Generate missing data according to missing data rates and missing mechanisms
alpha=0 for MAR
alpha=1 for MANR
alpha=0.5 for mixture of MAR and MNAR

```
data.gen<-function(i,N,T, r=0.5, alpha=0){
##read in data
data.file<-paste('Q-N-500-T-5-',i,'.txt',sep='')
  y<-read.table(data.file)
##missing data or complete data
  if (r>0){
    for (i in 2:T){
      ## find which element is mssing for y[,t-1]
      ind.c<-NULL
      ind.m<-NULL
      for (j in 1:N){
          if (!is.na(y[j,i-1])){
             ind.c<-c(ind.c,j)
          }else{
             ind.m<-c(ind.m,j)}
      }

      if (length(ind.m)!=0){
         y[ind.m,i]<-NA }

      cut.r<-quantile(y[1:N,i-1],r,na.rm=T)

      for (k in 1:N){
         if ( !is.na(y[k,i])){
            if (y[k,i-1]>cut.r){
               if (runif(1)<r*alpha){y[k,i]<-NA}
            }else{
               if (runif(1)<1-alpha+r*alpha){y[k,i]<-NA}
            }
         }
      }
    }
  }
  write.table(cbind(y[1:N,1:T],y[1:N,6]),'data.txt',row.names=F,col.names=F,na='999')
}

```

Procedure to run the simulation and obtain chi-square statistics

```
## Run Mplus and get the chi-square statistics
ptm <- proc.time()
R<-10000
```

```

N.v<-c(500,400, 300, 200, 100)                                                                    sample sizes
T.v<-4:5                                                                                           number of measurement occasions
alpha.v<-c(0, 0.5, 1)                                                                              missing mechanisms
r.v<-c(.3,.1, 0)                                                                                   missing data rate

for (T in T.v){
  gen.model(T)                                                                                     Generate mplus model scripts
  for (N in N.v){
    for (r in r.v){

       for (alpha in alpha.v){
            res.file<-paste('res-N',N,'-T',T,'-alpha',alpha,'-R',r,'-a.txt',sep='')                Name of the file to save the results from mplus
            for (i in 1:R){
                data.gen(i,N,T,r,alpha)                                                            generate missing data
                ## Model 1
                system("c:\\progra~1\\mplus\\mplus.exe m1null.inp",show.output.on.console=F)       Run Mplus for data analysis
                if (file.exists('res.txt')){
                   m1.out<-scan('res.txt')                                                         Read saved estimates from Mplus
                   unlink('res.txt')
                }else{ m1.out<-NA}

                system("c:\\progra~1\\mplus\\mplus.exe m2alt.inp",show.output.on.console=F)
                if (file.exists('res.txt')){
                   m2.out<-scan('res.txt')
                   unlink('res.txt')
                   unlink('data.txt')
                }else{ m2.out<-NA}

                cat(c(i,m1.out,m2.out), file=res.file,append=T)
                cat("\n", file=res.file,append=T)
             }

          }
       }
    }
}
proc.time()-ptm
```