**Description:**

The following programs and data were utilized in the article "Marginal analysis of measurement agreement among multiple raters with non-ignorable missing" by Zhen Chen and Yunlong Xie on Statistics and Its Interface 2014. The programs were written by Y Xie. If you have any questions regarding the programs please email: Yunlong.xie@nih.gov.

The programs in this file are:• visualization.R, this R code generates the clean data for analysis:

• R\_function\_file.R, this is the R code contains all of the necessary R functions.• AnalysisProj.R, this is the R code to analysis the data.

**Abstract of the article:**

In diagnostic medicine, several measurements have been developed to evaluate the agreements among raters when the data are complete. In practice, raters may not be able to give definitive ratings to some participants because symptoms may not be clear-cut. Simply removing subjects with missing ratings may produce biased estimates and result in loss of efficiency. In this article, we propose a within-cluster resampling (WCR) procedure and a   
marginal approach to handle non-ignorable missing data in measurement agreement. Simulation studies show that both WCR and marginal approach provide unbiased estimates and have coverage probabilities close to the nominal level. The proposed methods are applied to data set from the Physician Reliability Study in diagnosing endometriosis.

**Important functions in R\_code:** R\_function\_file.R

FLSkp computes Fleiss Kappa of given data set this function works for binary data only;

sim generates simulation data;

resample, does resampling from data and compute kappa

resampleMN does resampling from data and compute kappa,

WGEE, computes estimates based on weighted GEE

EMkp, computes the empirical kappa

Delete1, is for missing completely at random (MCAR)

Delete2, is for non-ignorable 1st missing, depending on P(Y=1)

Delete3, is non-ignorable 2nd missing, depending on the variability of rating per subject

EMmissingKP, computes the empirical kappa

**USeages:**

FLSkp{B}, sim {p,r, N, p2\_1,p3\_1,p4\_1,p5\_1,p6\_1,p7\_1,p8\_1}, resample{A}, resampleMN{A, J},

WGEE{A}, EMkp{p,r, N, p2\_1,p3\_1,p4\_1,R,J}, Delete1{X,Pmiss}, Delete2{X,Pmiss}, Delete3{X,a,b},

EMmissingKP{p,r, N, p2\_1,p3\_1,p4\_1,p5\_1,p6\_1,p7\_1,p8\_1,R,J,Pmiss,a,b}

**ARGUMENTS:**

B the matrix containing binary ratings (0,1) with row standing for subjects and column standing for raters

A A=B+1, It change binary labelling from (0,1) to (1,2)

r the number of raters, default r=4

p the marginal probability

N the sample size

p2\_1 conditional probability P(Y2=1|Y1=1); p3\_1, p4\_1, and so on have the similar definition.

J the number of times of resampling

N the sample size

R the number of replicates of the simulated data generated in order to compute the true Fleiss Kappa

Pmiss the probability of assigning missing

X simulated data

a in non-ignorable 2nd missing situation, such as in function delete3{} and EMmissingKP{}, Pmiss=pnorm(a + b \*Var(Y)), a is location parameter

b in non-ignorable 2nd missing situation, such as in function delete3{} and EMmissingKP{}, Pmiss=pnorm(a + b \*Var(Y)), b is scale parameter

**Authors:**

Zhen Chen and Yunlong Xie

[chenzhe@mail.nih.gov](mailto:chenzhe@mail.nih.gov)

**Examples:**

1. **Example with simulated data:**

Function sim{}:

set.seed(1)

A1=sim(0.7,4, 10000,0.8,0.9)

A1$cor12

A1$EMcor12

A1$cor13

A1$EMcor13

A1$cor23

A1$EMcor23

colSums(A1$Data)/nrow(A1$Data)

**Function EMkp {}:**

completeKP=EMkp(0.7,4, 100,0.8,0.9,0.85,1000, 1000)

Function EMmissingKP{}:

p=as.double(commandArgs()[3])#marginal probability

r=as.double(commandArgs()[4])#number of raters

N=as.double(commandArgs()[5])#sample size

p2\_1=as.double(commandArgs()[6])#P(Y2=1|Y1=1)

p3\_1=as.double(commandArgs()[7])#P(Y3=1|Y1=1)

p4\_1=as.double(commandArgs()[8])#P(Y4=1|Y1=1)

p5\_1=as.double(commandArgs()[9])#P(Y5=1|Y1=1)

p6\_1=as.double(commandArgs()[10])#P(Y6=1|Y1=1)

p7\_1=as.double(commandArgs()[11])#P(Y7=1|Y1=1)

p8\_1=as.double(commandArgs()[12])#P(Y8=1|Y1=1)

R=as.double(commandArgs()[13])#the number of replicates of the simulated data generated

J=as.double(commandArgs()[14])#the number of times of resampling

Pmiss=as.double(commandArgs()[15])#probability of assigning missingness

b=as.double(commandArgs()[16])#parameter determining probability of assigning missingness

a=-4

set.seed(1)

SimResult=EMmissingKP(p,r, N, p2\_1,p3\_1,p4\_1,p5\_1,p6\_1,p7\_1,p8\_1,R,J,Pmiss,a,b)

FLS=round(sim(p,r, N, p2\_1,p3\_1,p4\_1,p5\_1,p6\_1,p7\_1,p8\_1)$FLS,2)

1. **Example with analysis data set:**

(see AnalysisProj.R )