# Supplementary material to "Efficient regression modeling for correlated and overdispersed count data"

XIAOMENG NIU AND HYUNKEUN RYAN CHO *

## R code

```
library(MASS)
library(corcounts)

##########################
#AR(1) correlation matrix#
##########################

ar1 <- function(rho,n) {
  z <- outer(1:n,1:n,'-')
  z <- rho^(abs(z))
  diag(z) <- 1
  z
}

##############################
#Function for generating data#
```

*Hyunkeun Ryan Cho is Assistant Professor, Department of Biostatistics, University of Iowa, Iowa City, IA 52242 (E-mail: hyunkeun-cho@uiowa.edu).

```
##############################

x0 <- rep(1,n)
x1 <- runif(n,min=0,max=2)
x2 <- runif(n,min=0,max=2)
x <- cbind(x0,x1,x2)
Xbeta <- exp(x%*%t(beta))
mumatrix <- matrix(Xbeta,ncol=clsz,byrow=T)

Y <- array(0,dim=c(ncl,clsz))
for (i in 1:ncl){
Y[i,] <- rcounts(N=500,margins=rep("NB",clsz),mu=mumatrix[i,],
          psi=rep(1/c,clsz),corstr="AR1",corpar=rho)[1,]
}

#ncl: number of subjects
#clsz: cluster size
#n: ncl*clsz
#beta: true parameter
#c: overdispersion parameter
#rho: correlation coefficient

##################################################
#Estimate parameter through our proposed approach#
#using QIF under negative binomial               #
##################################################

qif_nb <- function(y,x,id,beta.ini,c.ini,corstr){
  Y <- as.matrix(y)
```

```
X <- as.matrix(x)
ind <- id
ncl <- length(unique(id))
T <- length(ind)/ncl
p <- length(beta.ini)

if (corstr=="AR1") {
  m1 <- matrix(0,T,T)
  m1[abs(row(m1) - col(m1)) == 1] <- 1
}
else if (corstr=="ex"){
  m1 <- matrix(1,T,T)
   diag(m1) <- 0
}
m2 <- matrix(1,2,2)
diag(m2) <- 0
M1 <- diag(T) %x% diag(2)
M2 <- diag(T) %x% m2

betadiff <- 1
cdiff <- 1
diff <- 1
iteration <- 0
betanew <- as.matrix(beta.ini)
cnew <- c.ini
tol<-1e-8
maxiter<-100
while (diff > tol && iteration < maxiter) {
   beta <-betanew
```

```
c <- cnew
if (corstr=="AR1" || corstr=="ex"){
  gi <- sumg <- array(0, dim = c(4*(p+1), 1))
  sumc <- array(0, dim = c(4*(p+1), 4*(p+1)))
  gifirstdev <- sumgfirstdev <- array(0, dim = c(4*(p+1), p+1))
}
else if (corstr=="indep"){
  gi <- sumg <- array(0, dim = c(2*(p+1), 1))
  sumc <- array(0, dim = c(2*(p+1), 2*(p+1)))
  gifirstdev <- sumgfirstdev <- array(0, dim = c(2*(p+1), p+1))
}

for (i in 1 : ncl) {
  yi <- as.matrix(Y[ind==i])
  yi2 <- yi^2
  fi <- c(t(cbind(yi,yi2)))
  xi <- as.matrix(X[ind==i,])
  ui <- exp(xi %*% beta)
  mi <- ui+(c+1)*ui^2
  mui <- c(t(cbind(ui,mi)))
  rui <- matrix(ui,T,p)
  devuib <- rui*xi
  devmib <- rui*xi+2*(c+1)*rui^2*xi
  devmic <- ui^2
  di <- matrix(c(t(cbind(devuib,as.matrix(rep(0,T)),
        devmib,devmic))),nrow=2*T,byrow=T)
  vari1 <- ui+c*ui^2
  vari2 <- ui+(6+7*c)*ui^2+(4+16*c+12*c^2)*ui^3+
           (4*c+10*c^2+6*c^3)*ui^4
```

```
covi <- diag(c(t(cbind(vari1, vari2))))
covit <- ginv(sqrt(covi))

if (corstr=="AR1" || corstr=="ex"){
  M3 <- m1 %x% diag(2)
  M4 <- m1 %x% m2
  gi[1:(p+1),] <- (1/ncl) * t(di) %*% covit %*%
                    M1 %*% covit %*% (fi - mui)
  gi[(p+2):(2*p+2),] <- (1/ncl) * t(di) %*% covit %*%
                    M2 %*% covit %*% (fi - mui)
  gi[(2*p+3):(3*p+3),] <- (1/ncl) * t(di) %*% covit %*%
                    M3 %*% covit %*% (fi - mui)
  gi[(3*p+4):(4*p+4),] <- (1/ncl) * t(di) %*% covit %*%
                    M4 %*% covit %*% (fi - mui)
  sumg <- sumg + gi
  sumc <- sumc + gi %*% t(gi)
  gifirstdev[1:(p+1),] <- -(1/ncl) * t(di) %*% covit
                    %*% M1 %*% covit %*% di
  gifirstdev[(p+2):(2*p+2),] <- -(1/ncl) * t(di) %*% covit
                    %*% M2 %*% covit %*% di
  gifirstdev[(2*p+3):(3*p+3),] <- -(1/ncl) * t(di) %*% covit
                    %*% M3 %*% covit %*% di
  gifirstdev[(3*p+4):(4*p+4),] <- -(1/ncl) * t(di) %*% covit
                    %*% M4 %*% covit %*% di
  sumgfirstdev <- sumgfirstdev + gifirstdev
  }
else if (corstr=="indep"){
  gi[1:(p+1),] <- (1/ncl) * t(di) %*% covit %*%
                    M1 %*% covit %*% (fi - mui)
```

```
        gi[(p+2):(2*p+2),] <- (1/ncl) * t(di) %*% covit %*%
                                M2 %*% covit %*% (fi - mui)
        sumg <- sumg + gi
        sumc <- sumc + gi %*% t(gi)
        gifirstdev[1:(p+1),]  <- -(1/ncl) * t(di) %*% covit
                                  %*% M1 %*% covit %*% di
        gifirstdev[(p+2):(2*p+2),]  <- -(1/ncl) * t(di) %*% covit
                                        %*% M2 %*% covit %*% di
        sumgfirstdev <- sumgfirstdev + gifirstdev
        }
    }
    Q <- t(sumg) %*% ginv(sumc) %*% sumg
    qif1dev <- t(sumgfirstdev) %*% ginv(sumc) %*% sumg
    invqif2dev <- ginv(t(sumgfirstdev) %*% ginv(sumc)
                       %*% sumgfirstdev)
    betanew <- beta - (invqif2dev %*% qif1dev)[1:p]
    cnew <- c - (invqif2dev %*% qif1dev)[p+1]
    betadiff <- abs(sum(betanew - beta))
    cdiff <- abs(sum(cnew - c))
    diff <- betadiff+cdiff
    iteration <- iteration + 1
  }
  se <- sqrt(diag(invqif2dev))
  para <- c(beta,c)
  return(list(para=para,se=se,Q=Q))
}

###############################################
#Estimate parameter through QIF under Poisson#
```

```
#################################################

qif_pois <- function(y,x,id,beta.ini,corstr){
  Y <- as.matrix(y)
  X <- as.matrix(x)
  ind <- id
  ncl <- length(unique(id))
  T <- length(ind)/ncl
  p <- length(beta.ini)

  M1 <- diag(T)
  if (corstr=="AR1"){
    M2 <- matrix(0,T,T)
    M2[abs(row(M2) - col(M2)) == 1] <- 1
  }
  else if (corstr=="ex") {
    M2 <- matrix(1,T,T)
    diag(M2) <- 0
  }

  betadiff <- 1
  iteration <- 0
  betanew <- as.matrix(beta.ini)
  tol<-1e-8
  maxiter<-100
  while (betadiff > tol && iteration < maxiter) {
    beta <- betanew
    if (corstr=="AR1" || corstr=="ex"){
      gi <- sumg <- array(0, dim = c(2*p, 1))
```

```
  sumc <- array(0, dim = c(2*p, 2*p))
  gifirstdev <- sumgfirstdev <- array(0, dim = c(2*p, p))
}
else if (corstr=="indep"){
  gi <- sumg <- array(0, dim = c(p, 1))
  sumc <- array(0, dim = c(p, p))
  gifirstdev <- sumgfirstdev <- array(0, dim = c(p, p))
}

for (i in 1 : ncl) {
  yi <- as.matrix(Y[ind==i])
  xi <- as.matrix(X[ind==i,])
  ui <- exp(xi %*% beta)
  rui <- matrix(ui,T,p)
  di <- rui*xi
  vari1 <- ui
  varm <- diag(as.vector(vari1),T,T)
  covit <- ginv(sqrt(varm))

  if (corstr=="AR1" || corstr=="ex"){
    gi[1:p,] <- (1/ncl) * t(di) %*% covit %*% M1
                %*% covit %*% (yi - ui)
    gi[(p+1):(2*p),] <- (1/ncl) * t(di) %*% covit %*% M2
                        %*% covit %*% (yi - ui)
    sumg <- sumg + gi
    sumc <- sumc + gi %*% t(gi)
    gifirstdev[1:p,]  <- -(1/ncl) * t(di) %*% covit %*%
                        M1 %*% covit %*% di
    gifirstdev[(p+1):(2*p),]  <- -(1/ncl) * t(di) %*% covit
```

```
                                        %*% M2 %*% covit %*% di
      sumgfirstdev <- sumgfirstdev + gifirstdev
    }
    else if (corstr=="indep"){
      gi[1:p,] <- (1/ncl) * t(di) %*% covit %*% M1
                   %*% covit %*% (yi - ui)
      sumg <- sumg + gi
      sumc <- sumc + gi %*% t(gi)
      gifirstdev[1:p,]  <- -(1/ncl) * t(di) %*% covit %*%
                             M1 %*% covit %*% di
      sumgfirstdev <- sumgfirstdev + gifirstdev
    }
  }
  Q <- t(sumg) %*% ginv(sumc) %*% sumg
  qif1dev <- t(sumgfirstdev) %*% ginv(sumc) %*% sumg
  invqif2dev <- ginv(t(sumgfirstdev) %*% ginv(sumc)
                 %*% sumgfirstdev)
  betanew <- beta - (invqif2dev %*% qif1dev)
  betadiff <- abs(sum(betanew - beta))
  iteration <- iteration + 1
 }
 se <- sqrt(diag(invqif2dev))
 return(list(beta=beta,se=se))
}

##################################################
#Obtain the test statistic under null hypothesis#
#through our proposed method                     #
##################################################
```

```
qif_fixed <- function(y,x,id,betanew,c=0,corstr){
  Y <- as.matrix(y)
  X <- as.matrix(x)
  ind <- id
  ncl <- length(unique(id))
  T <- length(ind)/ncl
  p <- length(betanew)

  if (corstr=="AR1") {
    m1 <- matrix(0,T,T)
    m1[abs(row(m1) - col(m1)) == 1] <- 1
  }
  else if (corstr=="ex"){
    m1 <- matrix(1,T,T)
    diag(m1) <- 0
  }
  m2 <- matrix(1,2,2)
  diag(m2) <- 0
  M1 <- diag(T) %x% diag(2)
  M2 <- diag(T) %x% m2

  if (corstr=="AR1" || corstr=="ex"){
    gi <- sumg <- array(0, dim = c(4*(p+1), 1))
    sumc <- array(0, dim = c(4*(p+1), 4*(p+1)))
    gifirstdev <- sumgfirstdev <- array(0, dim = c(4*(p+1), p+1))
  }
  else if (corstr=="indep"){
    gi <- sumg <- array(0, dim = c(2*(p+1), 1))
```

```
  sumc <- array(0, dim = c(2*(p+1), 2*(p+1)))
  gifirstdev <- sumgfirstdev <- array(0, dim = c(2*(p+1), p+1))
}

for (i in 1 : ncl) {
  yi <- as.matrix(Y[ind==i])
  yi2 <- yi^2
  fi <- c(t(cbind(yi,yi2)))
  xi <- as.matrix(X[ind==i,])
  ui <- exp(xi %*% betanew)
  mi <- ui+(c+1)*ui^2
  mui <- c(t(cbind(ui,mi)))
  rui <- matrix(ui,T,p)
  devuib <- rui*xi
  devmib <- rui*xi+2*(c+1)*rui^2*xi
  devmic <- ui^2
  di <- matrix(c(t(cbind(devuib,as.matrix(rep(0,T)),
        devmib,devmic))),nrow=2*T,byrow=T)
  vari1 <- ui+c*ui^2
  vari2 <- ui+(6+7*c)*ui^2+(4+16*c+12*c^2)*ui^3+
           (4*c+10*c^2+6*c^3)*ui^4
  covi <- diag(c(t(cbind(vari1,vari2))))
  covit <- ginv(sqrt(covi))

  if (corstr=="AR1" || corstr=="ex"){
    M3 <- m1 %x% diag(2)
    M4 <- m1 %x% m2
    gi[1:(p+1),] <- (1/ncl) * t(di) %*% covit %*% M1
                     %*% covit %*% (fi - mui)
```

```
    gi[(p+2):(2*p+2),] <- (1/ncl) * t(di) %*% covit %*% M2
                             %*% covit %*% (fi - mui)
    gi[(2*p+3):(3*p+3),] <- (1/ncl) * t(di) %*% covit %*% M3
                             %*% covit %*% (fi - mui)
    gi[(3*p+4):(4*p+4),] <- (1/ncl) * t(di) %*% covit %*% M4
                             %*% covit %*% (fi - mui)
    sumg <- sumg + gi
    sumc <- sumc + gi %*% t(gi)
    gifirstdev[1:(p+1),]  <- -(1/ncl) * t(di) %*% covit %*%
                             M1 %*% covit %*% di
    gifirstdev[(p+2):(2*p+2),]  <- -(1/ncl) * t(di) %*% covit
                                   %*% M2 %*% covit %*% di
    gifirstdev[(2*p+3):(3*p+3),]  <- -(1/ncl) * t(di) %*% covit
                                   %*% M3 %*% covit %*% di
    gifirstdev[(3*p+4):(4*p+4),]  <- -(1/ncl) * t(di) %*% covit
                                   %*% M4 %*% covit %*% di
    sumgfirstdev <- sumgfirstdev + gifirstdev
  }
  else if (corstr=="indep"){
    gi[1:(p+1),] <- (1/ncl) * t(di) %*% covit %*% M1 %*%
                    covit %*% (fi - mui)
    gi[(p+2):(2*p+2),] <- (1/ncl) * t(di) %*% covit %*% M2 %*%
                          covit %*% (fi - mui)
    sumg <- sumg + gi
    sumc <- sumc + gi %*% t(gi)
    gifirstdev[1:(p+1),]  <- -(1/ncl) * t(di) %*% covit %*%
                             M1 %*% covit %*% di
    gifirstdev[(p+2):(2*p+2),]  <- -(1/ncl) * t(di) %*% covit
                                   %*% M2 %*% covit %*% di
```

```
        sumgfirstdev <- sumgfirstdev + gifirstdev
      }
    }
    Q <- t(sumg) %*% ginv(sumc) %*% sumg
    return(Q)
}
```

# Results of additional simulation studies

Table 1: Absolute bias, Mean squared errors $(\times 1000)$, average length of confidence intervals (LCI), and coverage proportion (CP) at $95\%$ confidence level using the negative binomial regression under the AR(1), compound symmetry (CS), and independent (IN) working correlation structure.

| | | $\gamma = 0.5$ | | | | $\gamma = 0.1$ | | | | $\gamma = 0$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BIAS | MSE | LCI | CP | BIAS | MSE | LCI | CP | BIAS | MSE | LCI | CP |
| $n = 100$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.027 | 1.205 | 0.120 | 0.904 | 0.015 | 0.380 | 0.066 | 0.907 | 0.008 | 0.118 | 0.038 | 0.917 |
| | $\beta_2$ | 0.028 | 1.327 | 0.119 | 0.894 | 0.014 | 0.322 | 0.064 | 0.918 | 0.006 | 0.069 | 0.030 | 0.932 |
| | $\gamma$ | 0.050 | 3.865 | 0.199 | 0.881 | 0.010 | 0.184 | 0.047 | 0.913 | 0.001 | 0.003 | 0.006 | 0.912 |
| CS | $\beta_1$ | 0.031 | 1.538 | 0.139 | 0.915 | 0.016 | 0.446 | 0.074 | 0.923 | 0.009 | 0.141 | 0.042 | 0.914 |
| | $\beta_2$ | 0.031 | 1.629 | 0.139 | 0.910 | 0.016 | 0.420 | 0.073 | 0.924 | 0.007 | 0.082 | 0.034 | 0.946 |
| | $\gamma$ | 0.050 | 3.994 | 0.203 | 0.887 | 0.011 | 0.191 | 0.049 | 0.911 | 0.001 | 0.003 | 0.006 | 0.899 |
| IN | $\beta_1$ | 0.047 | 3.658 | 0.211 | 0.904 | 0.024 | 0.897 | 0.110 | 0.938 | 0.011 | 0.223 | 0.057 | 0.946 |
| | $\beta_2$ | 0.048 | 3.717 | 0.210 | 0.913 | 0.024 | 0.917 | 0.107 | 0.925 | 0.009 | 0.155 | 0.050 | 0.961 |
| | $\gamma$ | 0.052 | 4.340 | 0.225 | 0.904 | 0.011 | 0.196 | 0.050 | 0.919 | 0.002 | 0.006 | 0.009 | 0.925 |
| $n = 300$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.014 | 0.344 | 0.072 | 0.948 | 0.008 | 0.101 | 0.039 | 0.944 | 0.004 | 0.035 | 0.023 | 0.945 |
| | $\beta_2$ | 0.015 | 0.369 | 0.071 | 0.941 | 0.007 | 0.100 | 0.038 | 0.942 | 0.003 | 0.021 | 0.018 | 0.942 |
| | $\gamma$ | 0.030 | 1.430 | 0.130 | 0.918 | 0.006 | 0.064 | 0.030 | 0.940 | 0.001 | 0.001 | 0.004 | 0.958 |
| CS | $\beta_1$ | 0.017 | 0.458 | 0.084 | 0.957 | 0.009 | 0.127 | 0.044 | 0.956 | 0.005 | 0.041 | 0.025 | 0.943 |
| | $\beta_2$ | 0.017 | 0.491 | 0.083 | 0.929 | 0.009 | 0.131 | 0.043 | 0.947 | 0.004 | 0.027 | 0.020 | 0.956 |
| | $\gamma$ | 0.030 | 1.479 | 0.133 | 0.916 | 0.006 | 0.061 | 0.030 | 0.947 | 0.001 | 0.001 | 0.004 | 0.957 |
| IN | $\beta_1$ | 0.026 | 1.089 | 0.128 | 0.957 | 0.013 | 0.278 | 0.066 | 0.950 | 0.006 | 0.071 | 0.033 | 0.952 |
| | $\beta_2$ | 0.026 | 1.134 | 0.127 | 0.944 | 0.013 | 0.298 | 0.064 | 0.942 | 0.006 | 0.055 | 0.029 | 0.952 |
| | $\gamma$ | 0.031 | 1.563 | 0.142 | 0.919 | 0.006 | 0.066 | 0.031 | 0.949 | 0.001 | 0.002 | 0.005 | 0.959 |
| $n = 1000$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.008 | 0.106 | 0.040 | 0.945 | 0.004 | 0.032 | 0.022 | 0.945 | 0.002 | 0.010 | 0.012 | 0.953 |
| | $\beta_2$ | 0.007 | 0.097 | 0.040 | 0.958 | 0.004 | 0.029 | 0.021 | 0.943 | 0.002 | 0.006 | 0.010 | 0.955 |
| | $\gamma$ | 0.017 | 0.461 | 0.078 | 0.926 | 0.003 | 0.020 | 0.017 | 0.947 | <0.001 | <0.001 | 0.002 | 0.954 |
| CS | $\beta_1$ | 0.009 | 0.137 | 0.046 | 0.946 | 0.005 | 0.042 | 0.024 | 0.945 | 0.002 | 0.012 | 0.013 | 0.945 |
| | $\beta_2$ | 0.009 | 0.130 | 0.046 | 0.959 | 0.004 | 0.037 | 0.024 | 0.950 | 0.002 | 0.007 | 0.011 | 0.955 |
| | $\gamma$ | 0.017 | 0.474 | 0.080 | 0.938 | 0.003 | 0.020 | 0.017 | 0.947 | <0.001 | <0.001 | 0.002 | 0.958 |
| IN | $\beta_1$ | 0.014 | 0.344 | 0.071 | 0.943 | 0.007 | 0.087 | 0.036 | 0.946 | 0.003 | 0.022 | 0.018 | 0.953 |
| | $\beta_2$ | 0.014 | 0.318 | 0.071 | 0.961 | 0.006 | 0.078 | 0.036 | 0.952 | 0.003 | 0.016 | 0.016 | 0.953 |
| | $\gamma$ | 0.017 | 0.477 | 0.083 | 0.944 | 0.003 | 0.020 | 0.017 | 0.951 | <0.001 | <0.001 | 0.003 | 0.949 |

Table 2: Absolute bias, Mean squared errors $(\times 1000)$, average length of confidence intervals (LCI), and coverage proportion (CP) at $95\%$ confidence level using the Poisson regression under the AR(1), compound symmetry (CS), and independent (IN) working correlation structure.

| | | $\gamma = 0.5$ | | | | $\gamma = 0.1$ | | | | $\gamma = 0$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BIAS | MSE | LCI | CP | BIAS | MSE | LCI | CP | BIAS | MSE | LCI | CP |
| $n = 100$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.034 | 1.901 | 0.164 | 0.933 | 0.017 | 0.484 | 0.080 | 0.927 | 0.008 | 0.112 | 0.039 | 0.928 |
| | $\beta_2$ | 0.034 | 1.892 | 0.161 | 0.932 | 0.016 | 0.433 | 0.077 | 0.932 | 0.006 | 0.064 | 0.030 | 0.948 |
| CS | $\beta_1$ | 0.033 | 1.747 | 0.165 | 0.944 | 0.017 | 0.495 | 0.082 | 0.936 | 0.009 | 0.134 | 0.042 | 0.934 |
| | $\beta_2$ | 0.035 | 1.963 | 0.168 | 0.938 | 0.017 | 0.485 | 0.082 | 0.936 | 0.007 | 0.077 | 0.034 | 0.955 |
| IN | $\beta_1$ | 0.054 | 4.654 | 0.259 | 0.944 | 0.025 | 1.028 | 0.126 | 0.947 | 0.011 | 0.222 | 0.058 | 0.945 |
| | $\beta_2$ | 0.054 | 4.541 | 0.256 | 0.954 | 0.025 | 1.042 | 0.124 | 0.946 | 0.009 | 0.151 | 0.050 | 0.967 |
| $n = 300$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.019 | 0.581 | 0.098 | 0.958 | 0.009 | 0.137 | 0.047 | 0.957 | 0.004 | 0.035 | 0.023 | 0.950 |
| | $\beta_2$ | 0.019 | 0.615 | 0.095 | 0.948 | 0.009 | 0.139 | 0.045 | 0.941 | 0.003 | 0.021 | 0.018 | 0.941 |
| CS | $\beta_1$ | 0.018 | 0.542 | 0.097 | 0.960 | 0.009 | 0.148 | 0.048 | 0.959 | 0.005 | 0.040 | 0.025 | 0.945 |
| | $\beta_2$ | 0.020 | 0.636 | 0.099 | 0.950 | 0.009 | 0.154 | 0.048 | 0.949 | 0.004 | 0.026 | 0.020 | 0.959 |
| IN | $\beta_1$ | 0.029 | 1.366 | 0.151 | 0.964 | 0.014 | 0.327 | 0.074 | 0.955 | 0.006 | 0.071 | 0.033 | 0.956 |
| | $\beta_2$ | 0.030 | 1.479 | 0.151 | 0.952 | 0.015 | 0.368 | 0.073 | 0.947 | 0.005 | 0.055 | 0.029 | 0.954 |
| $n = 1000$ | | | | | | | | | | | | | |
| AR(1) | $\beta_1$ | 0.010 | 0.181 | 0.054 | 0.952 | 0.005 | 0.044 | 0.026 | 0.954 | 0.002 | 0.010 | 0.012 | 0.952 |
| | $\beta_2$ | 0.010 | 0.174 | 0.053 | 0.957 | 0.005 | 0.042 | 0.025 | 0.952 | 0.002 | 0.006 | 0.010 | 0.962 |
| CS | $\beta_1$ | 0.010 | 0.179 | 0.053 | 0.949 | 0.005 | 0.049 | 0.026 | 0.943 | 0.002 | 0.012 | 0.013 | 0.949 |
| | $\beta_2$ | 0.011 | 0.186 | 0.055 | 0.954 | 0.005 | 0.045 | 0.026 | 0.951 | 0.002 | 0.007 | 0.011 | 0.956 |
| IN | $\beta_1$ | 0.016 | 0.439 | 0.083 | 0.954 | 0.008 | 0.106 | 0.040 | 0.955 | 0.003 | 0.022 | 0.018 | 0.951 |
| | $\beta_2$ | 0.016 | 0.436 | 0.083 | 0.953 | 0.007 | 0.101 | 0.040 | 0.951 | 0.003 | 0.015 | 0.016 | 0.958 |

Table 3: Relative efficiencies under the AR(1), compound symmetry (CS), and independent (IN) working correlation structure.

| | $\gamma = 0.5$ | | | $\gamma = 0.1$ | | | $\gamma = 0$ | | |
|---|---|---|---|---|---|---|---|---|---|
| | AR(1) | CS | IN | AR(1) | CS | IN | AR(1) | CS | IN |
| $n = 100$ | | | | | | | | | |
| $\beta_1$ | 1.577 | 1.136 | 1.272 | 1.270 | 1.109 | 1.146 | 0.948 | 0.953 | 0.992 |
| $\beta_2$ | 1.426 | 1.204 | 1.221 | 1.344 | 1.153 | 1.137 | 0.924 | 0.938 | 0.969 |
| $n = 300$ | | | | | | | | | |
| $\beta_1$ | 1.689 | 1.183 | 1.255 | 1.356 | 1.165 | 1.176 | 0.980 | 0.981 | 0.989 |
| $\beta_2$ | 1.664 | 1.295 | 1.304 | 1.390 | 1.176 | 1.235 | 0.990 | 0.981 | 0.985 |
| $n = 1000$ | | | | | | | | | |
| $\beta_1$ | 1.708 | 1.308 | 1.274 | 1.351 | 1.168 | 1.217 | 0.990 | 0.995 | 0.998 |
| $\beta_2$ | 1.785 | 1.427 | 1.369 | 1.419 | 1.223 | 1.300 | 0.981 | 0.982 | 0.983 |

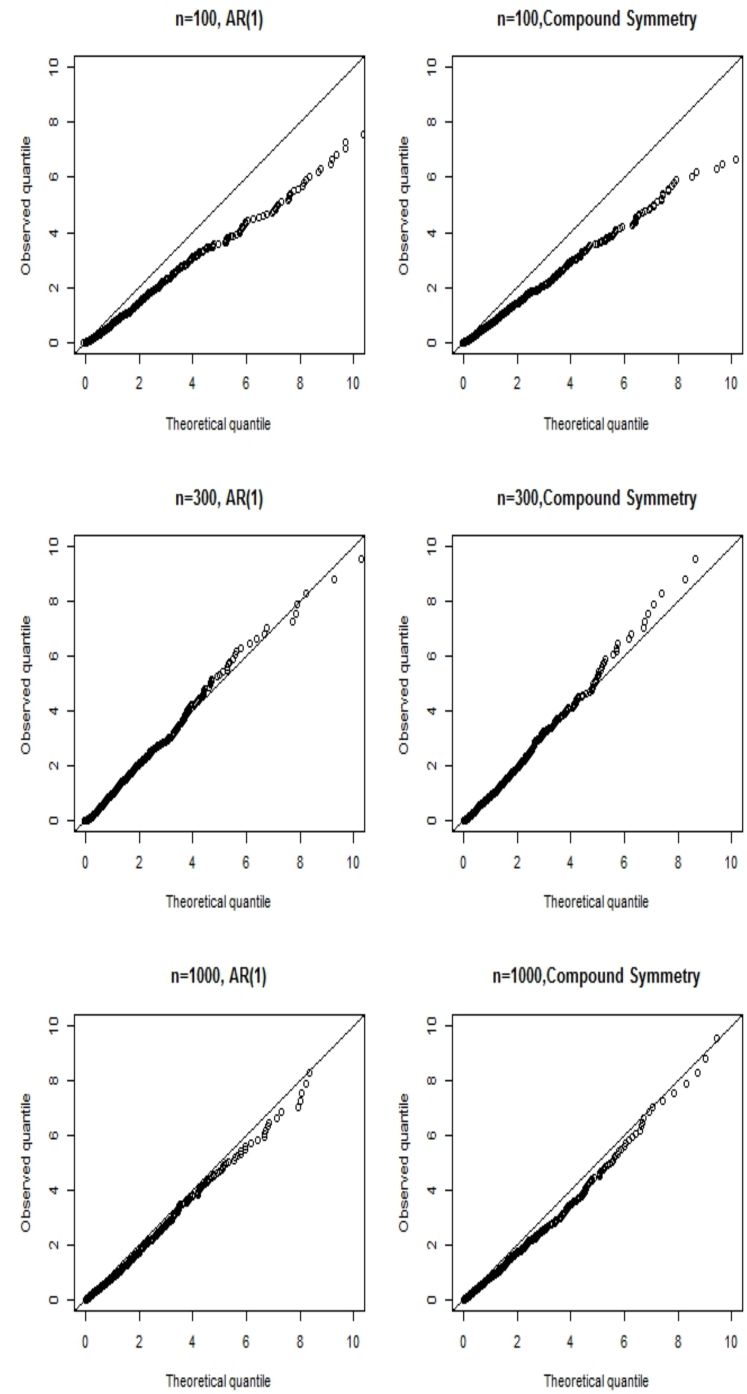


Figure 1: Quantile-Quantile plots under the AR(1) and compound symmetry working correlation structure without overdispersion.

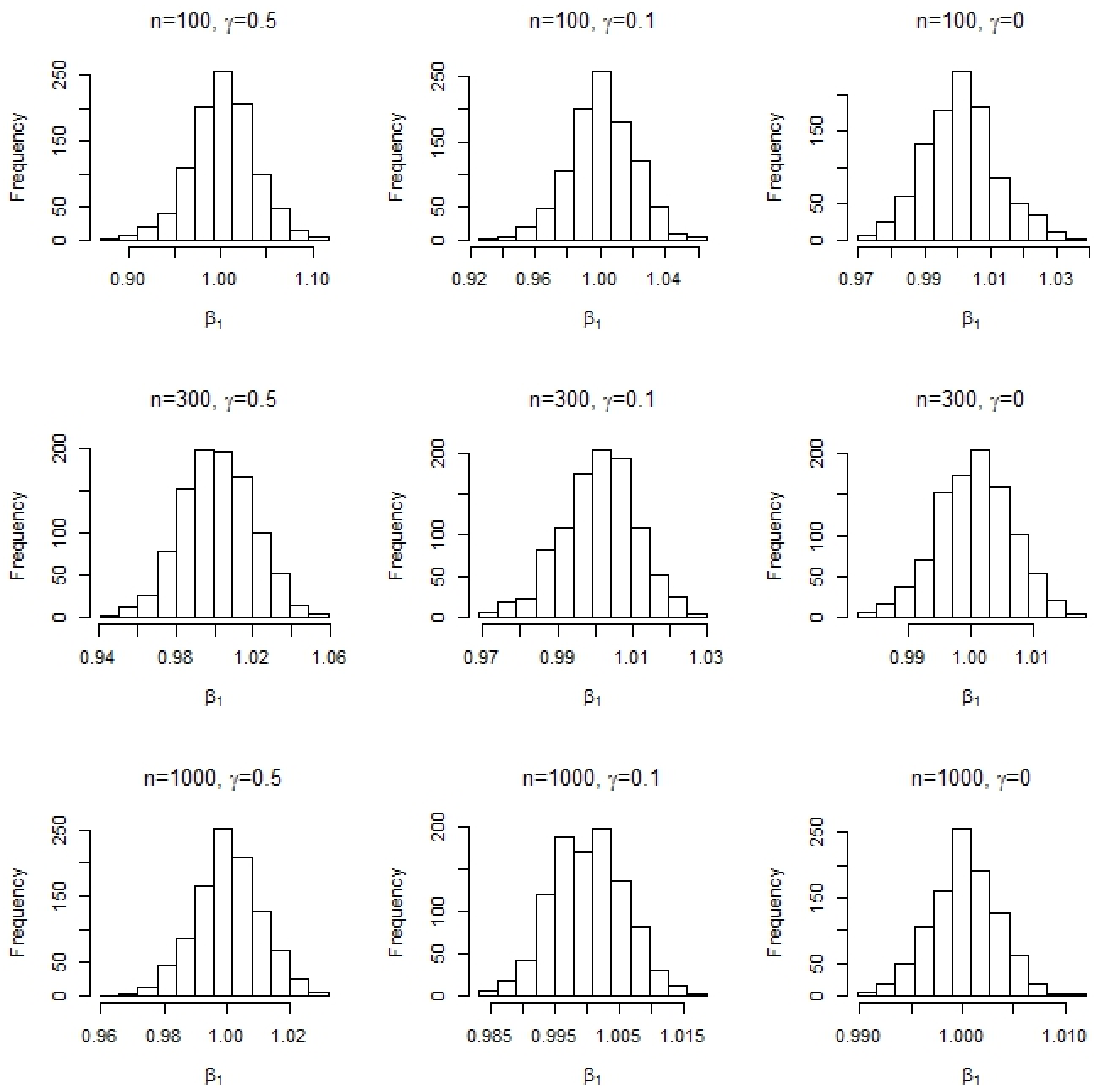


Figure 2: Histograms of 1000 estimators of $\beta_1$ under the AR(1) working correlation structure.

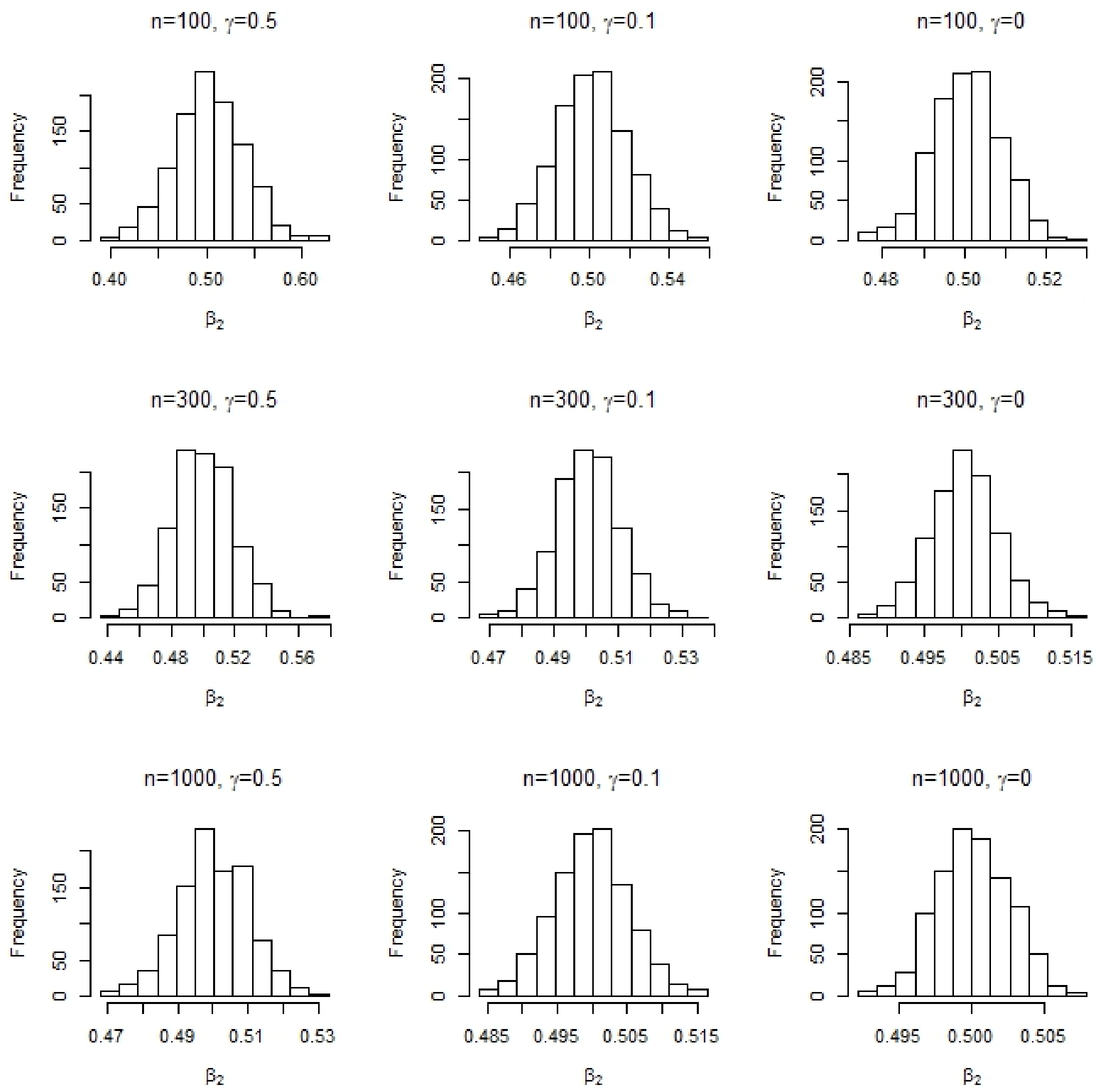


Figure 3: Histograms of 1000 estimators of $\beta_2$ under the AR(1) working correlation structure.

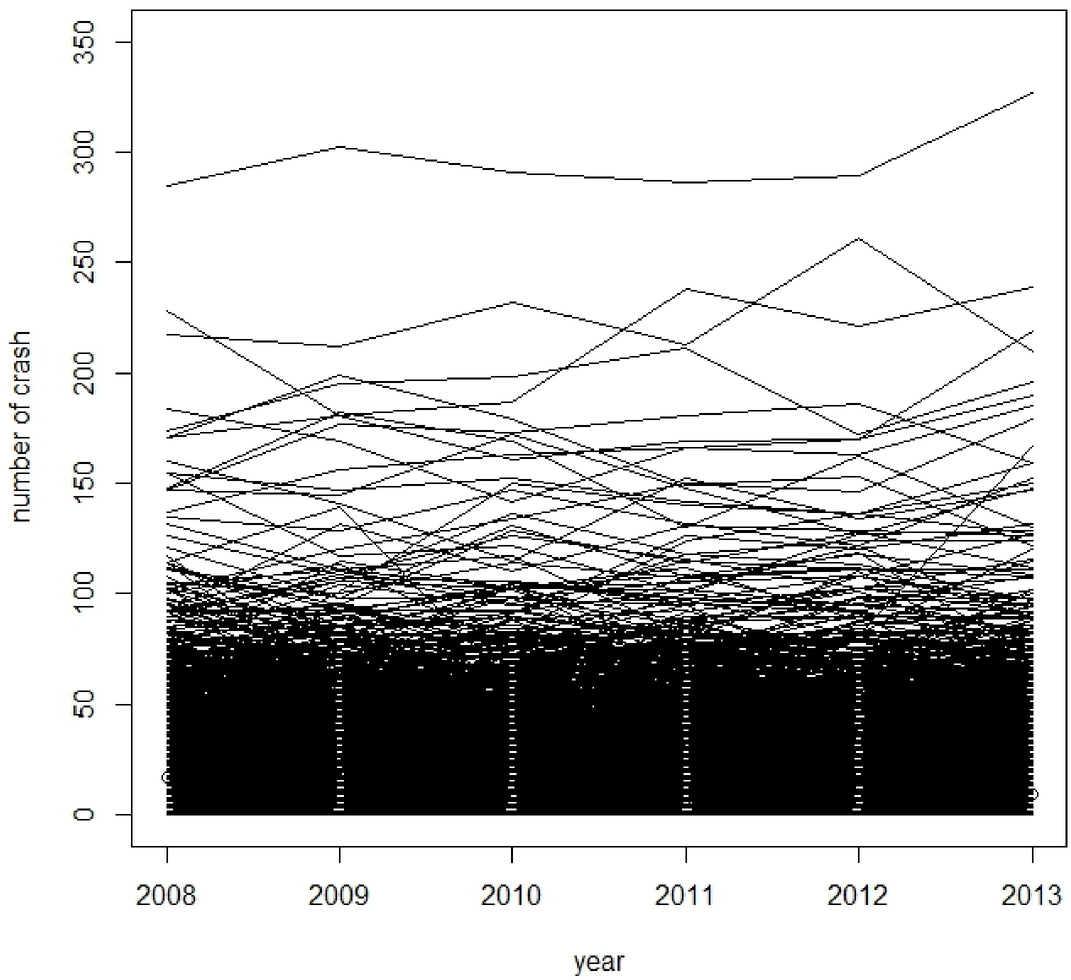


Figure 4: Individual segment trajectories of crashes in 2008 - 2013.