โปรแกรม R

```
##################################################################################
                         Function of  Beta Exponential  distribution
##################################################################################
#########################      Probability mass  function      #########################
dnbgl<-function(y,r,theta,alpha,gamma){
k<-numeric()
nbs <- function(y,r,theta,alpha,gamma)
        {

                if (y==0){
                p<-factorial(r+y-1)/(factorial(r-1)*(theta/(theta+r))^(alpha+1)
                        *(theta+r+gamma)/(theta+gamma)

                }
                else if (y>0)
                {
                pp1<-
(theta/(theta+r))^(alpha+1)*(theta+r+gamma)/(theta+gamma)
                for(j in 1:y)
                {
                        p1<- (factorial(y)/(factorial(y-j)*factorial(j)))*((-1)^j)*
(theta/(theta+r+j))^(alpha+1)*(theta+r+j+gamma)/(theta+gamma)
                        pp1<-pp1+p1
                }
                        p<-(factorial(r+y-1))/(factorial(r-1)*factorial(y))*pp1
                }
        p
        }

                for(i in 1 : length(y))
                {
                        k[i]<-nbs(y[i],r,theta,alpha,gamma)
                }
return(k)
}
#########################      Plot of Probability mass  function

#########################
x<-seq(0,20,1)
r=5
theta=3
```

```
alpha=2
gamma=3
pmf<-dnbgl(x,r,theta,alpha,gamma)
sum(pmf)
plot(x,pmf,type="h",ylim=c(0,max(pmf)+0.05),main="")
points(x,pmf, cex =1.5,pch=20, type="p",col = "black")
pmf


###############     Negative Binomial - Generalized Lindley random variables

###################
rnbgl<-function(n,r,theta,alpha,gamma)
{
   rglindley<-function(n,theta,alpha,gamma)
  {
  x<-rep(0,n)
  for(i in 1:n){
    u<-runif(1)
    v1<-rgamma(1,shape=alpha,rate = theta)
    v2<-rgamma(1,shape=alpha+1,rate = theta)
    w<-theta/(theta+gamma)
    if(u<=w){x[i]=v1} else
      if(u>w){x[i]=v2}
    sample<-x
            }
sample
  }
lambda<-rglindley(n,theta,alpha,gamma)
y<-rnbinom(n,size=r,prob=exp(-lambda))
sample<-y
}

x<-rnbgl(n,True_r,True_theta,True_alpha,True_gamma)


##############################################################################

        Estimate Parameter for Negative Binomial - Generalized Lindley random variables

##############################################################################

##########################    Maximum likelihood Estimation  #######################
```

```
mlogl<-function(t,y){
fnbgl<-function(t,y){
k<-numeric()
nbgl <- function(t,y)
        {
                if (y==0){
                p<--log(factorial(t[1]+y-1))+log(factorial(t[1]-1))-
log((t[3]/(t[3]+t[1]))^(t[2]+1))-log((t[3]+t[1]+t[4])/(t[3]+t[4]))
                }
                else if (y>0)
                {
                pp1<- (t[3]/(t[3]+t[1]))^(t[2]+1)*(t[3]+t[1]+t[4])/(t[3]+t[4])
                for(j in 1:y)
                {
                        p1<- (factorial(y)/(factorial(y-j)*factorial(j)))*((-1)^j)*
(t[3]/(t[3]+t[1]+j))^(t[2]+1)*(t[3]+t[1]+j+t[4])/(t[3]+t[4])
                        pp1<-pp1+p1
                }
                        p<--log(factorial(t[1]+y-1))+log(factorial(t[1]-1))-
log(factorial(y))-log(pp1)
                }
        p
        }
                for(i in 1 : length(y))
                {
                        k[i]<-nbgl(t,y[i])
                }
return(k)
}
sum(fnbgl(t,y))
}
t.start <- c(2,2,2,2)
out put.mle<- nlm(mlogl, t.start, y = x)
###############################      Method of Moment

###############################

library(gmm)      ## Using gmm package
theta.start <- c(start1,start2,start3,start4)
generalized_moment<-function(theta,x){
para1<-theta[1]
para2<-theta[2]
para3<-theta[3]
para4<-theta[4]

m1<-x-( para1*(((para2/(para2-1))^(para3+1)*(para2-1+para4)/(para2+para4))-
1))
```

```
m2<-(x^2)-((para1*(para1+1)*((para2)^(para3+1))/(para2+para4))* ( ((para2-
2+para4)/((para2-2)^(para3+1)))-(2*(para2-1+para4)/((para2-1)^(para3+1)))
+((para2+para4)/((para2)^(para3+1)))))
m3<-(x^3)-((para1*(para1+1)*(para1+2)*((para2)^(para3+1))/(para2+para4))*(
((para2-3+para4)/((para2-3)^(para3+1))-(3*(para2-2+para4)/((para2-
2)^(para3+1)))+(3*(para2-1+para4)/((para2-1)^(para3+1))) -
((para2+para4)/((para2)^(para3+1))))))
m4<-(x^4)-
((para1*(para1+1)*(para1+2)*(para1+3)*((para2)^(para3+1))/(para2+para4))*((
(para2-4+para4)/((para2-4)^(para3+1)))-(4*(para2-3+para4)/((para2-
3)^(para3+1)))+(6*(para2-2+para4)/((para2-2)^(para3+1))) -(4*(para2-
1+para4)/((para2-1)^(para3+1)))+ ((para2+para4)/((para2)^(para3+1)))))
f<-cbind(m1,m2,m3,m4)
}
Dg<-function(theta,x){
para1<-theta[1]
para2<-theta[2]
para3<-theta[3]
para4<-theta[4]
m1.exp <- expression(x-( para1*(((para2/(para2-1))^(para3+1)*(para2-
1+para4)/(para2+para4))-1)))
m2.exp <- expression((x^2)-
((para1*(para1+1)*((para2)^(para3+1))/(para2+para4))* ( ((para2-
2+para4)/((para2-2)^(para3+1)))-(2*(para2-1+para4)/((para2-1)^(para3+1)))
+((para2+para4)/((para2)^(para3+1))))))
m3.exp <- expression((x^3)-
((para1*(para1+1)*(para1+2)*((para2)^(para3+1))/(para2+para4))*( ((para2-
3+para4)/((para2-3)^(para3+1))-(3*(para2-2+para4)/((para2-
2)^(para3+1)))+(3*(para2-1+para4)/((para2-1)^(para3+1))) -
((para2+para4)/((para2)^(para3+1)))))))
m4.exp <- expression((x^4)-
((para1*(para1+1)*(para1+2)*(para1+3)*((para2)^(para3+1))/(para2+para4))*((
(para2-4+para4)/((para2-4)^(para3+1)))-(4*(para2-3+para4)/((para2-
3)^(para3+1)))+(6*(para2-2+para4)/((para2-2)^(para3+1))) -(4*(para2-
1+para4)/((para2-1)^(para3+1)))+ ((para2+para4)/((para2)^(para3+1))))))
dpara11<-deriv(m1.exp[[1]],"para1")
dpara21<-deriv(m1.exp[[1]],"para2")
dpara31<-deriv(m1.exp[[1]],"para3")
dpara41<-deriv(m1.exp[[1]],"para4")

dpara12<-deriv(m2.exp[[1]],"para1")
dpara22<-deriv(m2.exp[[1]],"para2")
dpara32<-deriv(m2.exp[[1]],"para3")
dpara42<-deriv(m2.exp[[1]],"para4")

dpara13<-deriv(m3.exp[[1]],"para1")
dpara23<-deriv(m3.exp[[1]],"para2")
```

```
dpara33<-deriv(m3.exp[[1]],"para3")
dpara43<-deriv(m3.exp[[1]],"para4")

dpara14<-deriv(m4.exp[[1]],"para1")
dpara24<-deriv(m4.exp[[1]],"para2")
dpara34<-deriv(m4.exp[[1]],"para3")
dpara44<-deriv(m4.exp[[1]],"para4")

dpara111<-eval(dpara11)
dpara211<-eval(dpara21)
dpara311<-eval(dpara31)
dpara411<-eval(dpara41)

dpara122<-eval(dpara12)
dpara222<-eval(dpara22)
dpara322<-eval(dpara32)
dpara422<-eval(dpara42)

dpara133<-eval(dpara13)
dpara233<-eval(dpara23)
dpara333<-eval(dpara33)
dpara433<-eval(dpara43)

dpara144<-eval(dpara14)
dpara244<-eval(dpara24)
dpara344<-eval(dpara34)
dpara444<-eval(dpara44)

G<-
matrix(c(dpara111,dpara211,dpara311,dpara411,dpara122,dpara222,dpara322,
dpara422,dpara133,dpara233,dpara333,dpara433,dpara144,dpara244,dpara344,
dpara444),nrow=4,ncol=4)
return(G)
}
output.mm<-gmm(generalized_moment,x,theta.start,grad=Dg)

#################################### End ######################################
```