R Code for ‘Testing for random effects in compound risk models via Bregman divergence’

#### Load data files and creating training and test sets ####  
library(MASS)  
  
# load rawdata from URL  
load(url("https://sites.google.com/a/wisc.edu/jed-frees/home/documents/data.RData"))  
load(url("https://sites.google.com/a/wisc.edu/jed-frees/home/documents/dataout.RData"))  
  
train <- data[,c(1:2,4,9:14,21:25)] # Only IM claim is used  
rm(data)  
head(train)

trainp <- subset(train,log(yAvgIM)>0)  
  
test <- dataout[,c(1:2,4,9:14,21:25)]  
rm(dataout)  
head(test)

#### Estimation of alpha via marginal frequency likelihood ####  
  
NBEst <- function(x,n,id,init.alpha, r) { # optimization of proposed marginal likelihood for frequency  
 x <- cbind(rep(1,nrow(x)),x)  
 colnames(x)[1] <- "intercept"  
   
 # marginal likelihood of multivariate NB distribution  
 "negll.NB" <- function(parm) {  
 e <- ncol(x);  
 reg\_eqn <- as.matrix(x) %\*% parm[1:e]  
 data <- cbind(id,exp(reg\_eqn),n);  
 colnames(data)[2] <- "sv";  
 colnames(data)[3] <- "n";  
   
 temp1 = sum(n\*reg\_eqn-log(gamma(n+1)))+length(unique(id))\*(r\*log(r)-log(gamma(r)));  
 temp2 = -sum((as.matrix(aggregate(n~id,data,sum))[,2]+r)\*log(as.matrix(aggregate(sv~id,data,sum))[,2]+r)  
 )+sum(log(gamma(as.matrix(aggregate(n~id,data,sum))[,2]+r)));  
 result = -temp1-temp2  
 return(result)  
 }   
 init.est <- as.vector(init.alpha)  
   
 fit.NB <- optim(init.est, negll.NB, NULL)  
 parm.hat <- fit.NB$par  
 loglik.NB <- -fit.NB$value  
   
 # next estimate the standard errors.  
 library(nlme)  
 negll.NB.Hess <- fdHess(parm.hat, negll.NB);  
 inv.NB.Hess <- solve(negll.NB.Hess$Hessian);  
 parm.se <- sqrt(diag(inv.NB.Hess));  
 # put together the model with the est, se, t, pval, AIC, BIC  
 dfe <- length(n-length(parm.hat));  
 t\_ratio<-parm.hat/parm.se;  
 #test if diff. from 1 t\_ratio[1:3]<-(parm.hat[1:3]-1)/parm.se[1:3];  
 pval <- pf(t\_ratio\*t\_ratio,df1=1,df2=dfe,lower.tail=F);  
 ttable <- cbind(parm.hat,parm.se,t\_ratio,pval)   
 ttable <- round(ttable,digits=4)  
   
 rownames(ttable)<- c(colnames(x))  
 colnames(ttable)<- c("estimate", "std error", "t-val","Pr>|t|");  
   
 AIC<- 2\*negll.NB(parm.hat) + 2\*length(parm.hat);  
 BIC<- 2\*negll.NB(parm.hat) + log(length(n))\*length(parm.hat);  
 loglik <- negll.NB(parm.hat)  
 return(list(ttable=ttable,AIC=AIC,BIC=BIC,loglik=loglik,coef=parm.hat));  
}  
  
glm.freq <- glm(FreqIM~.,data=train[-c(1,2,3,6,13)], family="poisson") # optimization of naive marginal likelihood for frequency  
summary(glm.freq)

xn <- train[-c(1,2,3,6,13,14)]  
n <- train$FreqIM  
idn <- train$PolicyNum  
glmalpha <- coefficients(glm.freq)   
# the proposed marginal likelihood for frequency is optimized with r=3.8 and initial alpha from the naive model  
system.time(  
NBm <- NBEst(x=xn,id=idn,n=n,init.alpha=glmalpha, r=3.8))

## user system elapsed   
## 32.78 0.04 33.43

#### Estimation of beta via marginal severity likelihood ####  
  
GPEst <- function(c,x,n,id,init.beta,init.phi=1,k=11) { # optimization of proposed marginal likelihood for severity  
   
 x <- cbind(rep(1,nrow(x)),x)  
 colnames(x)[1] <- "intercept"  
   
 e <- ncol(x)  
 # marginal likelihood of multivariate GP distribution  
 "negll.GP" <- function(parm) {  
 e <- ncol(x);  
 reg\_eqn <- as.matrix(x) %\*% parm[1:e];  
 data <- cbind(id,n\*c/exp(reg\_eqn),n);  
 colnames(data)[2] <- "sv";  
   
 temp1 = (sum(lgamma(as.matrix(aggregate(n~id,data,sum))[,2]/parm[e+1]+k+1))-sum(log(c))  
 +sum(n\*(log(n\*c)-reg\_eqn-log(parm[e+1])))/parm[e+1]+length(unique(id))\*(k+1)\*log(k)  
 -sum(lgamma(n/parm[e+1]))-length(unique(id))\*lgamma(k+1))  
 temp2 = (-sum((as.matrix(aggregate(n~id,data,sum))[,2]/parm[e+1]+k+1)\*  
 log(as.matrix(aggregate(sv~id,data,sum))[,2]/parm[e+1]+k)))  
 result = -temp1-temp2  
 return(result) }  
 init.est <- as.vector(c(init.beta,init.phi))  
   
 fit.GP <- optim(init.est, negll.GP, NULL)  
 parm.hat <- fit.GP$par  
   
 # next estimate the standard errors.  
 library(nlme)  
 negll.GP.Hess <- fdHess(parm.hat, negll.GP);  
 inv.GP.Hess <- solve(negll.GP.Hess$Hessian);  
 parm.se <- sqrt(diag(inv.GP.Hess));  
 # put together the model with the est, se, t, pval, AIC, BIC  
 dfe <- length(c-length(parm.hat));  
 t\_ratio<-parm.hat/parm.se;  
 ##test if diff. from 1 t\_ratio[1:3]<-(parm.hat[1:3]-1)/parm.se[1:3];  
 pval <- pf(t\_ratio\*t\_ratio,df1=1,df2=dfe,lower.tail=F);  
 ttable <- cbind(parm.hat,parm.se,t\_ratio,pval)   
 ttable <- round(ttable,digits=4)  
   
 rownames(ttable)<- c(colnames(x),"phi")  
 colnames(ttable)<- c("estimate", "std error", "t-val","Pr>|t|");  
   
 AIC<- 2\*negll.GP(parm.hat) + 2\*length(parm.hat);  
 BIC<- 2\*negll.GP(parm.hat) + log(length(c))\*length(parm.hat);  
 loglik <- -negll.GP(parm.hat)  
 return(list(ttable=ttable,AIC=AIC,BIC=BIC,loglik=loglik,coef=parm.hat));  
}  
glm.avgsev\_dep <- glm(trainp$yAvgIM~.,data=trainp[-c(1,2,3,6,13)],  
 family=Gamma(link="log"),weights = trainp$FreqIM) # optimization of naive marginal likelihood for severity  
glmbeta <- coefficients(glm.avgsev\_dep)  
  
# the proposed marginal likelihood for severity is optimized with k=11 and initial beta from the naive model  
system.time(  
GPm <- GPEst(c=trainp$yAvgIM,x=trainp[-c(1,2,3,6,13)],id=trainp$PolicyNum,  
 n=trainp$FreqIM,init.beta=glmbeta) )

## user system elapsed   
## 7.81 0.00 7.95

#### Out-of sample validation ####  
  
x <- train[-c(1,2,3,6,13)]  
id <- train$PolicyNum  
x <- cbind(rep(1,nrow(x)),x)  
n <- train$FreqIM  
c <- train$yAvgIM  
r <- 3.8  
k <- 11  
  
# calculation of bonus-malus factor based on the proposed models for frequency and average severity  
Nreg\_eqn <- as.matrix(x[,-10]) %\*% as.matrix(NBm$coef[1:9])  
Ndata <- cbind(id,exp(Nreg\_eqn),n);  
colnames(Ndata)[2] <- "nv";  
Npost <- aggregate(n~id,Ndata,sum) # aggregate actual claim counts for years  
Npost$nv <- aggregate(nv~id,Ndata,sum)[,2] # aggregate expected claim counts for years  
Npost$nweight <- (Npost$n + r) / (Npost$nv + r) # bonus-malus factor for a policyholder on frequency  
colnames(Npost)[1] <- "PolicyNum"  
Npost$n <- NULL  
Npost$nv <- NULL  
  
Creg\_eqn <- as.matrix(x) %\*% GPm$coef[1:ncol(x)]  
Cdata <- cbind(id,n\*c/exp(Creg\_eqn),n);  
colnames(Cdata)[2] <- "sv";  
Cpost <- aggregate(n~id,Cdata,sum) # aggregate actual exposures for years  
Cpost$sv <- aggregate(sv~id,Cdata,sum)[,2] # aggregate (actual claim size/expected claim size) for years  
Cpost$cweight <- (Cpost$sv + GPm$coef[11]\*k) / (Cpost$n + GPm$coef[11]\*k) # bonus-malus factor for a policyholder on severity  
colnames(Cpost)[1] <- "PolicyNum"  
Cpost$n <- NULL  
Cpost$sv <- NULL  
  
# attach the bonus-malus factor for each policyholder on the test set  
Ptest <- merge(x = test, y = Npost, by = "PolicyNum", all.x = TRUE)  
Ptest$nweight[is.na(Ptest$nweight)] <- 1  
  
Ptest <- merge(x = Ptest, y = Cpost, by = "PolicyNum", all.x = TRUE)  
Ptest$cweight[is.na(Ptest$cweight)] <- 1 # if there is no observation on the previous years, then bonus-malus factor is set as default, 1.  
  
xt <- test[-c(1,2,3,6,13,14)]  
xt <- cbind(rep(1,nrow(xt)),xt)  
  
n\_npred <- exp(as.matrix(xt) %\*% glmalpha) # frequency premium with the naive model  
n\_ppred <- exp(as.matrix(xt) %\*% as.matrix(NBm$coef[1:9]))\*Ptest$nweight # frequency premium with the proposed model  
  
c\_npred <- exp(as.matrix(xt) %\*% glmbeta[1:ncol(xt)] + n\_npred\*glmbeta[10]) # severity premium with the naive model  
c\_ppred <- exp(as.matrix(xt) %\*% GPm$coef[1:ncol(xt)] # severity premium with the proposed model  
 + n\_ppred\*GPm$coef[10])\*Ptest$cweight  
  
S\_npred <- n\_npred\*c\_npred # total premium with the naive model  
S\_ppred <- n\_ppred\*c\_ppred # total premium with the proposed model  
  
# root mean squared errors  
RMSE\_naive <- sqrt(mean((S\_npred - test$ClaimIM)^2))  
RMSE\_proposed <- sqrt(mean((S\_ppred - test$ClaimIM)^2))  
  
# mean absoulte errors  
MAE\_naive <- mean(abs(S\_npred - test$ClaimIM))  
MAE\_proposed <- mean(abs(S\_ppred - test$ClaimIM))  
  
#### function declaration ####  
  
sdn <- sqrt(1/r) # standard deviation of theta\_N with the proposed prior  
sdc <- sqrt(1/(k-1)) # standard deviation of theta\_C with the proposed prior  
  
psi1ftn <- function(x) x\*log(x)-x+1 # convex function to be used in Bregman divergence  
qn1 <- function(x) dunif(x,1-sdn\*sqrt(3)/2,1+sdn\*sqrt(3)/2) # density of freq uniform perturbation  
qn2 <- function(x) dlnorm(x,-log(1+(sdn/2)^2)/2,  
 sqrt(log(1+(sdn/2)^2))) # density of freq lognormal perturbation  
qn3 <- function(x) dnorm(x,1,sdn/2) # density of freq normal perturbation  
  
qc1 <- function(x) dunif(x,1-sdc\*sqrt(3)/2,1+sdc\*sqrt(3)/2) # density of sev uniform perturbation  
qc2 <- function(x) dlnorm(x,-log(1+(sdc/2)^2)/2,  
 sqrt(log(1+(sdc/2)^2))) # density of sev lognormal perturbation  
qc3 <- function(x) dnorm(x,1,sdc/2) # density of sev normal perturbation  
  
  
pms <- function(u) dnorm(u,1,10^(-12)) # naive prior (point mass at 1)  
gpi <- function(u) dgamma(u,shape=r,scale=1/r) # proposed prior for frequency (gamma)  
igpi <- function(u) 1/gamma(k+1)\*(k/u)^{k+1}\*exp(-k/u)/u # proposed prior for severity (inverse gamma)  
  
  
fC <- function(c,mu,theta,psi) {  
 1/c/gamma(psi)\*(c\*psi/mu/theta)^{psi}\*exp(-c\*psi/mu/theta) # data likelihood for severity (gamma specified with mu)  
}  
  
fN <- function(n,nu,theta) {  
 1/factorial(n)\*(nu\*theta)^{n}\*exp(-nu\*theta) # data likelihood for frequency (poisson specified with nu)  
}  
  
# According to Lemma 1 in the paper, one needs the following ratios of marginal densities  
  
# ratio of marginal frequency density with q prior to marginal frequency density with naive prior (1 observation during 5 years)   
rNm\_q1 <- function(z1,nu1,theta) {   
 mean(fN(z1,nu1,theta))/fN(z1,nu1,1)   
}  
# ratio of marginal frequency density with q prior to marginal frequency density with naive prior (2 observations during 5 years)   
rNm\_q2 <- function(z1,nu1,theta,z2,nu2) {   
 mean(fN(z1,nu1,theta)\*fN(z2,nu2,theta))/fN(z1,nu1,1)\*fN(z2,nu2,1)  
}  
# ratio of marginal frequency density with q prior to marginal frequency density with naive prior (3 observations during 5 years)   
rNm\_q3 <- function(z1,nu1,theta,z2,nu2,z3,nu3) {   
 mean(fN(z1,nu1,theta)\*fN(z2,nu2,theta)\*fN(z3,nu3,theta)  
 )/fN(z1,nu1,1)\*fN(z2,nu2,1)\*fN(z3,nu3,1)  
}  
# ratio of marginal frequency density with q prior to marginal frequency density with naive prior (4 observations during 5 years)   
rNm\_q4 <- function(z1,nu1,theta,z2,nu2,z3,nu3,z4,nu4) {   
 mean(fN(z1,nu1,theta)\*fN(z2,nu2,theta)\*fN(z3,nu3,theta)\*fN(z4,nu4,theta)  
 )/fN(z1,nu1,1)\*fN(z2,nu2,1)\*fN(z3,nu3,1)\*fN(z4,nu4,1)  
}  
# ratio of marginal frequency density with q prior to marginal frequency density with naive prior (5 observations during 5 years)   
rNm\_q5 <- function(z1,nu1,theta,z2,nu2,z3,nu3,z4,nu4,z5,nu5){  
 mean(fN(z1,nu1,theta)\*fN(z2,nu2,theta)\*fN(z3,nu3,theta)\*fN(z4,nu4,theta)\*fN(z5,nu5,theta)  
 )/fN(z1,nu1,1)\*fN(z2,nu2,1)\*fN(z3,nu3,1)\*fN(z4,nu4,1)\*fN(z5,nu5,1)  
}  
  
# ratio of marginal severity density with q prior to marginal severity density with naive prior (1 observation during 5 years)   
rCm\_q1 <- function(z1,mu1,theta,psi1) {   
 mean(fC(z1,mu1,theta,psi1))/fC(z1,mu1,1,psi1)  
}  
# ratio of marginal severity density with q prior to marginal severity density with naive prior (2 observations during 5 years)   
rCm\_q2 <- function(z1,mu1,theta,psi1,z2,mu2,psi2) {   
 mean(fC(z1,mu1,theta,psi1)\*fC(z2,mu2,theta,psi2))/fC(z1,mu1,1,psi1)\*fC(z2,mu2,1,psi2)  
}  
# ratio of marginal severity density with q prior to marginal severity density with naive prior (3 observations during 5 years)   
rCm\_q3 <- function(z1,mu1,theta,psi1,z2,mu2,psi2,z3,mu3,psi3) {   
 mean(fC(z1,mu1,theta,psi1)\*fC(z2,mu2,theta,psi2)\*fC(z3,mu3,theta,psi3)  
 )/fC(z1,mu1,1,psi1)\*fC(z2,mu2,1,psi2)\*fC(z3,mu3,1,psi3)  
}  
# ratio of marginal severity density with q prior to marginal severity density with naive prior (4 observations during 5 years)   
rCm\_q4 <- function(z1,mu1,theta,psi1,z2,mu2,psi2,z3,mu3,psi3,z4,mu4,psi4) {   
 mean(fC(z1,mu1,theta,psi1)\*fC(z2,mu2,theta,psi2)\*fC(z3,mu3,theta,psi3)\*fC(z4,mu4,theta,psi4)  
 )/fC(z1,mu1,1,psi1)\*fC(z2,mu2,1,psi2)\*fC(z3,mu3,1,psi3)\*fC(z4,mu4,1,psi4)  
}  
# ratio of marginal severity density with q prior to marginal severity density with naive prior (5 observations during 5 years)   
rCm\_q5 <- function(z1,mu1,theta,psi1,z2,mu2,psi2,z3,mu3,psi3,z4,mu4,psi4,z5,mu5,psi5) {   
 mean(fC(z1,mu1,theta,psi1)\*fC(z2,mu2,theta,psi2)\*fC(z3,mu3,theta,psi3)\*fC(z4,mu4,theta,psi4)\*fC(z5,mu5,theta,psi5)  
 )/fC(z1,mu1,1,psi1)\*fC(z2,mu2,1,psi2)\*fC(z3,mu3,1,psi3)\*fC(z4,mu4,1,psi4)\*fC(z5,mu5,1,psi5)  
}  
  
  
#### sensitivity for frequency with Uniform ####  
  
set.seed(108)  
theta1 <- runif(1000,1-sdn\*sqrt(3)/2,1+sdn\*sqrt(3)/2)  
# MC samples of theta\_N are generated from uniform distribution  
  
count <- aggregate(Year~PolicyNum,train,length)  
colnames(count)[2] <- "Repeat"  
train <- merge(train,count)  
head(train, 11)

rm(count)  
# training set is ordered by number of repetition, policynumber, and years to apply ratios of densities accordingly with loops  
train <- train[ with(train, order(Repeat, PolicyNum,Year)), ]  
Repeatt <- train$Repeat  
train$Repeat <- NULL  
table(Repeatt)

N1 <- as.numeric(table(Repeatt)[1]) # number of policyholders with 1 frequency observation  
N2 <- as.numeric(table(Repeatt)[2]) # number of policyholders with 2 frequency observations  
N3 <- as.numeric(table(Repeatt)[3]) # number of policyholders with 3 frequency observations  
N4 <- as.numeric(table(Repeatt)[4]) # number of policyholders with 4 frequency observations  
N5 <- as.numeric(table(Repeatt)[5]) # number of policyholders with 5 frequency observations  
N <- N1 + N2 + N3 + N4 + N5  
  
xnn <- cbind(rep(1,nrow(xn)),xn)  
nu <- exp(as.matrix(xnn) %\*% as.matrix(NBm$coef[1:9]))  
n <- train$FreqIM  
rm(xnn)  
e <- 1:99/100 # level of perturbation: 0.00-0.99  
  
gpsens11 <- rep(0,99)  
Npmsens11 <- rep(0,99)  
for (j in 1:N1) { # Contribution to the sensitivity for every perturbation level for the policyholders with 1 frequency observation  
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Npmsens11 <- Npmsens11 + psi1ftn( (1-e)+e\*qn1(1)/pms(1) / (1-e+e\*rNm\_q1(n[j],nu[j],theta1)) ) /N  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_N are generated from the proposed posterior)  
 thetap <- rgamma(1000,shape=r+n[j],scale=1/(r+nu[j]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta1 <- (1-err)+err\*qn1(thetap)/gpi(thetap)  
 gpdelta1 <- gpdelta1/mean(gpdelta1)  
 gpsens11[i] <- gpsens11[i] + mean(psi1ftn(gpdelta1))/N  
 }  
}  
  
gpsens12 <- rep(0,99)  
Npmsens12 <- rep(0,99)  
for (j in 1:N2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 frequency observations  
 j1 <- N1+2\*j-1  
 j2 <- N1+2\*j  
   
 Npmsens12 <- Npmsens12 + psi1ftn( (1-e)+e\*qn1(1)/pms(1) / (1-e+e\*rNm\_q2  
 (n[j1],nu[j1],theta1,n[j2],nu[j2])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2],scale=1/(r+nu[j1]+nu[j2]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta1 <- (1-err)+err\*qn1(thetap)/gpi(thetap)  
 gpdelta1 <- gpdelta1/mean(gpdelta1)  
 gpsens12[i] <- gpsens12[i] + mean(psi1ftn(gpdelta1))/N  
 }  
}  
  
gpsens13 <- rep(0,99)  
Npmsens13 <- rep(0,99)  
for (j in 1:N3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 frequency observations  
 j1 <- N1+N2+3\*j-2  
 j2 <- N1+N2+3\*j-1  
 j3 <- N1+N2+3\*j  
 Npmsens13 <- Npmsens13 + psi1ftn( (1-e)+e\*qn1(1)/pms(1) / (1-e+e\*rNm\_q3  
 (n[j1],nu[j1],theta1,n[j2],nu[j2],n[j3],nu[j3])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3],scale=1/(r+nu[j1]+nu[j2]+nu[j3]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta1 <- (1-err)+err\*qn1(thetap)/gpi(thetap)  
 gpdelta1 <- gpdelta1/mean(gpdelta1)  
 gpsens13[i] <- gpsens13[i] + mean(psi1ftn(gpdelta1))/N  
 }  
}  
  
gpsens14 <- rep(0,99)  
Npmsens14 <- rep(0,99)  
for (j in 1:N4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 frequency observations  
 j1 <- N1+N2+N3+4\*j-3  
 j2 <- N1+N2+N3+4\*j-2  
 j3 <- N1+N2+N3+4\*j-1  
 j4 <- N1+N2+N3+4\*j  
 Npmsens14 <- Npmsens14 + psi1ftn( (1-e)+e\*qn1(1)/pms(1) / (1-e+e\*rNm\_q4  
 (n[j1],nu[j1],theta1,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta1 <- (1-err)+err\*qn1(thetap)/gpi(thetap)  
 gpdelta1 <- gpdelta1/mean(gpdelta1)  
 gpsens14[i] <- gpsens14[i] + mean(psi1ftn(gpdelta1))/N  
 }  
}  
  
gpsens15 <- rep(0,99)  
Npmsens15 <- rep(0,99)  
for (j in 1:N5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 frequency observations  
 j1 <- N1+N2+N3+N4+5\*j-4  
 j2 <- N1+N2+N3+N4+5\*j-3  
 j3 <- N1+N2+N3+N4+5\*j-2  
 j4 <- N1+N2+N3+N4+5\*j-1  
 j5 <- N1+N2+N3+N4+5\*j  
 Npmsens15 <- Npmsens15 + psi1ftn( (1-e)+e\*qn1(1)/pms(1) / (1-e+e\*rNm\_q5  
 (n[j1],nu[j1],theta1,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4],n[j5],nu[j5])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4]+n[j5],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]+nu[j5]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta1 <- (1-err)+err\*qn1(thetap)/gpi(thetap)  
 gpdelta1 <- gpdelta1/mean(gpdelta1)  
 gpsens15[i] <- gpsens15[i] + mean(psi1ftn(gpdelta1))/N   
 }  
}  
  
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Npmsens1 <- Npmsens11+Npmsens12+Npmsens13+Npmsens14+Npmsens15   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level   
gpsens1 <- gpsens11+gpsens12+gpsens13+gpsens14+gpsens15   
plot(e,Npmsens1,type='l',col="blue")  
lines(e,gpsens1)

#### sensitivity for frequency with Lognormal ####  
  
set.seed(108)  
theta2 <- rlnorm(1000,-log(1+(sdn/2)^2)/2,sqrt(log(1+(sdn/2)^2)))  
# MC samples of theta\_N are generated from lognormal distribution  
  
gpsens21 <- rep(0,99)  
Npmsens21 <- rep(0,99)  
for (j in 1:N1) { # Contribution to the sensitivity for every perturbation level for the policyholders with 1 frequency observation  
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Npmsens21 <- Npmsens21 + psi1ftn( (1-e)+e\*qn2(1)/pms(1) / (1-e+e\*rNm\_q1(n[j],nu[j],theta2)) ) /N  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_N are generated from the proposed posterior)  
 thetap <- rgamma(1000,shape=r+n[j],scale=1/(r+nu[j]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta2 <- (1-err)+err\*qn2(thetap)/gpi(thetap)  
 gpdelta2 <- gpdelta2/mean(gpdelta2)  
 gpsens21[i] <- gpsens21[i] + mean(psi1ftn(gpdelta2))/N  
 }  
}  
  
gpsens22 <- rep(0,99)  
Npmsens22 <- rep(0,99)  
for (j in 1:N2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 frequency observations  
 j1 <- N1+2\*j-1  
 j2 <- N1+2\*j  
   
 Npmsens22 <- Npmsens22 + psi1ftn( (1-e)+e\*qn2(1)/pms(1) /   
 (1-e+e\*rNm\_q2(n[j1],nu[j1],theta2,n[j2],nu[j2])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2],scale=1/(r+nu[j1]+nu[j2]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta2 <- (1-err)+err\*qn2(thetap)/gpi(thetap)  
 gpdelta2 <- gpdelta2/mean(gpdelta2)  
 gpsens22[i] <- gpsens22[i] + mean(psi1ftn(gpdelta2))/N  
 }  
}  
  
gpsens23 <- rep(0,99)  
Npmsens23 <- rep(0,99)  
for (j in 1:N3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 frequency observations  
 j1 <- N1+N2+3\*j-2  
 j2 <- N1+N2+3\*j-1  
 j3 <- N1+N2+3\*j  
 Npmsens23 <- Npmsens23 + psi1ftn( (1-e)+e\*qn2(1)/pms(1) /   
 (1-e+e\*rNm\_q3(n[j1],nu[j1],theta2,n[j2],nu[j2],n[j3],nu[j3])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3],scale=1/(r+nu[j1]+nu[j2]+nu[j3]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta2 <- (1-err)+err\*qn2(thetap)/gpi(thetap)  
 gpdelta2 <- gpdelta2/mean(gpdelta2)  
 gpsens23[i] <- gpsens23[i] + mean(psi1ftn(gpdelta2))/N  
 }  
}  
  
gpsens24 <- rep(0,99)  
Npmsens24 <- rep(0,99)  
for (j in 1:N4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 frequency observations  
 j1 <- N1+N2+N3+4\*j-3  
 j2 <- N1+N2+N3+4\*j-2  
 j3 <- N1+N2+N3+4\*j-1  
 j4 <- N1+N2+N3+4\*j  
 Npmsens24 <- Npmsens24 + psi1ftn( (1-e)+e\*qn2(1)/pms(1) /   
 (1-e+e\*rNm\_q4(n[j1],nu[j1],theta2,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta2 <- (1-err)+err\*qn2(thetap)/gpi(thetap)  
 gpdelta2 <- gpdelta2/mean(gpdelta2)  
 gpsens24[i] <- gpsens24[i] + mean(psi1ftn(gpdelta2))/N  
 }  
}  
  
gpsens25 <- rep(0,99)  
Npmsens25 <- rep(0,99)  
for (j in 1:N5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 frequency observations  
 j1 <- N1+N2+N3+N4+5\*j-4  
 j2 <- N1+N2+N3+N4+5\*j-3  
 j3 <- N1+N2+N3+N4+5\*j-2  
 j4 <- N1+N2+N3+N4+5\*j-1  
 j5 <- N1+N2+N3+N4+5\*j  
 Npmsens25 <- Npmsens25 + psi1ftn( (1-e)+e\*qn2(1)/pms(1) /   
 (1-e+e\*rNm\_q5(n[j1],nu[j1],theta2,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4],n[j5],nu[j5])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4]+n[j5],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]+nu[j5]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta2 <- (1-err)+err\*qn2(thetap)/gpi(thetap)  
 gpdelta2 <- gpdelta2/mean(gpdelta2)  
 gpsens25[i] <- gpsens25[i] + mean(psi1ftn(gpdelta2))/N   
 }  
}  
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Npmsens2 <- Npmsens21+Npmsens22+Npmsens23+Npmsens24+Npmsens25   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level  
gpsens2 <- gpsens21+gpsens22+gpsens23+gpsens24+gpsens25   
plot(e,Npmsens2,type='l',col="blue")  
lines(e,gpsens2)

#### sensitivity for frequency with Normal ####  
set.seed(108)  
theta3 <- rnorm(1000,1,sdn/2)  
# MC samples of theta\_N are generated from normal distribution  
  
gpsens31 <- rep(0,99)  
Npmsens31 <- rep(0,99)  
for (j in 1:N1) { # Contribution to the sensitivity for every perturbation level for the policyholders with 1 frequency observation  
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Npmsens31 <- Npmsens31 + psi1ftn( (1-e)+e\*qn3(1)/pms(1) / (1-e+e\*rNm\_q1(n[j],nu[j],theta3)) ) /N  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_N are generated from the proposed posterior)  
 thetap <- rgamma(1000,shape=r+n[j],scale=1/(r+nu[j]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta3 <- (1-err)+err\*qn3(thetap)/gpi(thetap)  
 gpdelta3 <- gpdelta3/mean(gpdelta3)  
 gpsens31[i] <- gpsens31[i] + mean(psi1ftn(gpdelta3))/N  
 }  
}  
  
gpsens32 <- rep(0,99)  
Npmsens32 <- rep(0,99)  
for (j in 1:N2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 frequency observations  
 j1 <- N1+2\*j-1  
 j2 <- N1+2\*j  
   
 Npmsens32 <- Npmsens32 + psi1ftn( (1-e)+e\*qn3(1)/pms(1) /   
 (1-e+e\*rNm\_q2(n[j1],nu[j1],theta3,n[j2],nu[j2])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2],scale=1/(r+nu[j1]+nu[j2]))  
 for (i in 1:99) {   
 err <- e[i]  
 gpdelta3 <- (1-err)+err\*qn3(thetap)/gpi(thetap)  
 gpdelta3 <- gpdelta3/mean(gpdelta3)  
 gpsens32[i] <- gpsens32[i] + mean(psi1ftn(gpdelta3))/N  
 }  
}  
  
gpsens33 <- rep(0,99)  
Npmsens33 <- rep(0,99)  
for (j in 1:N3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 frequency observations  
 j1 <- N1+N2+3\*j-2  
 j2 <- N1+N2+3\*j-1  
 j3 <- N1+N2+3\*j  
 Npmsens33 <- Npmsens33 + psi1ftn( (1-e)+e\*qn3(1)/pms(1) /   
 (1-e+e\*rNm\_q3(n[j1],nu[j1],theta3,n[j2],nu[j2],n[j3],nu[j3])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3],scale=1/(r+nu[j1]+nu[j2]+nu[j3]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta3 <- (1-err)+err\*qn3(thetap)/gpi(thetap)  
 gpdelta3 <- gpdelta3/mean(gpdelta3)  
 gpsens33[i] <- gpsens33[i] + mean(psi1ftn(gpdelta3))/N  
 }  
}  
  
gpsens34 <- rep(0,99)  
Npmsens34 <- rep(0,99)  
for (j in 1:N4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 frequency observations  
 j1 <- N1+N2+N3+4\*j-3  
 j2 <- N1+N2+N3+4\*j-2  
 j3 <- N1+N2+N3+4\*j-1  
 j4 <- N1+N2+N3+4\*j  
 Npmsens34 <- Npmsens34 + psi1ftn( (1-e)+e\*qn3(1)/pms(1) /   
 (1-e+e\*rNm\_q4(n[j1],nu[j1],theta3,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta3 <- (1-err)+err\*qn3(thetap)/gpi(thetap)  
 gpdelta3 <- gpdelta3/mean(gpdelta3)  
 gpsens34[i] <- gpsens34[i] + mean(psi1ftn(gpdelta3))/N  
 }  
}  
  
gpsens35 <- rep(0,99)  
Npmsens35 <- rep(0,99)  
for (j in 1:N5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 frequency observations  
 j1 <- N1+N2+N3+N4+5\*j-4  
 j2 <- N1+N2+N3+N4+5\*j-3  
 j3 <- N1+N2+N3+N4+5\*j-2  
 j4 <- N1+N2+N3+N4+5\*j-1  
 j5 <- N1+N2+N3+N4+5\*j  
 Npmsens35 <- Npmsens35 + psi1ftn( (1-e)+e\*qn3(1)/pms(1) /   
 (1-e+e\*rNm\_q5(n[j1],nu[j1],theta2,n[j2],nu[j2],n[j3],nu[j3],n[j4],nu[j4],n[j5],nu[j5])) ) /N  
 thetap <- rgamma(1000,shape=r+n[j1]+n[j2]+n[j3]+n[j4]+n[j5],  
 scale=1/(r+nu[j1]+nu[j2]+nu[j3]+nu[j4]+nu[j5]))  
 for (i in 1:99) {  
 err <- e[i]  
 gpdelta3 <- (1-err)+err\*qn3(thetap)/gpi(thetap)  
 gpdelta3 <- gpdelta3/mean(gpdelta3)  
 gpsens35[i] <- gpsens35[i] + mean(psi1ftn(gpdelta3))/N   
 }  
}  
  
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Npmsens3 <- Npmsens31+Npmsens32+Npmsens33+Npmsens34+Npmsens35   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level  
gpsens3 <- gpsens31+gpsens32+gpsens33+gpsens34+gpsens35   
plot(e,Npmsens3,type='l',col="blue")  
lines(e,gpsens3)

#### sensitivity for severity with Uniform ####  
  
library(dplyr)

count <- aggregate(Year~PolicyNum,trainp,length)  
colnames(count)[2] <- "Repeat"  
trainp <- merge(trainp,count)  
head(trainp, 11)

rm(count)  
  
trainp <- trainp[ with(trainp, order(Repeat, PolicyNum,Year)), ]  
# training set is ordered by number of repetition, policynumber, and years to apply ratios of densities accordingly with loops  
Repeat <- trainp$Repeat  
trainp$Repeat <- NULL  
  
M1 <- as.numeric(table(Repeat)[1]) # number of policyholders with 1 severity observation  
M2 <- as.numeric(table(Repeat)[2]) # number of policyholders with 2 severity observations  
M3 <- as.numeric(table(Repeat)[3]) # number of policyholders with 3 severity observations  
M4 <- as.numeric(table(Repeat)[4]) # number of policyholders with 4 severity observations  
M5 <- as.numeric(table(Repeat)[5]) # number of policyholders with 5 severity observations  
M <- M1 + M2 + M3 + M4 + M5  
  
x <- trainp[-c(1,2,3,6,13)]  
id <- trainp$PolicyNum  
x <- cbind(rep(1,nrow(x)),x)  
n <- trainp$FreqIM  
c <- trainp$yAvgIM  
psi <- n/GPm$coef[11]  
k <- 11  
mu <- exp(as.matrix(x) %\*% GPm$coef[1:10])  
  
  
set.seed(108)  
theta1 <- runif(1000,1-sdc\*sqrt(3)/2,1+sdc\*sqrt(3)/2)  
# MC samples of theta\_C are generated from uniform distribution  
  
igpsens10 <- rep(0,99)  
Cpmsens10 <- rep(0,99)  
# For the data points without severity, sensitivity is measured only with prior   
 Cpmsens10 <- psi1ftn( (1-e)+e\*qc1(1)/pms(1) )  
 thetap <- 1/rgamma(1000,k+1,k)  
 for (i in 1:99) { # Contribution to the sensitivity for every perturbation level for the policyholders with 0 severity observation  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens10[i] <- igpsens10[i] + mean(psi1ftn(igpdelta1))  
 }  
  
igpsens11 <- rep(0,99)  
Cpmsens11 <- rep(0,99)  
for (j in 1:M1) { # Contribution to the sensitivity for every perturbation level for the policyholders with 1 severity observation  
   
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Cpmsens11 <- Cpmsens11 + psi1ftn( (1-e)+e\*qc1(1)/pms(1) / (1-e+e\*rCm\_q1(c[j],mu[j],theta1,psi[j])) ) /M  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_C are generated from the proposed posterior)  
   
 thetap <- 1/rgamma(1000,k+1+psi[j],k+psi[j]\*c[j]/mu[j])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens11[i] <- igpsens11[i] + mean(psi1ftn(igpdelta1))/M   
 }  
}  
  
igpsens12 <- rep(0,99)  
Cpmsens12 <- rep(0,99)  
for (j in 1:M2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 severity observations  
  
 j1 <- M1+2\*j-1  
 j2 <- M1+2\*j  
   
 Cpmsens12 <- Cpmsens12 + psi1ftn( (1-e)+e\*qc1(1)/pms(1) / (1-e+e\*rCm\_q2  
 (c[j1],mu[j1],theta1,psi[j1],c[j2],mu[j2],psi[j2])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2],k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens12[i] <- igpsens12[i] + mean(psi1ftn(igpdelta1))/M   
 }  
}  
  
igpsens13 <- rep(0,99)  
Cpmsens13 <- rep(0,99)  
for (j in 1:M3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 severity observations  
 j1 <- M1+M2+3\*j-2  
 j2 <- M1+M2+3\*j-1  
 j3 <- M1+M2+3\*j  
 Cpmsens13 <- Cpmsens13 + psi1ftn( (1-e)+e\*qc1(1)/pms(1) / (1-e+e\*rCm\_q3  
 (c[j1],mu[j1],theta1,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens13[i] <- igpsens13[i] + mean(psi1ftn(igpdelta1))/M   
 }  
}  
  
igpsens14 <- rep(0,99)  
Cpmsens14 <- rep(0,99)  
for (j in 1:M4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 severity observations  
 j1 <- M1+M2+M3+4\*j-3  
 j2 <- M1+M2+M3+4\*j-2  
 j3 <- M1+M2+M3+4\*j-1  
 j4 <- M1+M2+M3+4\*j  
 Cpmsens14 <- Cpmsens14 + psi1ftn( (1-e)+e\*qc1(1)/pms(1) / (1-e+e\*rCm\_q4  
 (c[j1],mu[j1],theta1,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens14[i] <- igpsens14[i] + mean(psi1ftn(igpdelta1))/M   
 }  
}  
  
igpsens15 <- rep(0,99)  
Cpmsens15 <- rep(0,99)  
for (j in 1:M5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 severity observations  
 j1 <- M1+M2+M3+M4+5\*j-4  
 j2 <- M1+M2+M3+M4+5\*j-3  
 j3 <- M1+M2+M3+M4+5\*j-2  
 j4 <- M1+M2+M3+M4+5\*j-1  
 j5 <- M1+M2+M3+M4+5\*j  
 Cpmsens15 <- Cpmsens15 + psi1ftn( (1-e)+e\*qc1(1)/pms(1) / (1-e+e\*rCm\_q5  
 (c[j1],mu[j1],theta1,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4],c[j5],mu[j5],psi[j5])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]+psi[j5]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4]+psi[j5]\*c[j5]/mu[j5])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta1 <- (1-err)+err\*qc1(thetap)/igpi(thetap)  
 igpdelta1 <- igpdelta1/mean(igpdelta1)  
 igpsens15[i] <- igpsens15[i] + mean(psi1ftn(igpdelta1))/M   
 }  
}  
  
w <- length(unique(trainp$PolicyNum))/length(unique(train$PolicyNum))   
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Cpmsens1 <- (1-w)\*Cpmsens10 + (Cpmsens11+Cpmsens12+Cpmsens13+Cpmsens14+Cpmsens15)\*w   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level  
igpsens1 <- (1-w)\*igpsens10 + (igpsens11+igpsens12+igpsens13+igpsens14+igpsens15)\*w   
plot(e,Cpmsens1,type='l',col="blue")  
lines(e,igpsens1)

d#### sensitivity for severity with Lognormal ####  
set.seed(108)  
theta2 <- rlnorm(1000,-log(1+(sdc/2)^2)/2,sqrt(log(1+(sdc/2)^2)))  
# MC samples of theta\_C are generated from lognormal distribution  
  
igpsens20 <- rep(0,99)  
Cpmsens20 <- rep(0,99)  
# For the data points without severity, sensitivity is measured only with prior  
  
Cpmsens20 <- psi1ftn( (1-e)+e\*qc2(1)/pms(1) )  
thetap <- 1/rgamma(1000,k+1,k)  
for (i in 1:99) { # Contribution to the sensitivity for every perturbation level for the policyholders with 0 severity observation  
  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens20[i] <- igpsens20[i] + mean(psi1ftn(igpdelta2))  
}  
  
  
igpsens21 <- rep(0,99)  
Cpmsens21 <- rep(0,99)  
for (j in 1:M1) {  
   
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Cpmsens21 <- Cpmsens21 + psi1ftn( (1-e)+e\*qc2(1)/pms(1) / (1-e+e\*rCm\_q1(c[j],mu[j],theta2,psi[j])) ) /M  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_C are generated from the proposed posterior)  
 thetap <- 1/rgamma(1000,k+1+psi[j],k+psi[j]\*c[j]/mu[j])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens21[i] <- igpsens21[i] + mean(psi1ftn(igpdelta2))/M   
 }  
}  
  
igpsens22 <- rep(0,99)  
Cpmsens22 <- rep(0,99)  
for (j in 1:M2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 severity observations  
 j1 <- M1+2\*j-1  
 j2 <- M1+2\*j  
   
 Cpmsens22 <- Cpmsens22 + psi1ftn( (1-e)+e\*qc2(1)/pms(1) /   
 (1-e+e\*rCm\_q2(c[j1],mu[j1],theta2,psi[j1],c[j2],mu[j2],psi[j2])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2],k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens22[i] <- igpsens22[i] + mean(psi1ftn(igpdelta2))/M   
 }  
}  
  
igpsens23 <- rep(0,99)  
Cpmsens23 <- rep(0,99)  
for (j in 1:M3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 severity observations  
 j1 <- M1+M2+3\*j-2  
 j2 <- M1+M2+3\*j-1  
 j3 <- M1+M2+3\*j  
 Cpmsens23 <- Cpmsens23 + psi1ftn( (1-e)+e\*qc2(1)/pms(1) /   
 (1-e+e\*rCm\_q3(c[j1],mu[j1],theta2,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens23[i] <- igpsens23[i] + mean(psi1ftn(igpdelta2))/M   
 }  
}  
  
igpsens24 <- rep(0,99)  
Cpmsens24 <- rep(0,99)  
for (j in 1:M4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 severity observations  
 j1 <- M1+M2+M3+4\*j-3  
 j2 <- M1+M2+M3+4\*j-2  
 j3 <- M1+M2+M3+4\*j-1  
 j4 <- M1+M2+M3+4\*j  
 Cpmsens24 <- Cpmsens24 + psi1ftn( (1-e)+e\*qc2(1)/pms(1) /  
 (1-e+e\*rCm\_q4(c[j1],mu[j1],theta2,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens24[i] <- igpsens24[i] + mean(psi1ftn(igpdelta2))/M   
 }  
}  
  
igpsens25 <- rep(0,99)  
Cpmsens25 <- rep(0,99)  
for (j in 1:M5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 severity observations  
 j1 <- M1+M2+M3+M4+5\*j-4  
 j2 <- M1+M2+M3+M4+5\*j-3  
 j3 <- M1+M2+M3+M4+5\*j-2  
 j4 <- M1+M2+M3+M4+5\*j-1  
 j5 <- M1+M2+M3+M4+5\*j  
 Cpmsens25 <- Cpmsens25 + psi1ftn( (1-e)+e\*qc2(1)/pms(1) /   
 (1-e+e\*rCm\_q5(c[j1],mu[j1],theta2,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4],c[j5],mu[j5],psi[j5])))/M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]+psi[j5]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4]+psi[j5]\*c[j5]/mu[j5])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta2 <- (1-err)+err\*qc2(thetap)/igpi(thetap)  
 igpdelta2 <- igpdelta2/mean(igpdelta2)  
 igpsens25[i] <- igpsens25[i] + mean(psi1ftn(igpdelta2))/M   
 }  
}  
  
w <- length(unique(trainp$PolicyNum))/length(unique(train$PolicyNum))  
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Cpmsens2 <- (1-w)\*Cpmsens20 + (Cpmsens21+Cpmsens22+Cpmsens23+Cpmsens24+Cpmsens25)\*w   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level   
igpsens2 <- (1-w)\*igpsens20 + (igpsens21+igpsens22+igpsens23+igpsens24+igpsens25)\*w   
  
plot(e,Cpmsens2,type='l',col="blue")  
lines(e,igpsens2)

#### sensitivity for severity with Normal ####  
set.seed(108)  
theta3 <- rnorm(1000,1,sdc/2)  
# MC samples of theta\_C are generated from normal distribution  
  
igpsens30 <- rep(0,99)  
Cpmsens30 <- rep(0,99)  
  
# For the data points without severity, sensitivity is measured only with prior  
Cpmsens30 <- psi1ftn( (1-e)+e\*qc3(1)/pms(1) )  
thetap <- 1/rgamma(1000,k+1,k)  
for (i in 1:99) { # Contribution to the sensitivity for every perturbation level for the policyholders with 0 severity observation  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens30[i] <- igpsens30[i] + mean(psi1ftn(igpdelta3))  
}  
  
  
igpsens31 <- rep(0,99)  
Cpmsens31 <- rep(0,99)  
for (j in 1:M1) { # Contribution to the sensitivity for every perturbation level for the policyholders with 1 severity observation  
   
 # use Lemma 1 directly for the sensitivity of naive prior - for every perturbation level  
 Cpmsens31 <- Cpmsens31 + psi1ftn( (1-e)+e\*qc3(1)/pms(1) / (1-e+e\*rCm\_q1(c[j],mu[j],theta3,psi[j])) ) /M  
   
 # Equation (9) is used to evaluate Bregman divergence for the proposed prior (theta\_C are generated from the proposed posterior)  
 thetap <- 1/rgamma(1000,k+1+psi[j],k+psi[j]\*c[j]/mu[j])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens31[i] <- igpsens31[i] + mean(psi1ftn(igpdelta3))/M   
 }  
}  
  
igpsens32 <- rep(0,99)  
Cpmsens32 <- rep(0,99)  
for (j in 1:M2/2) { # Contribution to the sensitivity for every perturbation level for the policyholders with 2 severity observations  
 j1 <- M1+2\*j-1  
 j2 <- M1+2\*j  
   
 Cpmsens32 <- Cpmsens32 + psi1ftn( (1-e)+e\*qc3(1)/pms(1) /   
 (1-e+e\*rCm\_q2(c[j1],mu[j1],theta3,psi[j1],c[j2],mu[j2],psi[j2])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2],k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens32[i] <- igpsens32[i] + mean(psi1ftn(igpdelta3))/M   
 }  
}  
igpsens33 <- rep(0,99)  
Cpmsens33 <- rep(0,99)  
for (j in 1:M3/3) { # Contribution to the sensitivity for every perturbation level for the policyholders with 3 severity observations  
 j1 <- M1+M2+3\*j-2  
 j2 <- M1+M2+3\*j-1  
 j3 <- M1+M2+3\*j  
 Cpmsens33 <- Cpmsens33 + psi1ftn( (1-e)+e\*qc3(1)/pms(1) /   
 (1-e+e\*rCm\_q3(c[j1],mu[j1],theta3,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens33[i] <- igpsens33[i] + mean(psi1ftn(igpdelta3))/M   
 }  
}  
  
igpsens34 <- rep(0,99)  
Cpmsens34 <- rep(0,99)  
for (j in 1:M4/4) { # Contribution to the sensitivity for every perturbation level for the policyholders with 4 severity observations  
 j1 <- M1+M2+M3+4\*j-3  
 j2 <- M1+M2+M3+4\*j-2  
 j3 <- M1+M2+M3+4\*j-1  
 j4 <- M1+M2+M3+4\*j  
 Cpmsens34 <- Cpmsens34 + psi1ftn( (1-e)+e\*qc3(1)/pms(1) /   
 (1-e+e\*rCm\_q4(c[j1],mu[j1],theta3,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4])) ) /M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens34[i] <- igpsens34[i] + mean(psi1ftn(igpdelta3))/M   
 }  
}  
  
igpsens35 <- rep(0,99)  
Cpmsens35 <- rep(0,99)  
for (j in 1:M5/5) { # Contribution to the sensitivity for every perturbation level for the policyholders with 5 severity observations  
 j1 <- M1+M2+M3+M4+5\*j-4  
 j2 <- M1+M2+M3+M4+5\*j-3  
 j3 <- M1+M2+M3+M4+5\*j-2  
 j4 <- M1+M2+M3+M4+5\*j-1  
 j5 <- M1+M2+M3+M4+5\*j  
 Cpmsens35 <- Cpmsens35 + psi1ftn( (1-e)+e\*qc3(1)/pms(1) /   
 (1-e+e\*rCm\_q5(c[j1],mu[j1],theta3,psi[j1],c[j2],mu[j2],psi[j2],c[j3],mu[j3],psi[j3],c[j4],mu[j4],psi[j4],c[j5],mu[j5],psi[j5])))/M  
 thetap <- 1/rgamma(1000,k+1+psi[j1]+psi[j2]+psi[j3]+psi[j4]+psi[j5]  
 ,k+psi[j1]\*c[j1]/mu[j1]+psi[j2]\*c[j2]/mu[j2]+psi[j3]\*c[j3]/mu[j3]+psi[j4]\*c[j4]/mu[j4]+psi[j5]\*c[j5]/mu[j5])  
 for (i in 1:99) {  
 err <- e[i]  
 igpdelta3 <- (1-err)+err\*qc3(thetap)/igpi(thetap)  
 igpdelta3 <- igpdelta3/mean(igpdelta3)  
 igpsens35[i] <- igpsens35[i] + mean(psi1ftn(igpdelta3))/M   
 }  
}  
  
w <- length(unique(trainp$PolicyNum))/length(unique(train$PolicyNum))  
# Sum of contributions to the sensitivity of naive prior from all observations for every perturbation level  
Cpmsens3 <- (1-w)\*Cpmsens30 + (Cpmsens31+Cpmsens32+Cpmsens33+Cpmsens34+Cpmsens35)\*w   
# Sum of contributions to the sensitivity of proposed prior from all observations for every perturbation level  
igpsens3 <- (1-w)\*igpsens30 + (igpsens31+igpsens32+igpsens33+igpsens34+igpsens35)\*w   
plot(e,Cpmsens3,type='l',col="blue")  
lines(e,igpsens3)

#### Plots for sensitivities ####  
  
par(mfrow=c(1,3),mar=c(4, 4, 2, 1) + 0.1,mgp=c(2.4, 1, 0))  
plot(e,Npmsens1,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Freq Uniform perturbation")  
lines(e,gpsens1)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))  
  
plot(e,Npmsens2,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Freq Lognormal perturbation")  
lines(e,gpsens2)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))  
  
plot(e,Npmsens3,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Freq Normal perturbation")  
lines(e,gpsens3)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))

par(mfrow=c(1,3),mar=c(4, 4, 2, 1) + 0.1,mgp=c(2.4, 1, 0))  
plot(e,Cpmsens1,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Sev Uniform perturbation")  
lines(e,igpsens1)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))  
  
plot(e,Cpmsens2,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Sev Lognormal perturbation")  
lines(e,igpsens2)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))  
  
plot(e,Cpmsens3,type='l',col="blue",ylab="Sensitivity",xlab="perturbation",  
 main="Sev Normal perturbation")  
lines(e,igpsens3)  
legend("topleft",   
 legend = c("Naive Prior", "Proposed Prior"),   
 col = c("blue","black"), lty = c(1,1),  
 cex = 1, bty = "n", text.col = "black",   
 horiz = F , inset = c(0.02, 0.02))