#setwd("C:\\Users\\User\\Google Drive\\Winter 2016\\STAT 531 - Time Series Analysis\\Final Project\\R Code")

#load libraries

require(ggplot2)

require(plyr)

require(reshape2)

require(magrittr)

require(pomp)

require(foreach)

set.seed(123456)

delhi <- read.csv("delhi.csv",head=T)

##---plot data----------------------------------------------------

month <- seq(1,72,1)

cases <- dat$reported

plot(delhi$month,delhi$cases,type='o',

main="Monthly Reported TB Cases in Delhi (2007-2012)",

xlab="Months",ylab="No. of Reported Cases")

##---r process----------------------------------------------------

sir\_step <- Csnippet("

double dN\_SE = rbinom(S,1-exp(-lambda\*I/N\*dt));

double dN\_EI = rbinom(E,1-exp(-gamma\*dt));

double dN\_IQ = rbinom(I,1-exp(-alpha\*dt));

double dN\_QR = rbinom(Q,1-exp(-eta\*dt));

double dN\_SV = rbinom(S,1-exp(-sigma\*dt));

double dN\_VS = rbinom(V,1-exp(-rho\*dt));

double dN\_RS = rbinom(R,1-exp(-epsilon\*dt));

double dN\_IR = rbinom(I,1-exp(-phi\*dt));

S -= dN\_SE - dN\_SV + dN\_VS + dN\_RS;

E += dN\_SE - dN\_EI;

I += dN\_EI - dN\_IQ - dN\_IR;

Q += dN\_IQ - dN\_QR;

R += dN\_QR - dN\_RS + dN\_IR;

V += dN\_SV - dN\_VS;

H += dN\_EI;

")

##---initialize variables in r process----------------------------

sir\_init <- Csnippet("

S = N-1;

E = 1;

I = 0;

Q = 0;

R = 0;

V = 0;

H = 0;

")

##---create pomp object

seirqv <- pomp(delhi,times = "month", t0=1,

rprocess=euler.sim(sir\_step,delta.t=1/2),

initializer=sir\_init,

paramnames=c("lambda","gamma","alpha","eta",

"sigma","rho","epsilon","phi","N"),

statenames=c("S","E","I","Q","R","V","H"))

##---add dmeasure and r measure into pomp object------------------

dmeas <- Csnippet("lik = dbinom(cases,H,psi,give\_log);")

rmeas <- Csnippet("cases = rbinom(H,psi);")

seirqv <- pomp(seirqv,rmeasure=rmeas,dmeasure=dmeas,

statenames="H",zeronames="H",paramnames="psi")

##---simulate from pomp model we created--------------------------

sims <- simulate(seirqv,

params=c(N=64000,psi=0.90,

lambda=2,gamma=0.5,alpha=0.3,eta=1,

sigma=1,rho=1,epsilon=1,phi=0.5),

nsim=20,as=TRUE,include=TRUE)

ggplot(sims,aes(x=time,y=cases,group=sim,color=sim=="data"))+

geom\_line()+guides(color=FALSE)

##---simulate and plot the simulated parameters-------------------

simulate(seirqv,params=c(N=64000,psi=0.90,

lambda=2,gamma=0.5,alpha=0.3,eta=1,

sigma=1,rho=1,epsilon=1,phi=0.5),

nsim=10000,states=TRUE) -> x

matplot(time(seirqv),t(x["H",1:50,]),type='l',lty=1,

xlab="time",ylab="H",bty='l',col='blue')

lines(time(seirqv),obs(seirqv,"cases"),lwd=2,col='black')

matplot(time(seirqv),t(x["psi",1:50,]),type='l',lty=1,

xlab="time",ylab="psi",bty='l',col='blue')

lines(time(seirqv),obs(seirqv,"cases"),lwd=2,col='black')

##---estimate parameters of pomp model----------------------------

cores <- 2 # The number of cores on this machine

registerDoParallel(cores)

mcopts <- list(set.seed=TRUE)

delhi\_box <- rbind(

psi=c(0.1,0.90),

lambda=c(0.5,3),

gamma=c(0.1,0.90),

alpha=c(0.1,0.90),

phi=c(0.1,0.90),

epsilon = c(0.1,0.90),

eta=c(0.1,0.90)

)

delhi\_rw.sd <- 0.02

run\_level <- 1

switch(run\_level,

{delhi\_Np=100; delhi\_Nmif=10; delhi\_Neval=10; delhi\_Nglobal=10; delhi\_Nlocal=10},

{delhi\_Np=20000; delhi\_Nmif=100; delhi\_Neval=10; delhi\_Nglobal=10; delhi\_Nlocal=10},

{delhi\_Np=60000; delhi\_Nmif=300; delhi\_Neval=10; delhi\_Nglobal=100; delhi\_Nlocal=20},

{delhi\_Np=50000; delhi\_Nmif=280; delhi\_Neval=10; delhi\_Nglobal=100; delhi\_Nlocal=20}

)

delhi\_cooling.fraction.50 <- 0.6

delhi\_fixed\_params <- c(N=64000,sigma=1,rho=1)

stew(file=sprintf("box\_eval-%d.rda",run\_level),{

t\_global <- system.time({

mifs\_global <- foreach(i=1:delhi\_Nglobal,.packages='pomp',

.combine=c, .options.multicore=mcopts) %dopar%

mif2(

seirqv,

start=c(apply(delhi\_box,1,function(x)runif(1,x[1],x[2])),delhi\_fixed\_params),

Np=delhi\_Np,

Nmif=delhi\_Nmif,

cooling.type="geometric",

cooling.fraction.50=delhi\_cooling.fraction.50,

transform=TRUE,

rw.sd=rw.sd(

psi=delhi\_rw.sd,

lambda=delhi\_rw.sd,

gamma=delhi\_rw.sd,

alpha=delhi\_rw.sd,

phi=delhi\_rw.sd,

epsilon = delhi\_rw.sd,

eta=delhi\_rw.sd

)

)

})

},seed=1270401374,kind="L'Ecuyer")

stew(file=sprintf("lik\_global\_eval-%d.rda",run\_level),{

t\_global\_eval <- system.time({

liks\_global <- foreach(i=1:bsflu\_Nglobal,.packages='pomp',.combine=rbind, .options.multicore=mcopts) %dopar% {

evals <- replicate(delhi\_Neval, logLik(pfilter(seirqv,params=coef(mifs\_global[[i]]),Np=delhi\_Np)))

logmeanexp(evals, se=TRUE)

}

})

},seed=442141592,kind="L'Ecuyer")