A spatio-temporal hierarchical Markov switching model for the early detection of influenza outbreaks. Supplemental Material, Appendix 1.

This appendix shows the WinBUGS code for the proposed model.

```
model{

  # Previous years
  ################################

  for (i in 1:nreg){
    for (j in 1:nhist){
      Y[1,i,j] ~ dnorm(R[1,i,j,Z[1,i,j]],tau[1,i,j])
      tau[1,i,j] <- pow(sigma[Z[1,i,j]],-2)*
                    (1-(Z[1,i,j]-1)*rho*rho)
      R[1,i,j,1] <- mu0[1,j-1]*sigmamu[1]
      R[1,i,j,2] <- mu1[1,j-1]*sigmamu[2]+
                    psi[1,j-1,i]*sigmapsi
    }
  }

  for (t in 2:weeks){
    for (i in 1:nreg){
      for (j in 1:nhist){
        Y[t,i,j]  ~  dnorm(R[t,i,j,Z[t,i,j]],tau[t,i,j])
        tau[t,i,j] <- pow(sigma[Z[t,i,j]],-2)
        R[t,i,j,1] <- mu0[t,j-1]*sigmamu[1]
        R[t,i,j,2] <- rho*Y[t-1,i,j]+mu1[t,j-1]*sigmamu[2]+
                      psi[t,j-1,i]*sigmapsi
      }
    }
  }

  # Spatio-temporal structure
  for(t in 1:weeks){
    for(j in 1:nhist){
```

```
    mu0[t,j] ~ dnorm(0,1)
    mu1[t,j] ~ dnorm(0,1)
    psi[t,j,1:nreg] ~ car.normal(adj[],weights[],num[],1)
  }
}

# Markovian modeling
for (t in 2:weeks){
  for (i in 1:nreg){
    for (j in 1:nhist){
      Z[t,i,j] ~ dcat(P.mat[Z[t-1,i,j], ])
    }
  }
}

for (i in 1:nreg){
  for (j in 1:nhist){
    Z[1,i,j] ~ dcat(P0[])
  }
}

# Prior distributions of parameters
rho ~ dunif(0,1)

sigmamu[1] <- sigma[1]*lambda
sigmamu[2] <- sigma[2]*lambda

lambda ~ dunif(g,h)

sigmapsi ~ dunif(e,f)

sigma[1] <- ranked(theta[],1)
sigma[2] <- ranked(theta[],2)

for(k in 1:2){
  theta[k] ~ dunif(a,b)
}

P0[1] ~ dbeta(0.5,0.5)
P0[2] <-1-P0[1]
P.mat[1,1] ~ dbeta(0.5,0.5)
P.mat[2,2] ~ dbeta(0.5,0.5)
P.mat[1,2] <- 1-P.mat[1,1]
P.mat[2,1] <- 1-P.mat[2,2]
```

```
# Current year
#############

for (i in 1:nreg){
  Y[1,i,nhist+1] ~ dnorm(R[1,i,nhist+1,Z[1,i,nhist+1]],
                         tau[1,i,nhist+1])
  tau[1,i,nhist+1] <- pow(sigma[Z[1,i,nhist+1]],-2)*
                      (1-(Z[1,i,nhist+1]-1)*rho*rho)
  R[1,i,nhist+1,1] <- munew0[1]*sigmamu[1]
  R[1,i,nhist+1,2] <- munew1[1]*sigmamu[2]+
                      psinew[1,i]*sigmapsi
}

for (t in 2:nobs.new){
  for (i in 1:nreg){
    Y[t,i,nhist+1]  ~  dnorm(R[t,i,nhist+1,Z[t,i,nhist+1]],
                             tau[t,i,nhist+1])
    tau[t,i,nhist+1]<- pow(sigma[Z[t,i,nhist+1]],-2)
    R[t,i,nhist+1,1]<- munew0[t]*sigmamu[1]
    R[t,i,nhist+1,2]<- rho*Y[t-1,i,nhist+1] +
                       munew1[t]*sigmamu[2] +
                       psinew[t,i]*sigmapsi
  }
}

# Spatio-temporal structure
for(t in 1:nobs.new){
  munew0[t] ~ dnorm(0,1)
  munew1[t] ~ dnorm(0,1)
  psinew[t,1:nreg] ~ car.normal(adj[],weights[],num[],1)
}

# Markovian modeling
for (t in 2:nobs.new){
  for (i in 1:nreg){
    Z[t,i,nhist+1] ~ dcat(P.mat[Z[t-1,i,nhist+1], ])
  }
}

for (i in 1:nreg){
  Z[1,i,nhist+1] ~ dcat(P0[])
}

for (i in 1:nreg){
  Zlast[i]<-Z[nobs.new,i,nhist+1]-1
}
```

```
  for (t in 1:nobs.new){
    for (i in 1:nreg){
      Zlastarray[t,i]<-Z[t,i,nhist+1]-1
    }
  }

}
```