Kalman filter implementation 2

Andreas C Charitos[andch552]

11/28/2019

Kalman filter implementation 2

```
# https://github.com/nickpoison/astsa/blob/master/R/Kfilter0.R

KalmanFilter <-
  function(num,y,A,mu0,Sigma0,Phi,cQ,cR){
    #
    # NOTE: must give cholesky decomp: cQ=chol(Q), cR=chol(R)
    Q=t(cQ)%*%cQ
    R=t(cR)%*%cR
    # y is num by q  (time=row series=col)
    # A is a q by p matrix
    # R is q by q
    # mu0 is p by 1
    # Sigma0, Phi, Q are p by p
    Phi=as.matrix(Phi)
    pdim=nrow(Phi)
    y=as.matrix(y)
    qdim=ncol(y)

    xp=array(NA, dim=c(pdim,1,num))          # xp=x_t^{t-1}
    Pp=array(NA, dim=c(pdim,pdim,num))       # Pp=P_t^{t-1}
    xf=array(NA, dim=c(pdim,1,num))          # xf=x_t^t
    Pf=array(NA, dim=c(pdim,pdim,num))       # Pf=x_t^t
    innov=array(NA, dim=c(qdim,1,num))       # innovations
    sig=array(NA, dim=c(qdim,qdim,num))      # innov var-cov matrix

    # initialize (because R can't count from zero)
    x00=as.matrix(mu0, nrow=pdim, ncol=1)
    P00=as.matrix(Sigma0, nrow=pdim, ncol=pdim)
    xp[,,1]=Phi%*%x00
    Pp[,,1]=Phi%*%P00%*%t(Phi)+Q
    sigtemp=A%*%Pp[,,1]%*%t(A)+R
    sig[,,1]=(t(sigtemp)+sigtemp)/2      # innov var - make sure it's symmetric
    siginv=solve(sig[,,1])
    K=Pp[,,1]%*%t(A)%*%siginv
    innov[,,1]=y[1,]-A%*%xp[,,1]
    xf[,,1]=xp[,,1]+K%*%innov[,,1]
    Pf[,,1]=Pp[,,1]-K%*%A%*%Pp[,,1]

    ########## start filter iterations ###################
    for (i in 2:num){
      if (num < 2) break
      xp[,,i]=Phi%*%xf[,,i-1]
      Pp[,,i]=Phi%*%Pf[,,i-1]%*%t(Phi)+Q
      sigtemp=A%*%Pp[,,i]%*%t(A)+R
```

```
    sig[,,i]=(t(sigtemp)+sigtemp)/2      # innov var - make sure it's symmetric
    siginv=solve(sig[,,i])
    K=Pp[,,i]%*%t(A)%*%siginv
    innov[,,i]=y[i,]-A%*%xp[,,i]
    xf[,,i]=xp[,,i]+K%*%innov[,,i]
    Pf[,,i]=Pp[,,i]-K%*%A%*%Pp[,,i]
  }
  return(list(xf=xf,Pf=Pf))
}
```