```
#######################################
###      IN-SAMPLE PREDICTION       ###
#######################################
#plots of methods for xom
#for other cvx, ibm, msft, ko, wmt, citi, boa, mcd, and dis
#stocks we have the same R implementation
# the time range for each stocks are as follows
#xom 05/25/2000-05/19/2004
#cvx 10/10/2001-07/23/2004 and 12/13/2004-02/22/2006
#ibm 05/25/2000-05/19/2004
#msft 01/03/2000-12/24/2003
#citi  10/17/2000-03/07/2005
#boa  03/13/2001-12/19/2005
#ko   05/25/2000-05/19/2004
#wmt  05/25/2000-05/19/2004
#mcd  10/17/2000-03/07/2005
#dis  05/25/2000-05/19/2004
xom = read.table("D://summer/REdata/Rn.2.XOM.txt")
colnames = read.table("D://REdata/Rn.2.XOM.colnames.txt")
rownames = read.table("D://REdata/Rn.2.XOM.rownames.txt")
colnames(data)=colnames
rownames[1] #04/09/1997
rownames[2511] #04/02/2007
sp.data = read.table("D://summer/REdata/Rn.2.sp.txt")
dim(sp.data) #[1] 2520  389
colnames.sp = read.table("C://REdata/Rn.2.sp.colnames.txt")
rownames.sp = read.table("C://REdata/Rn.2.sp.rownames.txt")
#errors #791,891,991,1091,1191,1291,1391,1491,1591, 1691
#all four methods
a = 791
b = 797
xom.100 = xom[a:(a+99),]
sp.100 = sp.data[b:(b+99),]
#transform to make one column represent one day
y = t(xom.100)
#get the mean function for the data
mean_y = matrix(rep(0, 1*100),1,100)
 for(i in 1:100){
   mean_y[,i]= mean(y[,i])
 }
x = t(sp.100)
mean_x = matrix(rep(0, 1*100),1,100)
 for(i in 1:100){
   mean_x[,i]= mean(x[,i])
 }
#center data
Y = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
   for(j in 1:389){
     Y[j,i] = y[j,i]-mean_y[,i]
     }
   }
X = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
    for(j in 1:389){
      X[j,i] = x[j,i]-mean_x[,i]
    }
```

```
  }
t = (1:389)/389
K=389
N = 100
#create 99 fourier basis
fbf = create.fourier.basis(rangeval=c(0,389)/389, nbasis=99)
#we can also use 99 B-spline basis
#fbf = create.bspline.basis(rangeval=c(0,389)/389, nbasis=99)
#transfer the matrix into functional objects
XX = data2fd(X, t, fbf)
YY = data2fd(Y, t, fbf)
xx = data2fd(x, t, fbf)
yy = data2fd(y, t, fbf)
#model sfcapm ag=0
#get the moment of estimator for psi
psi1 = sum(inprod(yy,xx))/sum(inprod(xx,xx))
#prediction
Z1 = matrix(rep(0, K*N),K,N)
 for(i in 1:N){
   Z1[,i] = psi1*x[,i]
   }
#calculate the msep
mse1 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
  for(j in 1:N){
     mse1[i,j]= (Z1[i,j]-y[i,j])^2
                }
 }
#model sfcapm ag
#get the moment of estimator for psi
psi2 = sum(inprod(YY,XX))/sum(inprod(XX,XX))
#prediction
Z2 = matrix(rep(0, K*N),K,N)
   for(i in 1:N){
     Z2[,i] = mean_y[,i]+psi2*X[,i]
    }
#calculate the msep
mse2 = matrix(rep(0, K*N),K,N)
for(i in 1:K){
   for(j in 1:N){
       mse2[i,j]= (Z2[i,j]-y[i,j])^2
       }
   }
#model fcapm ag=0
#number of principal components
nharm = 3
#pca
pca1 = pca.fd(yy, nharm, centerfns=FALSE)
#check the variation explained
variability <- cumsum(pca1$varprop*100)
harmonics <- pca1$harmonics
scores <- pca1$scores
values <- pca1$values
scores1 <- scores[1:N,]
pcax1 = pca.fd(xx, nharm, centerfns=FALSE)
variabilityx <- cumsum(pcax1$varprop*100)
harmonicsx <- pcax1$harmonics
```

```
scoresx <- pcax1$scores
valuesx <- pcax1$values
scores2 <- scoresx[1:N,]
Psi <- matrix(rep(0, nharm*nharm), nrow=nharm)
  for (i in 1:nharm){
    for (j in 1:nharm){
        Psi[j,i] <- (scores2[,i]%*%scores1[,j])/(N*(valuesx[i]))
      }
  }
vy <- eval.fd(t,harmonics)
vx <- eval.fd(t,harmonicsx)
F <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
  for (i in 1:K){
    for (j in 1:K){
        for (k in 1:nharm){
                for (m in 1:nharm){
                        F[i,j,k,m] <- Psi[k,m]*vx[i,k]*vy[j,m]
                }
        }
}
}
psi3 <- matrix(rep(0, K*K), nrow=K)
  for (i in 1:K){
      for (j in 1:K){
        psi3[i,j] <- sum(F[i,j,,])
}
}
#in sample prediction
Z3 = matrix(rep(0, K*N),K,N)
  for(i in 1:N){
    Z3[,i] = (psi3/K)%*%x[,i]
    }
#MSEP
mse3 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
   for(j in 1:N){
      mse3[i,j]= (Z3[i,j]-y[i,j])^2
      }
  }
#fcapm ag
pca1c = pca.fd(YY, nharm, centerfns=FALSE)
variabilityc <- cumsum(pca1c$varprop*100)
harmonicsc <- pca1c$harmonics
scoresc <- pca1c$scores
valuesc <- pca1c$values
scores1c <- scoresc[1:N,]
pcax1c = pca.fd(XX, nharm, centerfns=FALSE)
variabilityxc <- cumsum(pcax1c$varprop*100)
harmonicsxc <- pcax1c$harmonics
scoresxc<- pcax1c$scores
valuesxc <- pcax1c$values
scores2c <- scoresxc[1:N,]
Psic <- matrix(rep(0, nharm*nharm), nrow=nharm)
 for (i in 1:nharm){
   for (j in 1:nharm){
        Psic[j,i] <- (scores2c[,i]%*%scores1c[,j])/(N*(valuesxc[i]))
   }
```

```
  }
vyc <- eval.fd(t,harmonicsc)
vxc <- eval.fd(t,harmonicsxc)
Fc <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
  for (j in 1:K){
   for (k in 1:nharm){
        for (m in 1:nharm){
                Fc[i,j,k,m] <- Psic[k,m]*vxc[i,k]*vyc[j,m]
                }
        }
   }
 }
 psi4 <- matrix(rep(0, K*K), nrow=K)
 for (i in 1:K){
        for (j in 1:K){
                psi4[i,j] <- sum(Fc[i,j,,])
        }
 }
 #in sample prediction
 Z4 = matrix(rep(0, K*N),K,N)
 for(i in 1:N){
     Z4[,i] = mean_y[,i]+(psi4/K)%*%X[,i]
 }
 mse4 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
   for(j in 1:N){
       mse4[i,j]= (Z4[i,j]-y[i,j])^2
    }
  }
#FFD
eghat = matrix(rep(0, K*N),K,N)
 for(i in 1:N){
   eghat[,i] = Y[,i]-(1/K)*psi4%*%X[,i]
 }
ee = data2fd(eghat, t, fbf)
pcaee = pca.fd(ee, nharm, centerfns=FALSE)
variabilityee <- cumsum(pcaee$varprop*100)
variabilityee #59.27140 76.38963 83.55706
harmonicsee <- pcaee$harmonics
scoresee <- pcaee$scores
valuesee <- pcaee$values
scores1ee <- scoresee[1:N-1,]
scores2ee <- scoresee[2:N,]
Phiee <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
        for (j in 1:nharm){
                Phiee[j,i] <-
(scores1ee[,i]%*%scores2ee[,j])/((N-1)*(valuesee[i]))
        }
}
vee <- eval.fd(t,harmonicsee)
Fee <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
        for (j in 1:K){
                for (k in 1:nharm){
                        for (m in 1:nharm){
```

```
                                Fee[i,j,k,m] <-
Phiee[k,m]*vee[i,k]*vee[j,m]
                        }
                }
        }
}
phiee <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                phiee[i,j] <- sum(Fee[i,j,,])
        }
}
Y1 = matrix(rep(0, K*N),K,N)
Y1[,1]=Y[,1]
 for( i in 2:N){
   Y1[,i]=Y[,i]-(phiee/K)%*%Y[,i-1]
 }
X1 = matrix(rep(0, K*N),K,N)
X1[,1]=X[,1]
for( i in 2:N){
  X1[,i]=X[,i]-(phiee/K)%*%X[,i-1]
 }
YY1 = data2fd(Y1, t, fbf)
XX1 = data2fd(X1, t, fbf)
pca2 = pca.fd(YY1, nharm, centerfns=FALSE)
harmonics2 <- pca2$harmonics
scores2 <- pca2$scores
values2 <- pca2$values
scores12 <- scores2[1:N,]
pca2x = pca.fd(XX1, nharm, centerfns=FALSE)
harmonics2x <- pca2x$harmonics
scores2x <- pca2x$scores
values2x <- pca2x$values
scores12x <- scores2x[1:N,]
Psi2 <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
        for (j in 1:nharm){
                Psi2[j,i] <-
(scores12[,i]%*%scores12x[,j])/(N*(values2x[i]))
        }
}
v2y <- eval.fd(t,harmonics2)
v2x <- eval.fd(t,harmonics2x)
F2 <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
        for (j in 1:K){
                for (k in 1:nharm){
                        for (m in 1:nharm){
                                F2[i,j,k,m] <- Psi2[k,m]*v2x[i,k]*v2y[j,m]
                        }
                }
        }
}
psi5 <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                psi5[i,j] <- sum(F2[i,j,,])
```

```
        }
}
#make prediction using method2
Z5 = matrix(rep(0, K*N),K,N)
Z5[,1]=y[,1]
for(i in 2:N)
{
Z5[,i]= (psi5/K)%*%X1[,i]+(phiee/K)%*%Y[,i-1]+mean_y[,i]
}
mse5 = matrix(rep(0, K*N),K,N)
for(i in 1:K)
{for(j in 1:N)
{
mse5[i,j]= (Z5[i,j]-y[i,j])^2
}
}
mean(mse1)
mean(mse2)
mean(mse3)
mean(mse4)
mean(mse5)
#SFDE method
#sfde
psi.sfde = sum(inprod(YY,XX))/sum(inprod(XX,XX))
eghat = matrix(rep(0, K*N),K,N)
for(i in 1:N)
{
eghat[,i] = Y[,i]-psi.sfde*X[,i]
}
ee = data2fd(eghat, t, fbf)
pcaee = pca.fd(ee, nharm, centerfns=FALSE)
variabilityee <- cumsum(pcaee$varprop*100)
variabilityee #64.34244 79.78177 86.73400
harmonicsee <- pcaee$harmonics
scoresee <- pcaee$scores
valuesee <- pcaee$values
scores1ee <- scoresee[1:N-1,]
scores2ee <- scoresee[2:N,]
Phiee <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
        for (j in 1:nharm){
                Phiee[j,i] <-
(scores1ee[,i]%*%scores2ee[,j])/((N-1)*(valuesee[i]))
        }
}
vee <- eval.fd(t,harmonicsee)
Fee <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
        for (j in 1:K){
                for (k in 1:nharm){
                        for (m in 1:nharm){
                                Fee[i,j,k,m] <-
Phiee[k,m]*vee[i,k]*vee[j,m]
                        }
                }
        }
}
```

```
phiee <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                phiee[i,j] <- sum(Fee[i,j,,])
        }
}
Y1 = matrix(rep(0, K*N),K,N)
Y1[,1]=Y[,1]
  for( i in 2:N){
   Y1[,i]=Y[,i]-(phiee/K)%*%Y[,i-1]
  }
X1 = matrix(rep(0, K*N),K,N)
X1[,1]=X[,1]
  for( i in 2:N){
     X1[,i]=X[,i]-(phiee/K)%*%X[,i-1]
  }
#get the mean function
mean_Y1 = matrix(rep(0, 1*100),1,100)
for(i in 1:100){
       mean_Y1[,i]= mean(Y1[,i])
  }
mean_X1 = matrix(rep(0, 1*100),1,100)
   for(i in 1:100){
       mean_X1[,i]= mean(X1[,i])
   }
#center data
Y11 = matrix(rep(0, 389*100),389,100)
for(i in 1:100){
      for(j in 1:389){
           Y11[j,i] = Y1[j,i]-mean_Y1[,i]
        }
  }
X11 = matrix(rep(0, 389*100),389,100)
 for(i in 1:100){
     for(j in 1:389){
           X11[j,i] = X1[j,i]-mean_X1[,i]
    }
}
#make them into functional objects
YY1 = data2fd(Y11, t, fbf)
XX1 = data2fd(X11, t, fbf)
psi22 = sum(inprod(YY1,XX1))/sum(inprod(XX1,XX1))
#make prediction for Y*
P_Y1 = matrix(rep(0, K*N), K, N)
for(i in 1:N){
  P_Y1[,i] = mean_Y1[,i]+ psi22*X11[,i]
  }
Z.sfde = matrix(rep(0, K*N),K,N)
Z.sfde[,1] = psi2*X[,1]+mean_y[,1]
 for(i in 2:N){
   Z.sfde[,i] = P_Y1[,i]+(phiee/K)%*%Y[,i-1]+mean_y[,i]
 }
#make prediction using method2
#Z.sfde = matrix(rep(0, K*N),K,N)
#Z.sfde[,1]=y[,1]
#for(i in 2:N)
#{
```

```
#Z.sfde[,i]= psi22*X11[,i]+mean_Y1[,i]+mean_y[,i]+(phiee/K)%*%Y[,i-1]
#}
mse.sfde = matrix(rep(0, K*N),K,N)
  for(i in 1:K){
    for(j in 1:N){
          mse5[i,j]= (Z5[i,j]-y[i,j])^2
          }
     }
mean(mse.sfde)
#######################################
###      OUT-SAMPLE PREDICTION      ###
#######################################
a=1691
b=1697
xom.100 = xom[a:(a+99),]
sp.100 = sp.data[b:(b+99),]
xom1.100 = xom[(a+100):(a+199),]
sp1.100 = sp.data[(b+100):(b+199),]
#transform to make one column represent one day
#in-sample data
y = t(xom.100)
mean_y = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
     mean_y[,i]= mean(y[,i])
   }
x = t(sp.100)
mean_x = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
      mean_x[,i]= mean(x[,i])
   }
##out-sample data
yp = t(xom1.100)
mean_yp = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
    mean_yp[,i]= mean(yp[,i])
    }
xp = t(sp1.100)
mean_xp = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
      mean_xp[,i]= mean(xp[,i])
    }
#in-sample centered
Y = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
        for(j in 1:389){
           Y[j,i] = y[j,i]-mean_y[,i]
         }
   }
X = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
       for(j in 1:389){
            X[j,i] = x[j,i]-mean_x[,i]
         }
    }
##out-sample centered
Yp = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
```

```
      for(j in 1:389){
          Yp[j,i] = yp[j,i]-mean_yp[,i]
        }
   }
Xp = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
      for(j in 1:389){
          Xp[j,i] = xp[j,i]-mean_xp[,i]
         }
     }
t = (1:389)/389
K=389
N = 100
fbf = create.fourier.basis(rangeval=c(0,389)/389, nbasis=99)
XX = data2fd(X, t, fbf)
YY = data2fd(Y, t, fbf)
xx = data2fd(x, t, fbf)
yy = data2fd(y, t, fbf)
#sfcapm ag=0
psi1 = sum(inprod(yy,xx))/sum(inprod(xx,xx))
# out-sample prediction
Z1 = matrix(rep(0, K*N),K,N)
  for(i in 1:N){
      Z1[,i] = psi1*xp[,i]
   }
mse1 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
      for(j in 1:N){
          mse1[i,j]= (Z1[i,j]-yp[i,j])^2
        }
  }
#sfcapm ag
psi2 = sum(inprod(YY,XX))/sum(inprod(XX,XX))
#out-sample prediction
Z2 = matrix(rep(0, K*N),K,N)
  for(i in 1:N){
      Z2[,i] = mean_yp[,i]+psi2*Xp[,i]
      }
mse2 = matrix(rep(0, K*N),K,N)
  for(i in 1:K){
        for(j in 1:N){
             mse2[i,j]= (Z2[i,j]-yp[i,j])^2
        }
   }
#fcapm ag=0
nharm = 3
pca1 = pca.fd(yy, nharm, centerfns=FALSE)
variability <- cumsum(pca1$varprop*100)
harmonics <- pca1$harmonics
scores <- pca1$scores
values <- pca1$values
scores1 <- scores[1:N,]
pcax1 = pca.fd(xx, nharm, centerfns=FALSE)
variabilityx <- cumsum(pcax1$varprop*100)
harmonicsx <- pcax1$harmonics
scoresx <- pcax1$scores
valuesx <- pcax1$values
```

```
scores2 <- scoresx[1:N,]
Psi <- matrix(rep(0, nharm*nharm), nrow=nharm)
 for (i in 1:nharm){
        for (j in 1:nharm){
                Psi[j,i] <- (scores2[,i]%*%scores1[,j])/(N*(valuesx[i]))
        }
 }
vy <- eval.fd(t,harmonics)
vx <- eval.fd(t,harmonicsx)
F <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
 for (i in 1:K){
        for (j in 1:K){
                for (k in 1:nharm){
                        for (m in 1:nharm){
                                F[i,j,k,m] <- Psi[k,m]*vx[i,k]*vy[j,m]
                        }
                }
        }
 }
psi3 <- matrix(rep(0, K*K), nrow=K)
 for (i in 1:K){
        for (j in 1:K){
                psi3[i,j] <- sum(F[i,j,,])
        }
 }
# prediction
Z3 = matrix(rep(0, K*N),K,N)
for(i in 1:N){
 Z3[,i] = (psi3/K)%*%xp[,i]
 }
mse3 = matrix(rep(0, K*N),K,N)
for(i in 1:K){
   for(j in 1:N){
        mse3[i,j]= (Z3[i,j]-yp[i,j])^2
    }
 }
#fcapm ag
pca1c = pca.fd(YY, nharm, centerfns=FALSE)
variabilityc <- cumsum(pca1c$varprop*100)
harmonicsc <- pca1c$harmonics
scoresc <- pca1c$scores
valuesc <- pca1c$values
scores1c <- scoresc[1:N,]
pcax1c = pca.fd(XX, nharm, centerfns=FALSE)
variabilityxc <- cumsum(pcax1c$varprop*100)
harmonicsxc <- pcax1c$harmonics
scoresxc<- pcax1c$scores
valuesxc <- pcax1c$values
scores2c <- scoresxc[1:N,]
Psic <- matrix(rep(0, nharm*nharm), nrow=nharm)
 for (i in 1:nharm){
        for (j in 1:nharm){
                Psic[j,i] <-
(scores2c[,i]%*%scores1c[,j])/(N*(valuesxc[i]))
        }
 }
vyc <- eval.fd(t,harmonicsc)
```

```
vxc <- eval.fd(t,harmonicsxc)
Fc <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
	for (j in 1:K){
		for (k in 1:nharm){
			for (m in 1:nharm){
				Fc[i,j,k,m] <- Psic[k,m]*vxc[i,k]*vyc[j,m]
			}
		}
	}
 }
psi4 <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
	for (j in 1:K){
		psi4[i,j] <- sum(Fc[i,j,,])
	}
 }
#out sample prediction
Z4 = matrix(rep(0, K*N),K,N)
for(i in 1:N){
     Z4[,i] = mean_yp[,i]+(psi4/K)%*%Xp[,i]
   }
mse4 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
     for(j in 1:N){
         mse4[i,j]= (Z4[i,j]-yp[i,j])^2
        }
    }
#FFD
eghat = matrix(rep(0, K*N),K,N)
 for(i in 1:N){
    eghat[,i] = Y[,i]-(1/K)*psi4%*%X[,i]
    }
ee = data2fd(eghat, t, fbf)
pcaee = pca.fd(ee, nharm, centerfns=FALSE)
variabilityee <- cumsum(pcaee$varprop*100)
variabilityee #59.27140 76.38963 83.55706
harmonicsee <- pcaee$harmonics
scoresee <- pcaee$scores
valuesee <- pcaee$values
scores1ee <- scoresee[1:N-1,]
scores2ee <- scoresee[2:N,]
Phiee <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
	for (j in 1:nharm){
		Phiee[j,i] <-
(scores1ee[,i]%*%scores2ee[,j])/((N-1)*(valuesee[i]))
	}
}
vee <- eval.fd(t,harmonicsee)
Fee <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
	for (j in 1:K){
		for (k in 1:nharm){
			for (m in 1:nharm){
				Fee[i,j,k,m] <-
Phiee[k,m]*vee[i,k]*vee[j,m]
```

```
                        }
                }
        }
}
phiee <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                phiee[i,j] <- sum(Fee[i,j,,])
        }
}
Y1 = matrix(rep(0, K*N),K,N)
Y1[,1]=Y[,1]
for( i in 2:N){
   Y1[,i]=Y[,i]-(phiee/K)%*%Y[,i-1]
  }
X1 = matrix(rep(0, K*N),K,N)
X1[,1]=X[,1]
for( i in 2:N){
  X1[,i]=X[,i]-(phiee/K)%*%X[,i-1]
  }
YY1 = data2fd(Y1, t, fbf)
XX1 = data2fd(X1, t, fbf)
pca2 = pca.fd(YY1, nharm, centerfns=FALSE)
harmonics2 <- pca2$harmonics
scores2 <- pca2$scores
values2 <- pca2$values
scores12 <- scores2[1:N,]
pca2x = pca.fd(XX1, nharm, centerfns=FALSE)
harmonics2x <- pca2x$harmonics
scores2x <- pca2x$scores
values2x <- pca2x$values
scores12x <- scores2x[1:N,]
Psi2 <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
        for (j in 1:nharm){
                Psi2[j,i] <-
(scores12[,i]%*%scores12x[,j])/(N*(values2x[i]))
        }
}
v2y <- eval.fd(t,harmonics2)
v2x <- eval.fd(t,harmonics2x)
F2 <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
        for (j in 1:K){
                for (k in 1:nharm){
                        for (m in 1:nharm){
                                F2[i,j,k,m] <- Psi2[k,m]*v2x[i,k]*v2y[j,m]
                        }
                }
        }
}
psi5 <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                psi5[i,j] <- sum(F2[i,j,,])
        }
}
```

```
#out-sample sp data centered
Xp1 = matrix(rep(0, K*N),K,N)
Xp1[,1]=Xp[,1]
for( i in 2:N)
{
Xp1[,i]=Xp[,i]-(phiee/K)%*%Xp[,i-1]
}
#make prediction using method2
Z5 = matrix(rep(0, K*N),K,N)
Z5[,1]=y[,1]
for(i in 2:N){
    Z5[,i]= (psi5/K)%*%Xp1[,i]+(phiee/K)%*%Yp[,i-1]+mean_yp[,i]
    }
mse5 = matrix(rep(0, K*N),K,N)
 for(i in 1:K){
    for(j in 1:N){
      mse5[i,j]= (Z5[i,j]-yp[i,j])^2
    }
  }
mean(mse1)
mean(mse2)
mean(mse3)
mean(mse4)
mean(mse5)
#SFDE for out sample prediction
a=1591
b=1597
xom.100 = xom[a:(a+99),]
sp.100 = sp.data[b:(b+99),]
xom1.100 = xom[(a+100):(a+199),]
sp1.100 = sp.data[(b+100):(b+199),]
#transform to make one column represent one day
#in-sample data
y = t(xom.100)
mean_y = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
       mean_y[,i]= mean(y[,i])
      }
x = t(sp.100)
mean_x = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
        mean_x[,i]= mean(x[,i])
    }
##out-sample data
yp = t(xom1.100)
mean_yp = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
       mean_yp[,i]= mean(yp[,i])
     }
xp = t(sp1.100)
mean_xp = matrix(rep(0, 1*100),1,100)
  for(i in 1:100){
     mean_xp[,i]= mean(xp[,i])
   }
#in-sample centered
Y = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
```

```
        for(j in 1:389){
            Y[j,i] = y[j,i]-mean_y[,i]
         }
  }
X = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
        for(j in 1:389){
             X[j,i] = x[j,i]-mean_x[,i]
         }
   }
##out-sample centered
Yp = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
        for(j in 1:389){
             Yp[j,i] = yp[j,i]-mean_yp[,i]
        }
   }
Xp = matrix(rep(0, 389*100),389,100)
  for(i in 1:100){
       for(j in 1:389){
           Xp[j,i] = xp[j,i]-mean_xp[,i]
       }
   }
t = (1:389)/389
K=389
N = 100
fbf = create.fourier.basis(rangeval=c(0,389)/389, nbasis=99)
XX = data2fd(X, t, fbf)
YY = data2fd(Y, t, fbf)
xx = data2fd(x, t, fbf)
yy = data2fd(y, t, fbf)
psi2 = sum(inprod(YY,XX))/sum(inprod(XX,XX))
eghat = matrix(rep(0, K*N),K,N)
for(i in 1:N)
{
eghat[,i] = Y[,i]-psi2*X[,i]
}
ee = data2fd(eghat, t, fbf)
pcaee = pca.fd(ee, nharm, centerfns=FALSE)
variabilityee <- cumsum(pcaee$varprop*100)
variabilityee #64.34244 79.78177 86.73400
harmonicsee <- pcaee$harmonics
scoresee <- pcaee$scores
valuesee <- pcaee$values
scores1ee <- scoresee[1:N-1,]
scores2ee <- scoresee[2:N,]
Phiee <- matrix(rep(0, nharm*nharm), nrow=nharm)
for (i in 1:nharm){
        for (j in 1:nharm){
                Phiee[j,i] <-
(scores1ee[,i]%*%scores2ee[,j])/((N-1)*(valuesee[i]))
        }
}
vee <- eval.fd(t,harmonicsee)
Fee <- array(rep(0,K*K), dim=c(K,K,nharm,nharm))
for (i in 1:K){
        for (j in 1:K){
```

```
                for (k in 1:nharm){
                        for (m in 1:nharm){
                                Fee[i,j,k,m] <-
Phiee[k,m]*vee[i,k]*vee[j,m]
                        }
                }
        }
}
phiee <- matrix(rep(0, K*K), nrow=K)
for (i in 1:K){
        for (j in 1:K){
                phiee[i,j] <- sum(Fee[i,j,,])
        }
}
Y1 = matrix(rep(0, K*N),K,N)
Y1[,1]=Y[,1]
for( i in 2:N)
{
Y1[,i]=Y[,i]-(phiee/K)%*%Y[,i-1]
}
X1 = matrix(rep(0, K*N),K,N)
X1[,1]=X[,1]
for( i in 2:N)
{
X1[,i]=X[,i]-(phiee/K)%*%X[,i-1]
}
mean_Y1 = matrix(rep(0, 1*100),1,100)
for(i in 1:100)
{mean_Y1[,i]= mean(Y1[,i])}
mean_X1 = matrix(rep(0, 1*100),1,100)
for(i in 1:100)
{mean_X1[,i]= mean(X1[,i])}
Y11 = matrix(rep(0, 389*100),389,100)
for(i in 1:100){
     for(j in 1:389){
          Y11[j,i] = Y1[j,i]-mean_Y1[,i]
     }
}
X11 = matrix(rep(0, 389*100),389,100)
for(i in 1:100){
    for(j in 1:389){
         X11[j,i] = X1[j,i]-mean_X1[,i]
     }
}
YY1 = data2fd(Y11, t, fbf)
XX1 = data2fd(X11, t, fbf)
psi22 = sum(inprod(YY1,XX1))/sum(inprod(XX1,XX1))
#out sample data*
Xp1 = matrix(rep(0, K*N),K,N)
Xp1[,1]=Xp[,1]
for( i in 2:N)
{
Xp1[,i]=Xp[,i]-(phiee/K)%*%Xp[,i-1]
}
Yp1 = matrix(rep(0, K*N),K,N)
Yp1[,1]=Yp[,1]
for( i in 2:N)
```

```
{
Yp1[,i]=Yp[,i]-(phiee/K)%*%Yp[,i-1]
}
mean_Yp1 = matrix(rep(0, 1*100),1,100)
for(i in 1:100)
{mean_Yp1[,i]= mean(Yp1[,i])}
mean_Xp1 = matrix(rep(0, 1*100),1,100)
for(i in 1:100)
{mean_Xp1[,i]= mean(Xp1[,i])}
Yp11 = matrix(rep(0, 389*100),389,100)
for(i in 1:100){
    for(j in 1:389){
         Yp11[j,i] = Yp1[j,i]-mean_Yp1[,i]
     }
}
Xp11 = matrix(rep(0, 389*100),389,100)
for(i in 1:100){
   for(j in 1:389){
       Xp11[j,i] = Xp1[j,i]-mean_Xp1[,i]
    }
}
#make prediction for Y*
P_Y1 = matrix(rep(0, K*N), K, N)
for(i in 1:N)
{
P_Y1[,i] = mean_Yp1[,i]+ psi22*Xp11[,i]
}
Z.sfde = matrix(rep(0, K*N),K,N)
Z.sfde[,1] = psi2*Xp[,1]+mean_yp[,1]
for(i in 2:N){
    Z.sfde[,i] = P_Y1[,i]+(phiee/K)%*%Yp[,i-1]+mean_yp[,i]
  }
#make prediction using method2
#Z.sfde = matrix(rep(0, K*N),K,N)
#Z.sfde[,1]=y[,1]
#for(i in 2:N)
#{
#Z.sfde[,i]= psi22*X11[,i]+mean_Y1[,i]+mean_y[,i]+(phiee/K)%*%Y[,i-1]
#}
mse.sfde = matrix(rep(0, K*N),K,N)
for(i in 1:K)
{for(j in 1:N)
{
mse.sfde[i,j]= (Z.sfde[i,j]-yp[i,j])^2
}
}
mean(mse.sfde)
```