STAT547 Homework 2

Xingche Guo

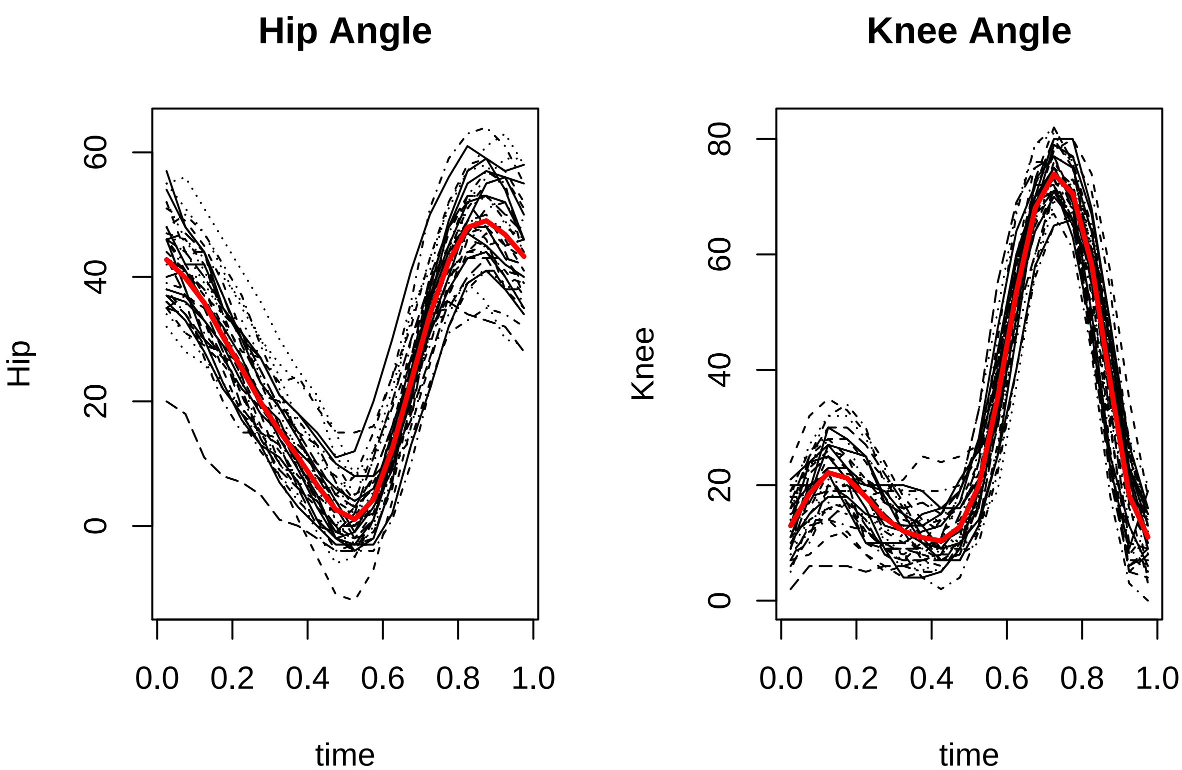
9/23/2019

Problem 1

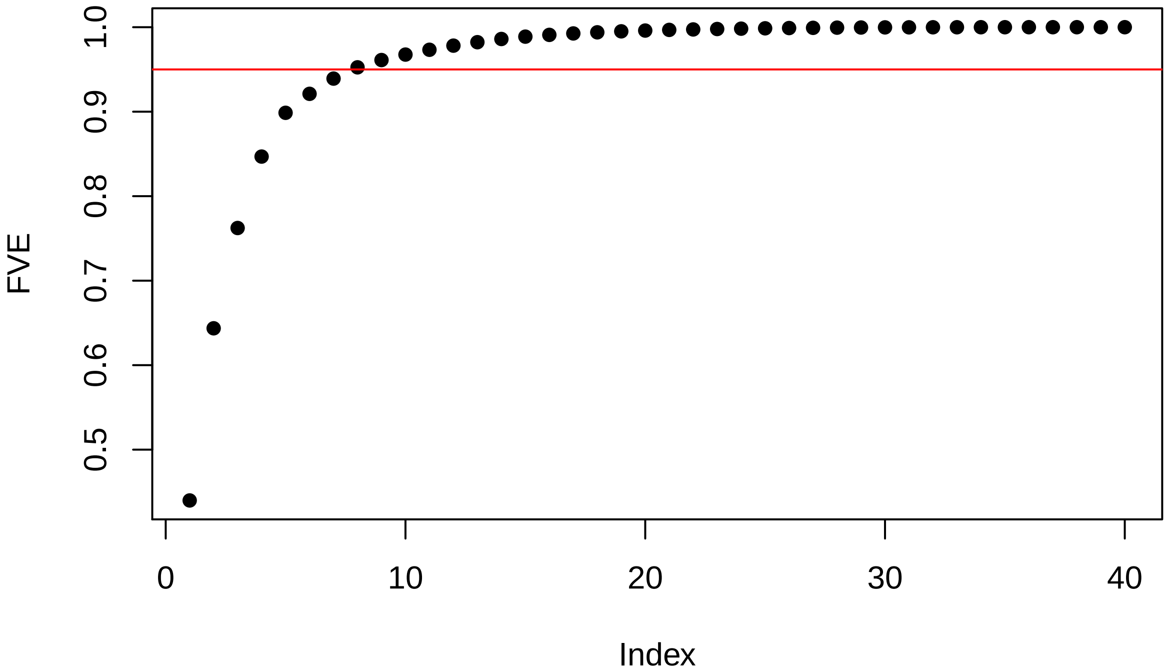
```
library(fda)
```

```
## Loading required package: splines

## Loading required package: Matrix

##
## Attaching package: 'fda'

## The following object is masked from 'package:graphics':
##
##     matplot
```

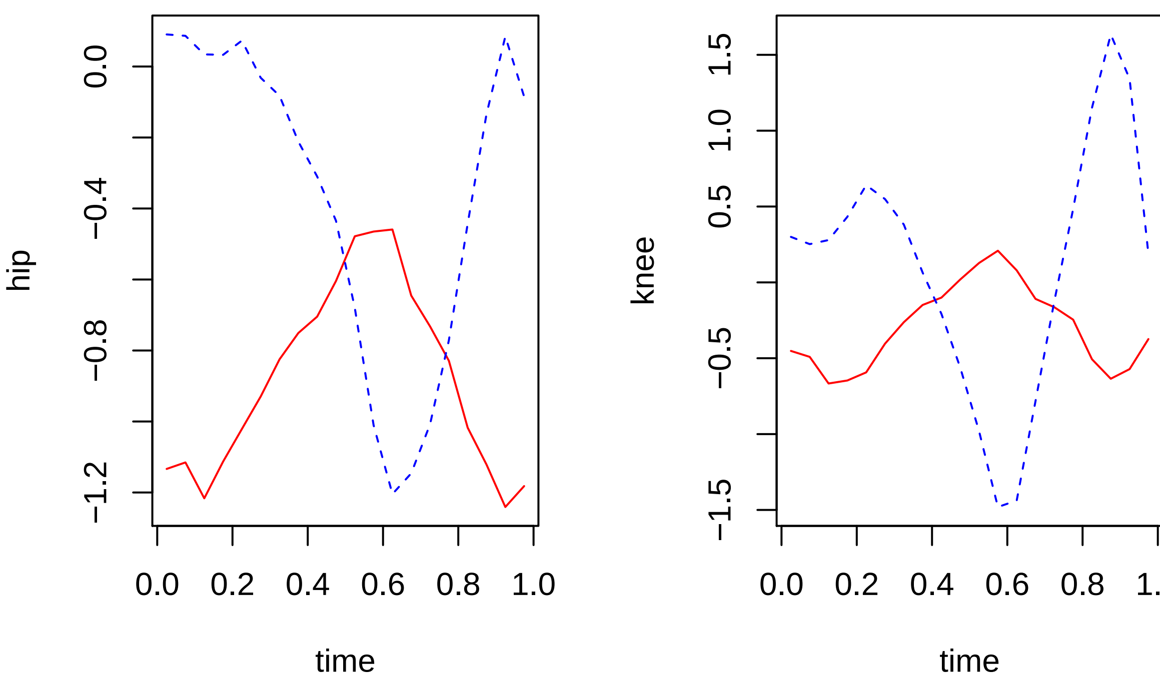
```
library(fdapace)

############### Problem 1 #################
data(gait)
Hip <- gait[, , 1]
Knee <- gait[, , 2]
time <- as.numeric( rownames(Hip) )
n <- ncol(Hip)
m <- nrow(Hip)

XX <- t(rbind(Hip, Knee))
mu <- colMeans(XX)
GG <- cov(XX)

## plot the mean function
par(mfrow = c(1,2))
matplot(time, Hip, type='l', col='black', main = "Hip Angle", xlab = "time")
lines(time, mu[1:m], col = "red", lwd = 2.5)

matplot(time, Knee, type='l', col='black', main = "Knee Angle", xlab = "time")
lines(time, mu[(m+1):(2*m)], col = "red", lwd = 2.5)
```




```
par(mfrow = c(1,1))

## check the total variation
eig <- eigen(GG)
lam <- eig$values / m
FVEeach <- lam / sum(lam)
FVE <- cumsum(FVEeach)

plot(FVE, pch = 16)
abline(h = 0.95, col = "red")
```

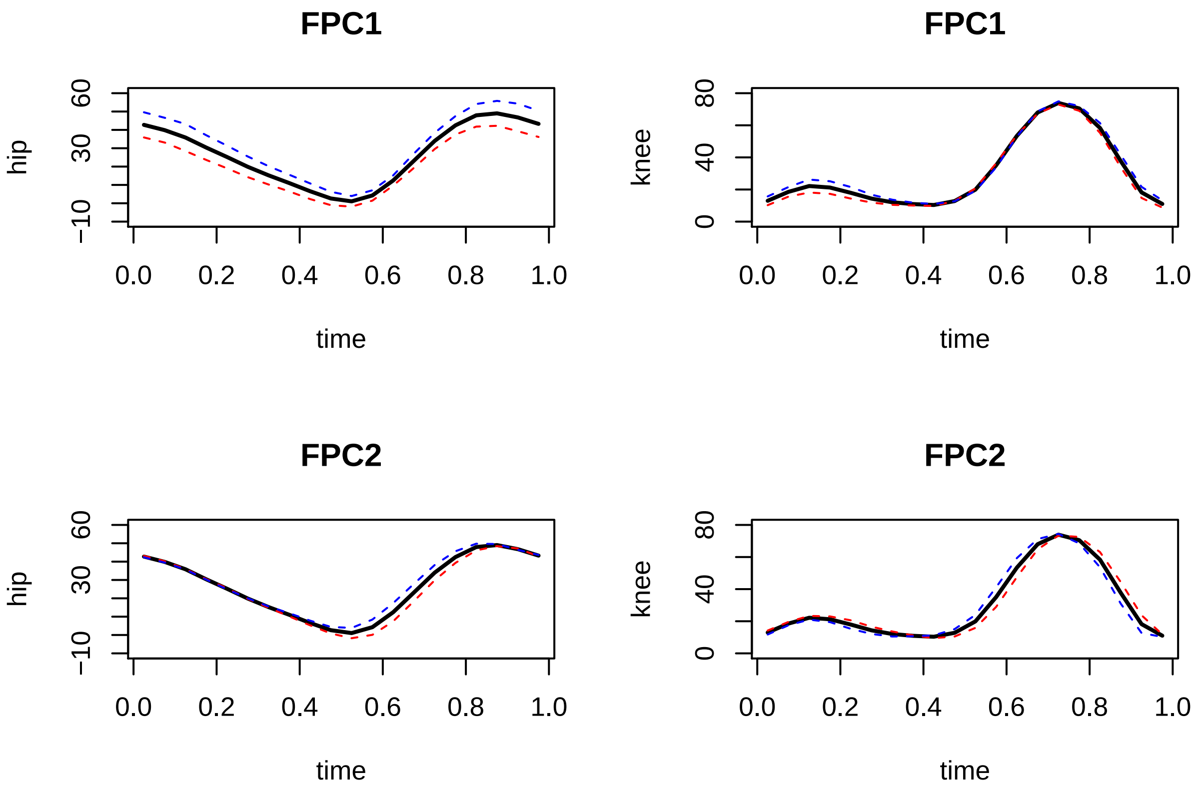



```r
## plot the eigen-functions
phi <- eig$vectors * sqrt(m)
phi1 <- phi[1:m,]
phi2 <- phi[(m+1):(2*m),]

par(mfrow = c(1,2))
matplot(time, phi1[,1:2], type='l', col=c('red', 'blue'), xlab = "time", ylab = "hip")
matplot(time, phi2[,1:2], type='l', col=c('red', 'blue'), xlab = "time", ylab = "knee")
```



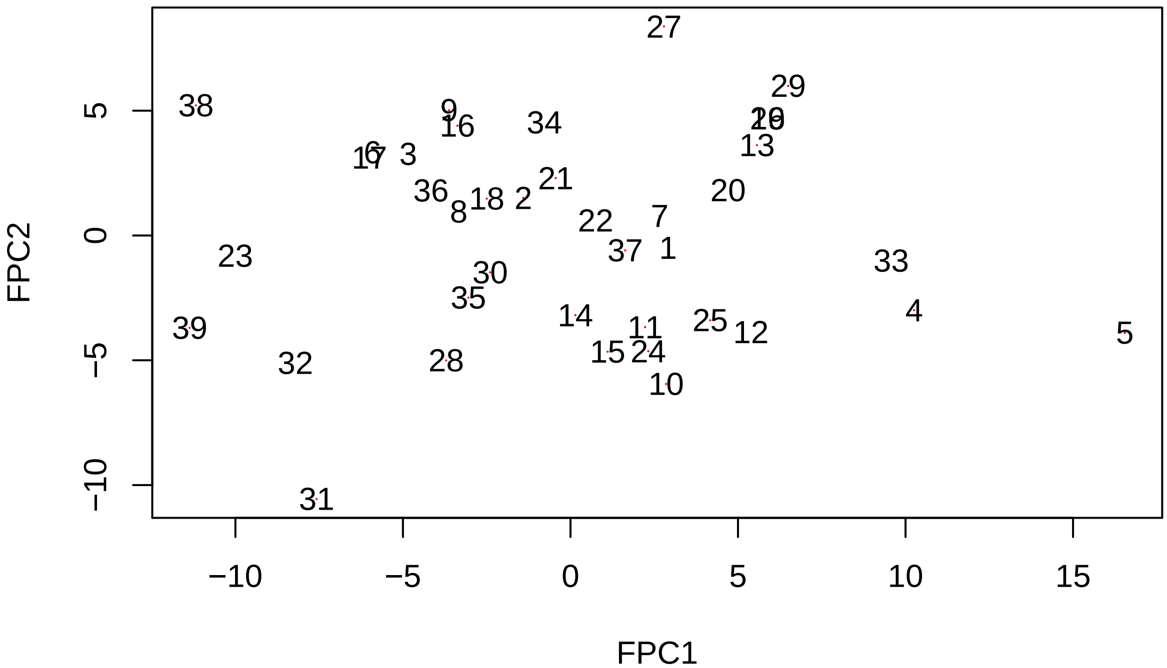

```r
par(mfrow = c(1,1))
```

```r
## Mode of variation plot
par(mfcol = c(2,2))
lamphi11 <- sqrt(lam[1]) * phi1[, 1]
plot(time, mu[1:m], type='l', lwd=2, ylim = c(-10, 60),
     main = "FPC1", xlab = "time", ylab = "hip")
lines(time, mu[1:m] + lamphi11, type='l', lty=2, col='red')
lines(time, mu[1:m] - lamphi11, type='l', lty=2, col='blue')

lamphi12 <- sqrt(lam[2]) * phi1[, 2]
plot(time, mu[1:m], type='l', lwd=2, ylim = c(-10, 60),
     main = "FPC2", xlab = "time", ylab = "hip")
lines(time, mu[1:m] + lamphi12, type='l', lty=2, col='red')
lines(time, mu[1:m] - lamphi12, type='l', lty=2, col='blue')

lamphi21 <- sqrt(lam[1]) * phi2[, 1]
plot(time, mu[(m+1):(2*m)], type='l', lwd=2, ylim = c(-0, 80),
     main = "FPC1", xlab = "time", ylab = "knee")
lines(time, mu[(m+1):(2*m)] + lamphi21, type='l', lty=2, col='red')
lines(time, mu[(m+1):(2*m)] - lamphi21, type='l', lty=2, col='blue')

lamphi22 <- sqrt(lam[2]) * phi2[, 2]
plot(time, mu[(m+1):(2*m)], type='l', lwd=2, ylim = c(-0, 80),
     main = "FPC2", xlab = "time", ylab = "knee")
lines(time, mu[(m+1):(2*m)] + lamphi22, type='l', lty=2, col='red')
lines(time, mu[(m+1):(2*m)] - lamphi22, type='l', lty=2, col='blue')
```



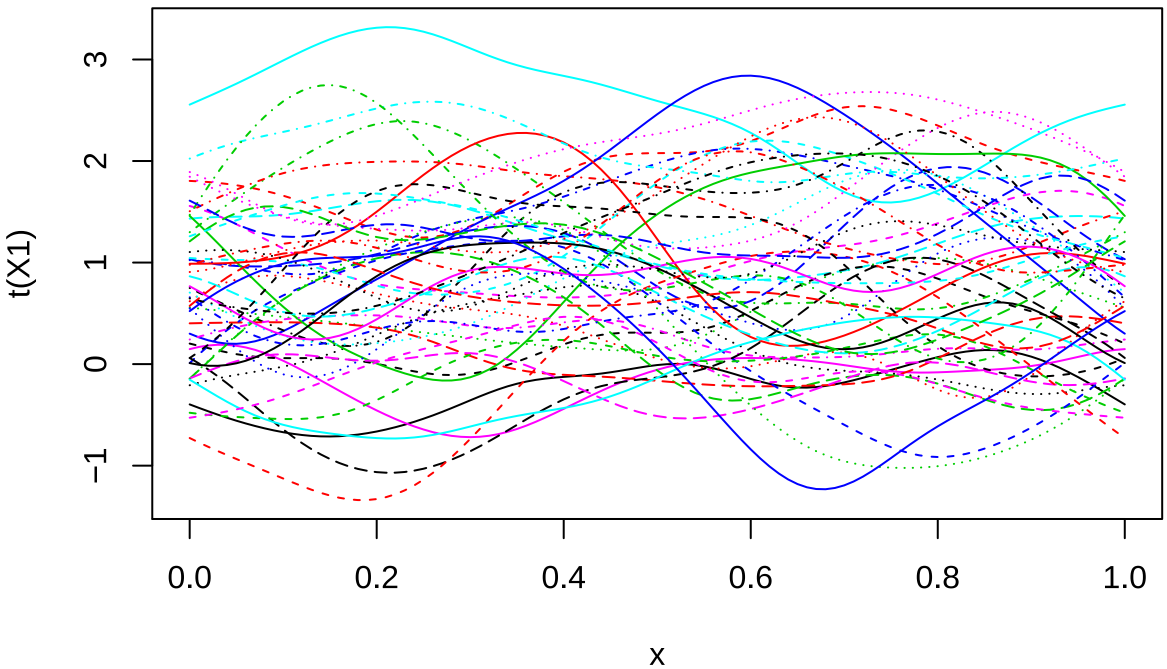

```r
par(mfrow = c(1,1))
```

```r
## FPCs
XXcenter <- XX - matrix( rep(mu, n), nrow=n, ncol=2*m, byrow=TRUE)
xi <- XXcenter %*% phi / m

plot(xi[, 1], xi[, 2], xlab = "FPC1", ylab = "FPC2",
     col = "red", cex = 0.001)
text(xi[, 1], xi[, 2], label= sub("boy", "", rownames(XX)))
```

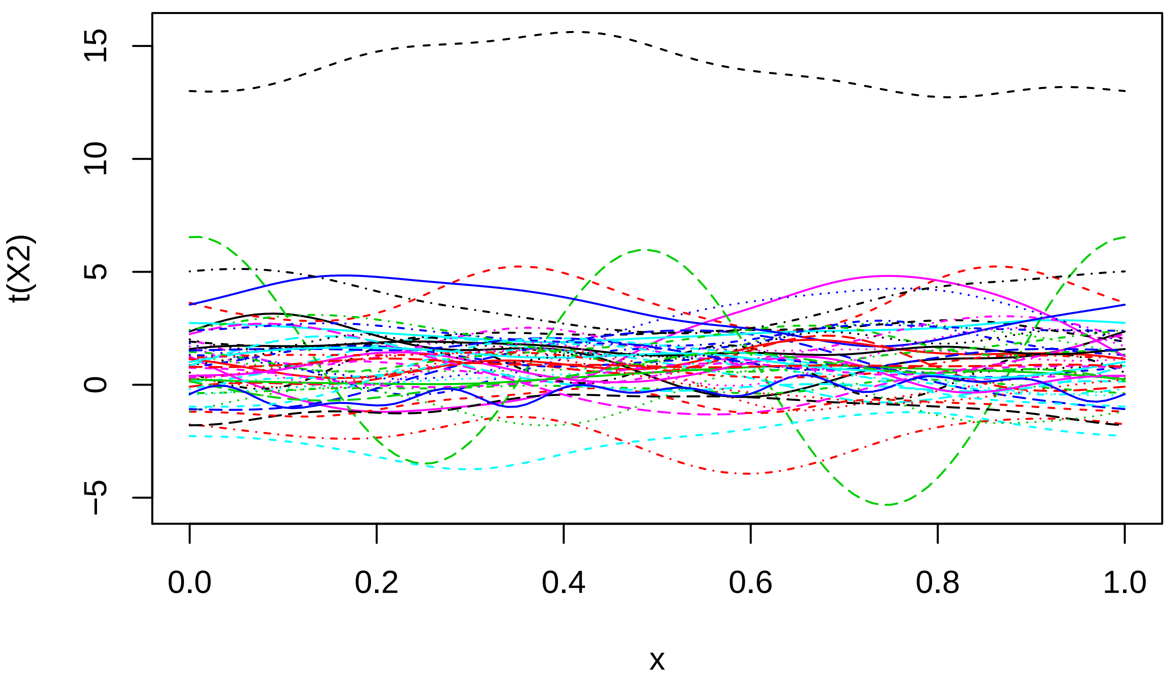


Problem 2

```r
############### Problem 2 #################
K <- 20
n <- 50
m <- 101
t <- seq(0, 1, length.out = m)
mu <- -(t - 0.5)^2 + 1


## Create Basis
phiSim <- CreateBasis(K=K, t, type='fourier')
#matplot(t, phiSim[, 1:5], type='l')
lamSim <- exp(-(1:K))

## Gaussian Process
# rows are for individuals, col are for different FPC
xi <- matrix(rnorm(n * K), nrow = n, ncol = K) %*% diag(sqrt(lamSim))
X1 <- matrix(mu, nrow=n, ncol=m, byrow=TRUE) + xi %*% t(phiSim)
matplot(t, t(X1), type='l')
```



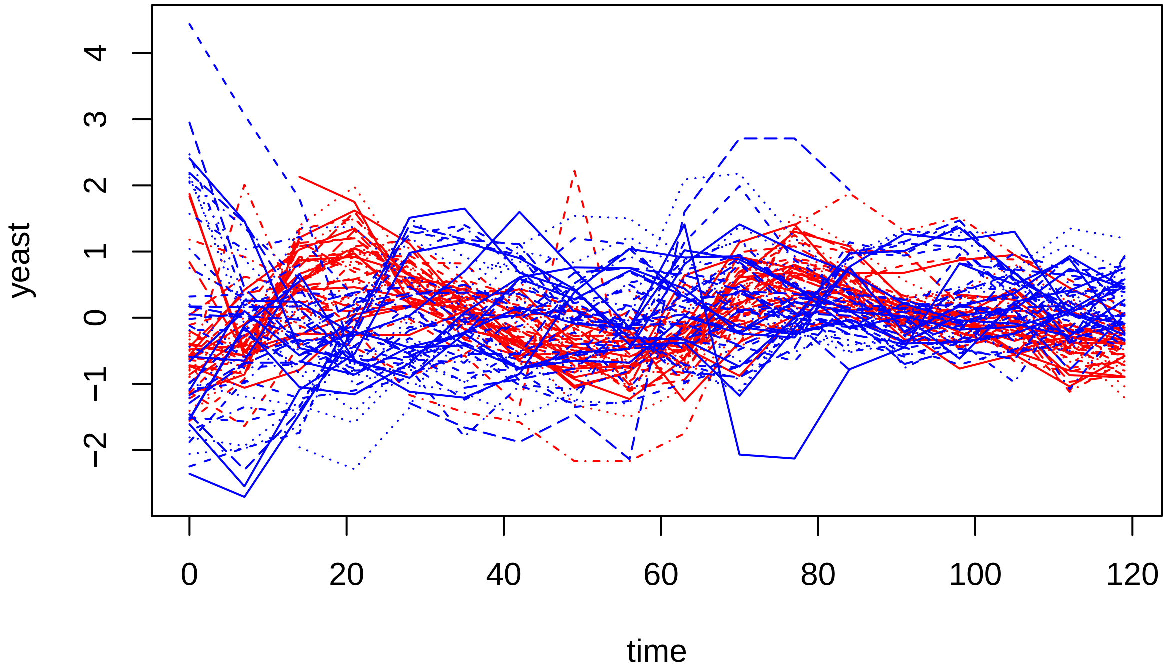

```
## non-Gaussian Process
xi <- matrix(rt(n * K, df = 2), nrow = n, ncol = K) %*% diag(sqrt(lamSim))
X2 <- matrix(mu, nrow=n, ncol=m, byrow=TRUE) + xi %*% t(phiSim)
matplot(t, t(X2), type='l')
```



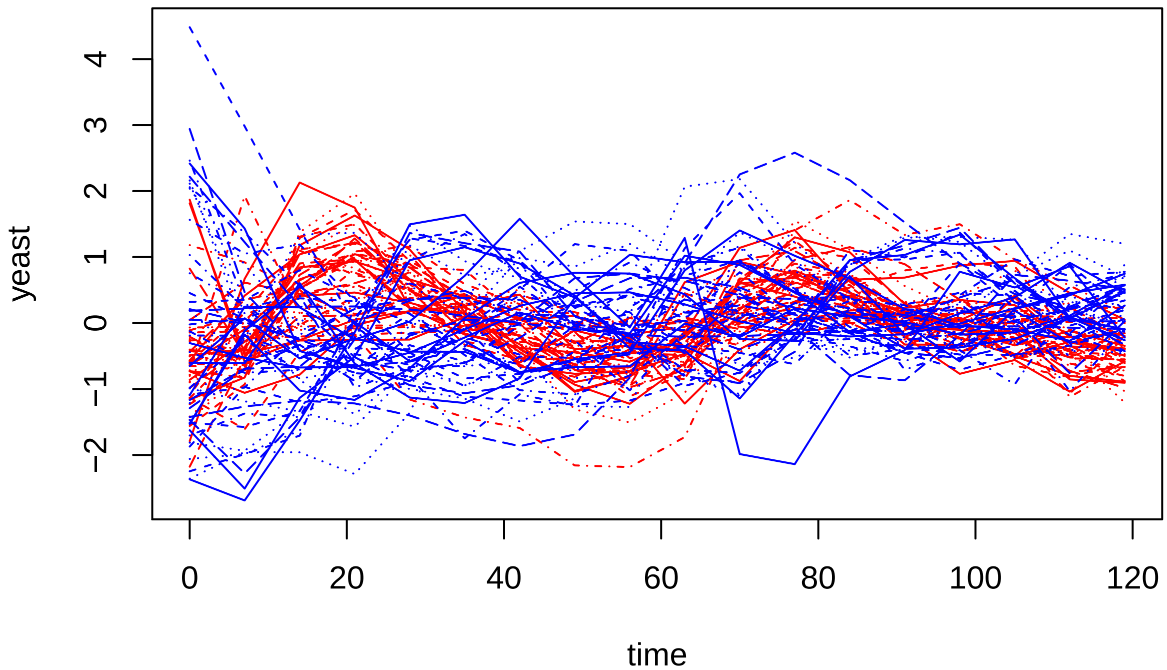
Problem 3

```
yeast <- read.table(file = "/Users/apple/Desktop/ISU 2019 fall/STAT547/data/yeast.txt")
time <- as.numeric( sub("alpha", "", names(yeast)) )
```

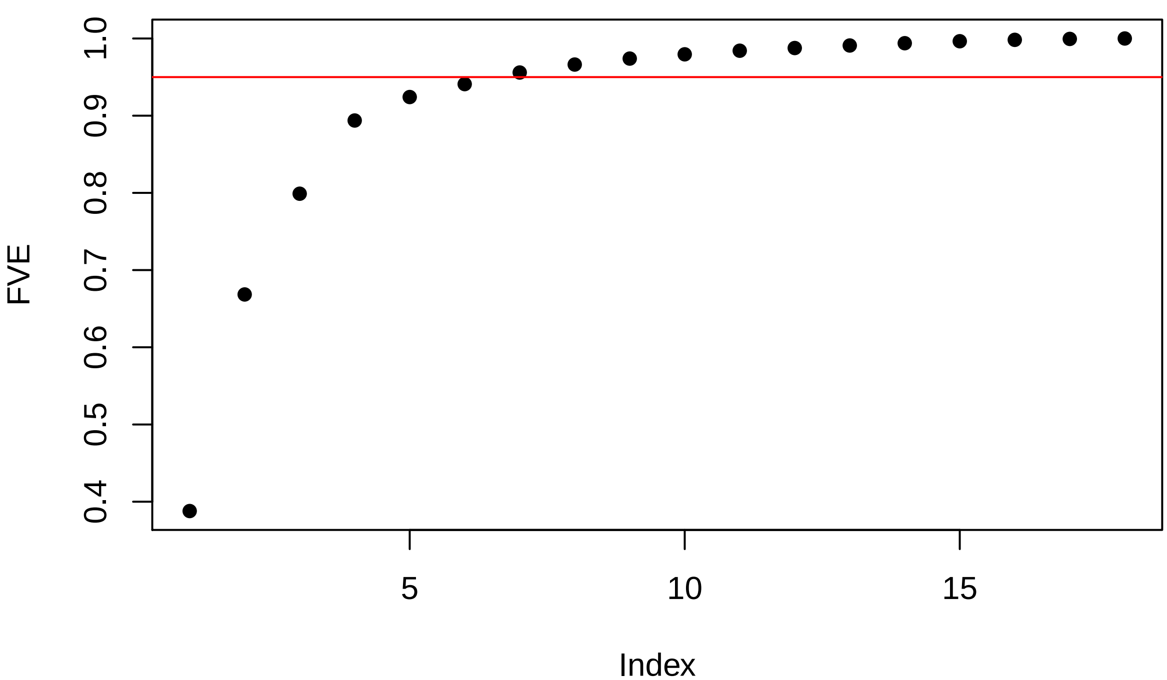
```r
## original data
X <- as.matrix(yeast)
matplot(time, t(X), type='l', col=c(rep(2,44), rep(4,45)),
        xlab = "time", ylab = "yeast")
```



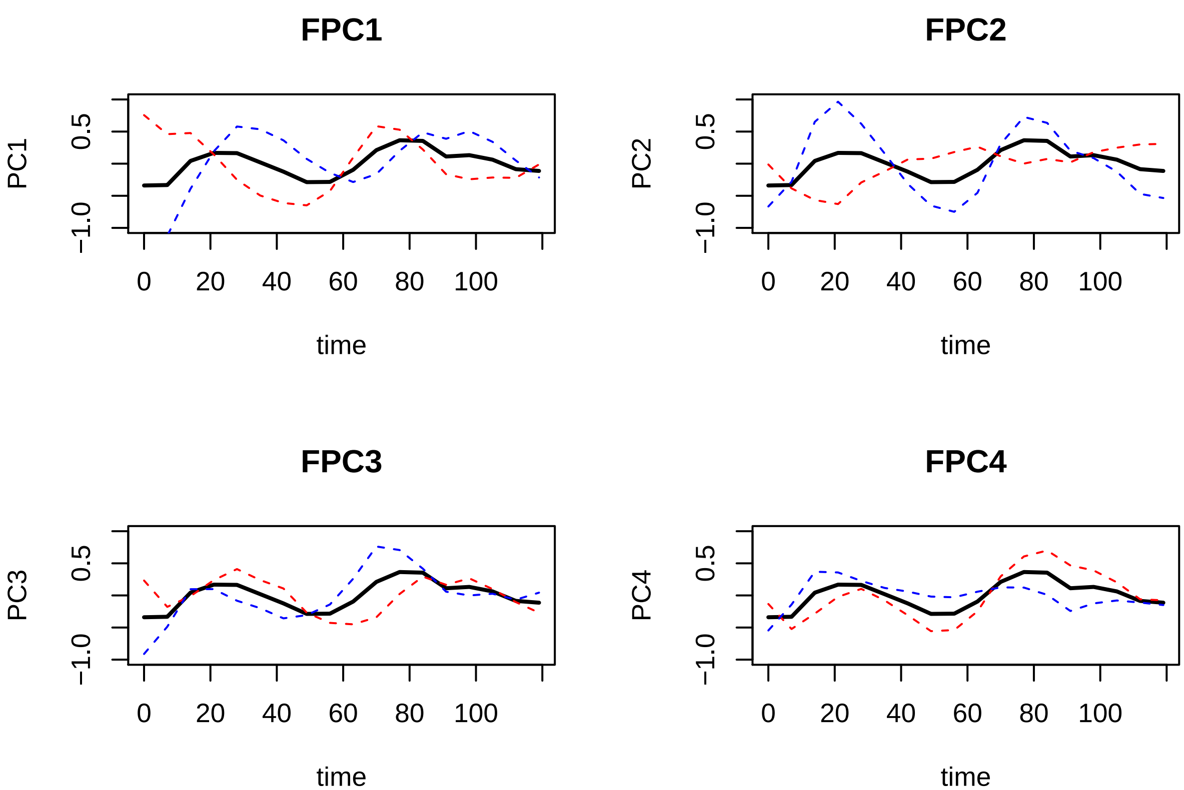

```r
n <- nrow(X)
m <- ncol(X)

## smoothed data
X_smooth <- matrix(0, ncol = m, nrow = n)

for (i in 1:n){
  ind <- which(is.na(X[i,]))
  if (length(ind) > 0){
    fit <- smooth.spline(time[-ind], X[i,][-ind])
  }else{
    fit <- smooth.spline(time, X[i,], all.knots = TRUE, spar = 0.1)
  }
    xhat <- predict(fit, as.data.frame(time))$y$time
    X_smooth[i,] <- xhat
}

matplot(time, t(X_smooth), type='l', col=c(rep(2,44), rep(4,45)),
        xlab = "time", ylab = "yeast")
```



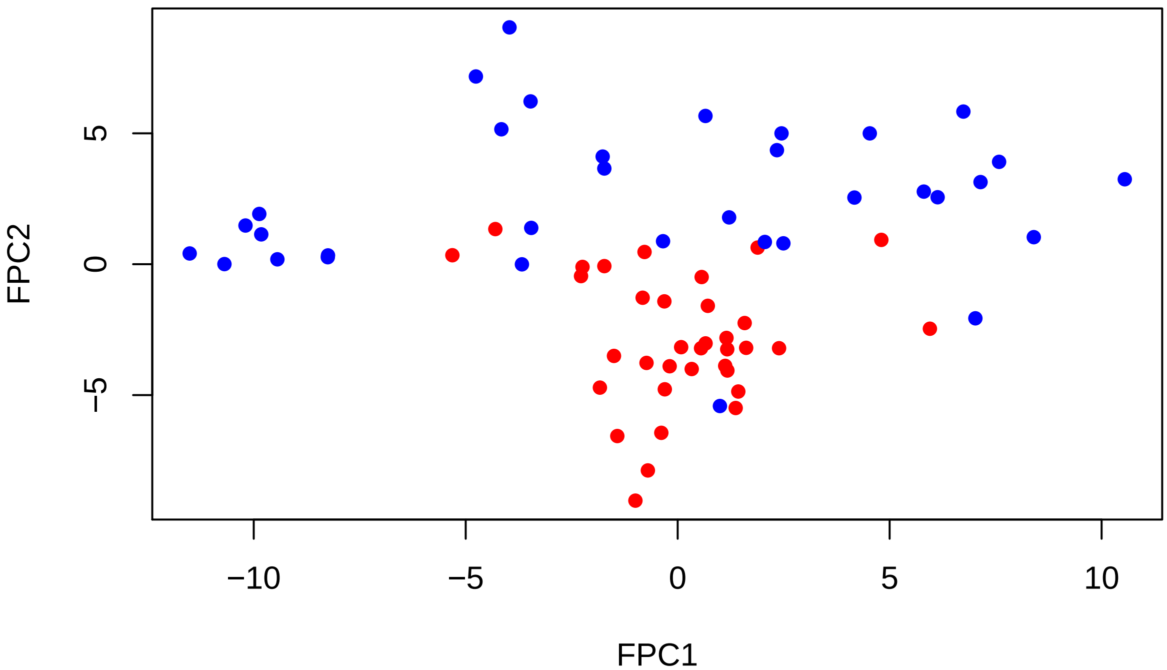

```r
## check the total variation
mu <- colMeans(X_smooth)
G <- cov(X_smooth)
eig <- eigen(G)
lam <- eig$values * diff(range(time)) / m
FVEeach <- lam / sum(lam)
FVE <- cumsum(FVEeach)

plot(FVE, pch = 16)
abline(h = 0.95, col = "red")
```



```
## Mode of variation plot
phi <- eig$vectors / sqrt(diff(range(time) / m))

par(mfrow = c(2,2))
lamphi1 <- sqrt(lam[1]) * phi[, 1]
plot(time, mu[1:m], type='l', lwd=2, ylim = c(-1,1),
     main = "FPC1", xlab = "time", ylab = "PC1")
lines(time, mu + lamphi1, type='l', lty=2, col='red')
lines(time, mu - lamphi1, type='l', lty=2, col='blue')

lamphi2 <- sqrt(lam[2]) * phi[, 2]
plot(time, mu, type='l', lwd=2, ylim = c(-1, 1),
     main = "FPC2", xlab = "time", ylab = "PC2")
lines(time, mu + lamphi2, type='l', lty=2, col='red')
lines(time, mu - lamphi2, type='l', lty=2, col='blue')

lamphi3 <- sqrt(lam[3]) * phi[, 3]
plot(time, mu, type='l', lwd=2, ylim = c(-1, 1),
     main = "FPC3", xlab = "time", ylab = "PC3")
lines(time, mu + lamphi3, type='l', lty=2, col='red')
lines(time, mu - lamphi3, type='l', lty=2, col='blue')

lamphi4 <- sqrt(lam[4]) * phi[, 4]
plot(time, mu, type='l', lwd=2, ylim = c(-1, 1),
     main = "FPC4", xlab = "time", ylab = "PC4")
lines(time, mu + lamphi4, type='l', lty=2, col='red')
lines(time, mu - lamphi4, type='l', lty=2, col='blue')
```



```r
par(mfrow = c(1,1))


## FPC
Xcenter <- X - matrix(mu, nrow=n, ncol=m, byrow=TRUE)
xi <- Xcenter %*% phi * diff(range(time)) / m
plot(xi[, 1], xi[, 2],  col=c(rep(2,44), rep(4,45)), pch = 16,
     xlab = "FPC1", ylab = "FPC2")
```



Problem 4

(a)

$$Z_k = \frac{1}{n}\sum_{i=1}^{n}\int_{\mathcal{T}}(X_i(t) - \hat{\mu}(t))\hat{\phi}_k(t)dt = \int_{\mathcal{T}}(\frac{1}{n}\sum_{i=1}^{n}X_i(t) - \hat{\mu}(t))\hat{\phi}_k(t)dt = 0$$

(b)

$$\frac{1}{n-1}\sum_{i=1}^{n}(\hat{\xi}_{ik} - Z_k)^2 = \frac{1}{n-1}\sum_{i=1}^{n}\hat{\xi}_{ik}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\int_{\mathcal{T}}\int_{\mathcal{T}}\hat{\phi}_k(s)(X_i(s) - \hat{\mu}(s))(X_i(t) - \hat{\mu}(t))\hat{\phi}_k(t)dsdt$$
$$= \int_{\mathcal{T}}\int_{\mathcal{T}}\hat{\phi}_k(s)\hat{G}(s,t)\hat{\phi}_k(t)dsdt = \int_{\mathcal{T}}\hat{\phi}_k(s)\hat{\phi}_k(s)\hat{\lambda}_k ds = \hat{\lambda}_k$$

(c)

$$\frac{1}{n-1}\sum_{i=1}^{n}(\hat{\xi}_{ik}-Z_k)(\hat{\xi}_{ik'}-Z_{k'}) = \frac{1}{n-1}\sum_{i=1}^{n}\hat{\xi}_{ik}\hat{\xi}_{ik'} = \int_{\mathcal{T}}\int_{\mathcal{T}}\hat{\phi}_k(s)\hat{G}(s,t)\hat{\phi}_k(t)dsdt$$
$$= \int_{\mathcal{T}}\hat{\phi}_{k'}(s)\hat{\phi}_k(s)\hat{\lambda}_{k'}ds = \hat{\lambda}_k\delta_{kk'}$$

Problem 5

Define: $\mathcal{J} = \sum_{i=1}^{n}(x_i-\bar{x})\otimes(x_i-\bar{x})$, then $\forall y\in\mathbb{H}$:

$$\mathcal{J}(y) = \sum_{i=1}^{n}(x_i-\bar{x}) < x_i-\bar{x}, y >,$$

which means $\mathcal{I}\Updownarrow(\mathcal{J}) = \mathrm{Span}\{\mathrm{x_i-\bar{x}}\}_{i=1}^{n}$. Also note that $\{x_i-\bar{x}\}_{i=1}^{n}$ is one degree of freedom less than n sine $\sum_{i=1}^{n}x_i-\bar{x}=0$, therefore:

$$\mathrm{Rank}(\mathcal{J}) = \dim(\mathcal{I}\Updownarrow(\mathcal{J})) = n-1.$$

Problem 6

(a)

To show absolute convergence, we only need to prove:

$$\sum_{j=1}^{\infty}||a_j\lambda_j e_j||_K < \infty,$$

This is true because:

$$\sum_{j=1}^{\infty}||a_j\lambda_j e_j||_K = \sum_{j=1}^{\infty} < a_j\lambda_j e_j, a_j\lambda_j e_j >_K = \sum_{j=1}^{\infty}a_j^2\lambda_j < \infty.$$

(b)

Because:

$$K(\cdot,t) = \sum_{j=1}^{\infty}\lambda_j e_j(\cdot)e_j(t),$$

Let $a_{t,j} = e_j(t)$, then:

$$\sum_{j=1}^{\infty}\lambda_j a_{t,j}^2 = \sum_{j=1}^{\infty}\lambda_j e_j(t)e_j(t) = K(t,t) < \infty,$$

The last inequality holds because $\{K(t,t), t\in[0,1]\}$ is compact.

(c)

$\forall f \in \mathcal{H}_K$, $\exists f = \sum_{j=1}^{\infty} a_j \lambda_j e_j$, therefore:

$$< K(\cdot, t), f >_K = < \sum_{j=1}^{\infty} \lambda_j e_j(t) e_j, \sum_{j=1}^{\infty} a_j \lambda_j e_j >_K = \sum_{j=1}^{\infty} \lambda_j a_j e_j(t) = f(t).$$