HW 4

Wenjuan Bian

2023-07-24

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

5.17 [Concurrent functional model] Dynamic biometric measurements moti□vated the early development of functional data analysis. A classical example, explained in some detail in Section 10.2.3. of Ramsay et al. (2009), is the gait data. Figure 5.9 shows suitably defined angles made at matching times by knees and hips of 39 boys. A question of interest is the extent to which the hip angle can explain the knee angle. A model proposed to study this question takes the form Yi(t) = β0(t) + β1(t)Xi(t) + εi(t). It is know as the concurrent model because the value Yi(t) of the response curve Yi depends only on the value of the regressor Xi at the same (concurrent) time t. This should be contrasted with the fully functional model (5.10). The coefficient functions β0 and β1 can be estimated in several ways, saturated basis expansion with a penalty term being the most popular one. (a) Use the following code to create functional objects containing hip and knee angles: require(fda) gaittime <- as.numeric(dimnames(gait)[[1]])*20 gaitrange <- c(0,20) gaitbasis <- create.fourier.basis(gaitrange, nbasis=21) harmaccelLfd <- vec2Lfd(c(0, (*2pi/20)ˆ2, 0), rangeval= gaitrange) gaitfd <- smooth.basisPar(gaittime, gait, gaitbasis, Lfdobj=harmaccelLfd, lambda=1e-2)$fd hipfd <- gaitfd[,1]; kneefd <- gaitfd[,2] Using the objects defined above, reproduce Figure 5.9

```
library(fda)
```

```
## Warning: package 'fda' was built under R version 4.2.3
```

```
## Loading required package: splines
```

```
## Loading required package: fds
```

```
## Warning: package 'fds' was built under R version 4.2.3
```

```
## Loading required package: rainbow
```

```
## Warning: package 'rainbow' was built under R version 4.2.3
```

```
## Loading required package: MASS
## Loading required package: pcaPP
```

```
## Warning: package 'pcaPP' was built under R version 4.2.3
```

```
## Loading required package: RCurl
```

```
## Warning: package 'RCurl' was built under R version 4.2.3
```

```
## Loading required package: deSolve
```

```
## Warning: package 'deSolve' was built under R version 4.2.3
```

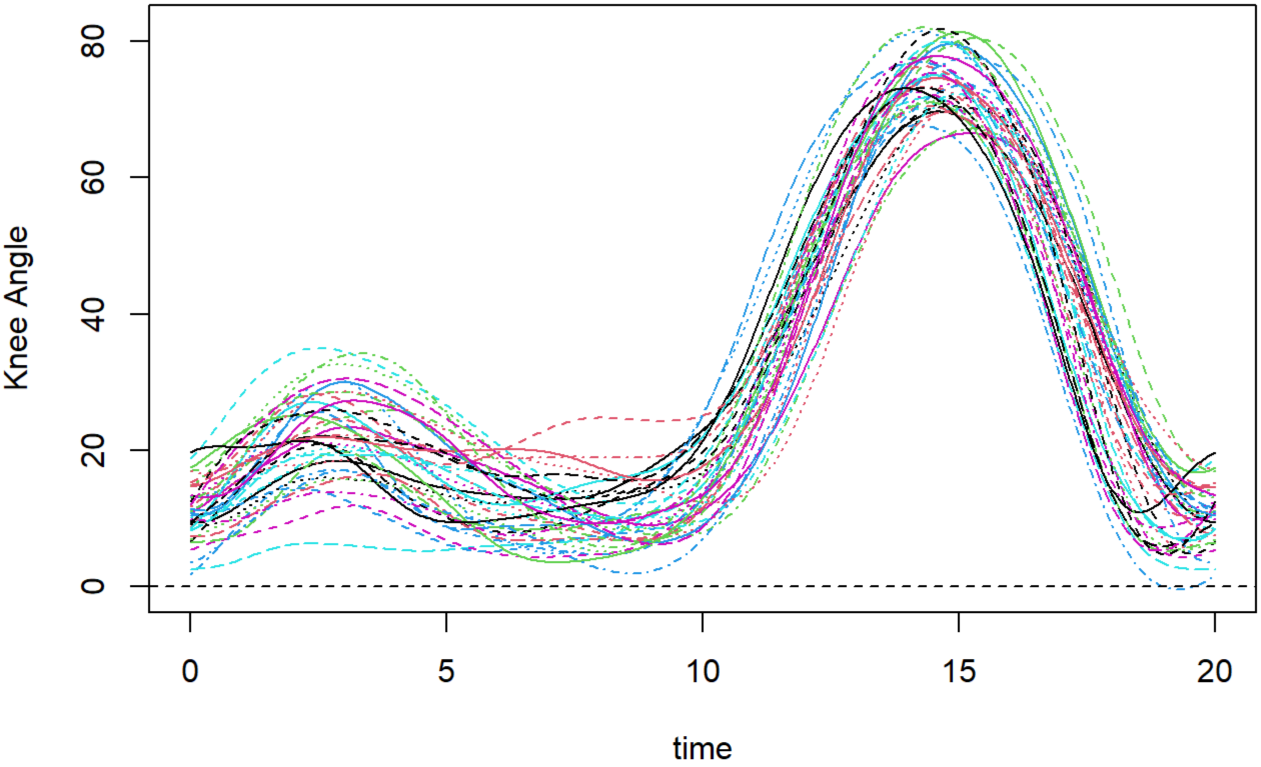
```
##
## Attaching package: 'fda'
```

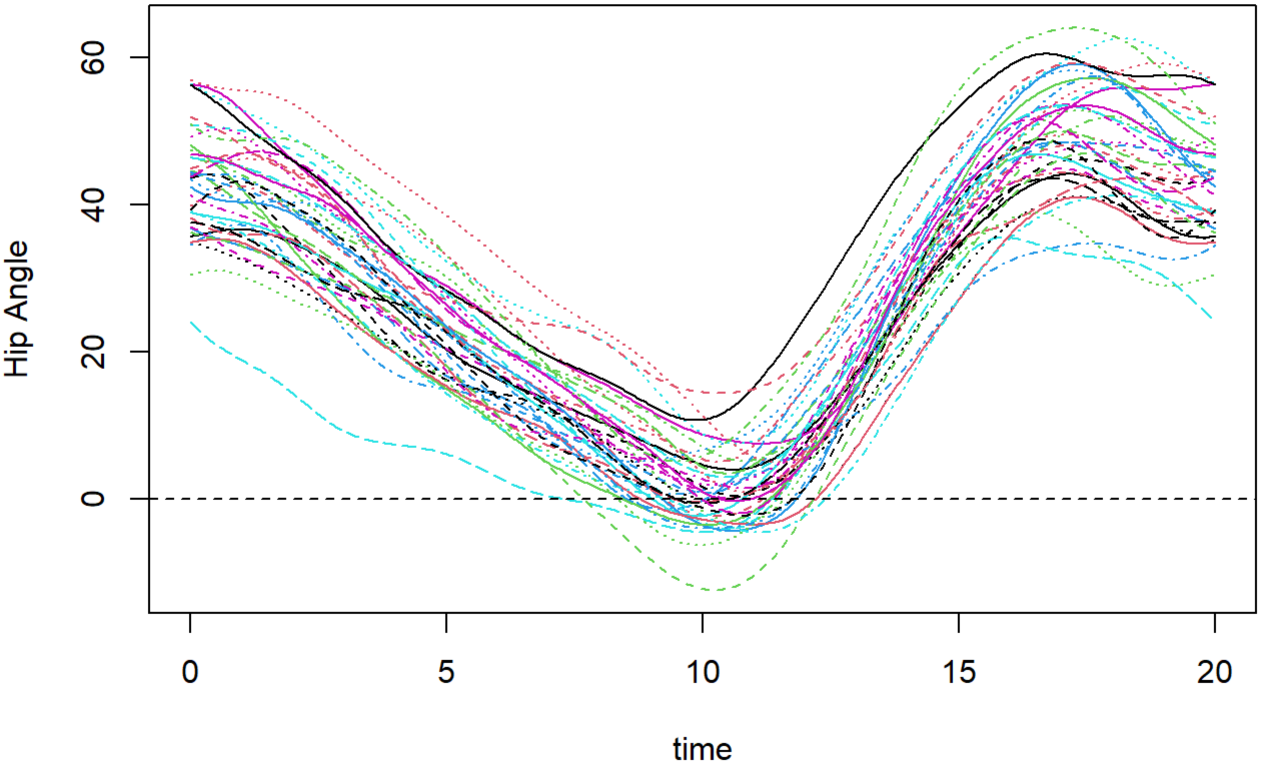
```
## The following object is masked from 'package:graphics':
##
##     matplot
```

```
library(refund)

gaittime <- as.numeric(dimnames(gait)[[1]])*20
gaitrange <- c(0,20)
gaitbasis <- create.fourier.basis(gaitrange, nbasis=21)
harmaccelLfd <- vec2Lfd(c(0, (2*pi/20)^2, 0), rangeval=gaitrange)

gaitfd <- smooth.basisPar(gaittime, gait, gaitbasis, Lfdobj=harmaccelLfd, lambda=1e-2)$fd
hipfd <- gaitfd[,1]; kneefd <- gaitfd[,2]

par(mfrow = c(2, 1))
plot(kneefd)
```

```
## [1] "done"
```

```
plot(hipfd)
```


Knee Angle
0
20
40
60
80
0
5
10
15
20
time


Hip Angle
0
20
40
60
0
5
10
15
20
time

```
## [1] "done"
```

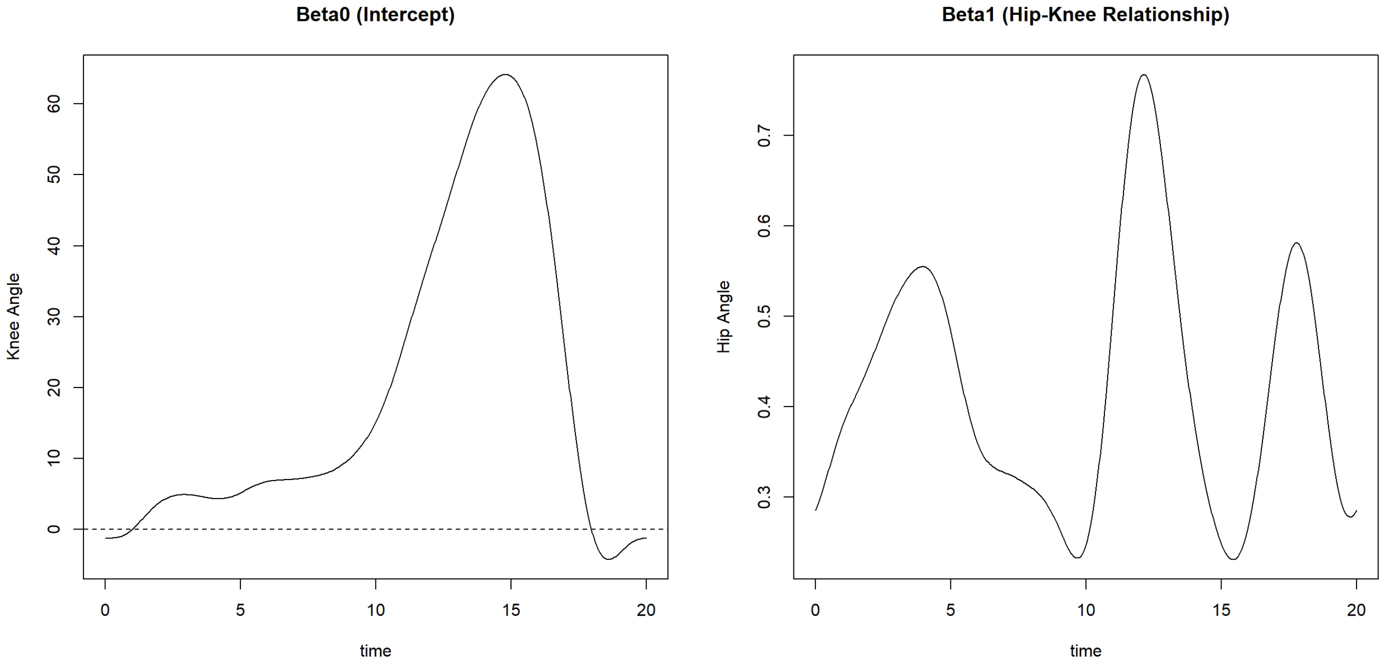
```
par(mfrow = c(1, 1))
```

b. Use the call knee.hip.f <- fRegress(kneefd ~ hipfd) to estimate the concurrent model to predict the knee angle from the hip angle. Plot the functions β0 and β1 and comment on the information contained in their shape.

```
knee.hip.f <- fRegress(kneefd ~ hipfd)

beta0 <- knee.hip.f$betaestlist[[1]]

beta1 <- knee.hip.f$betaestlist[[2]]

par(mfrow = c(1, 2))
plot(beta0$fd, main = "Beta0 (Intercept)")
```

```
## [1] "done"
```

```
plot(beta1$fd, main = "Beta1 (Hip-Knee Relationship)")
```



```
## [1] "done"
```

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

The shape of β0 shows a rise and subsequent peak at around t=15, followed by a sharp decline. This indicates that the baseline knee angle increases over time, peaks at around t=15, and then decreases abruptly. This suggests some sort of change that is occurring over time, which is independent of the

predictor variable. The oscillatory shape of β1 suggests that the effect of the predictor variable on the knee angle is not constant but varies in a cyclical pattern over time. The three peaks at around t=4, t=12.5, and t=18, with the highest at t=12.5, suggests that the influence of the predictor variable on the knee angle is strongest at these times, but wanes in between. This could suggest an interaction between the predictor variable and some sort of cyclic process affecting the knee angle.

c. We want to compare the predicted curve for boy 1 in the sample with his observed knee angle curve. Use the the following calls to plot these two curves in separate panels: plot(knee.hip.f$yhatfd$ y[,1],type=‘l’,lwd=2,lty=2, ylab=“Knee Angle”) plot(kneefd[‘boy1’], lwd=2) Notice and explain the different ranges of the x–arguments in the two plots.

```
print(range(kneefd$basis$rangeval))
```

```
## [1]  0 20
```

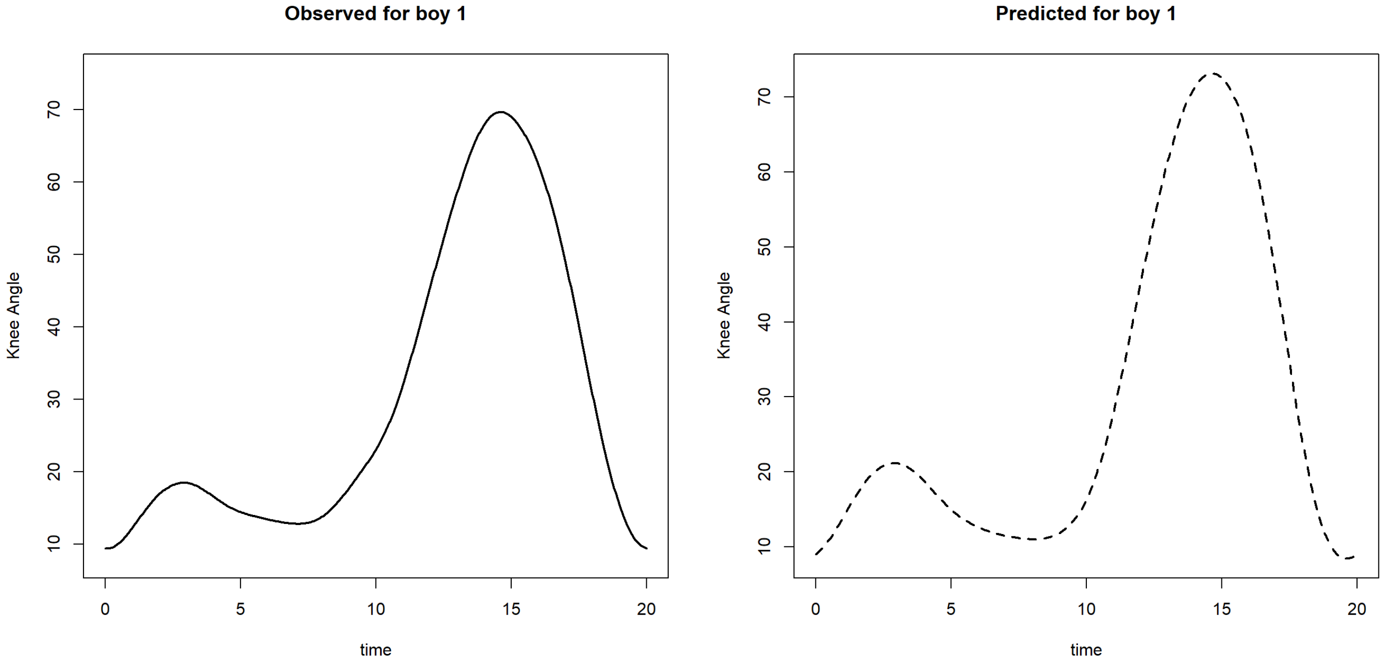
```
print(range(knee.hip.f$yhatfd$fdnames$time))
```

```
## [1]   1 501
```

```
par(mfrow = c(1, 2))
plot(kneefd['boy1'], ylim=c(8, 75), lwd=2, main="Observed for boy 1")
```

```
## [1] "done"
```

```
plot(knee.hip.f$yhatfd[1],lwd=2, lty=2, ylab="Knee Angle", main = "Predicted for boy 1")
```



Observed for boy 1

Predicted for boy 1

```
## [1] "done"
```

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

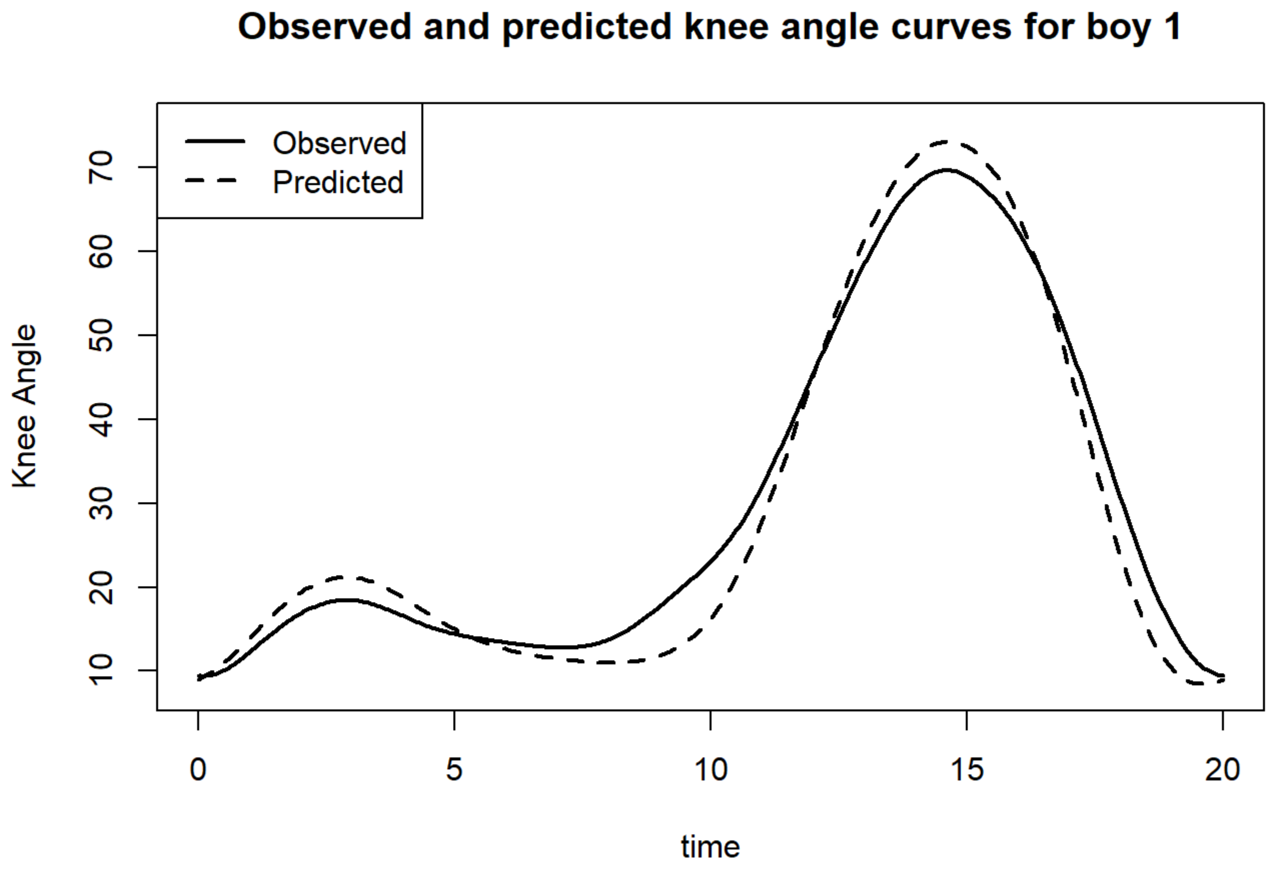
The ranges of the x-arguments (time points) for these two functional data are not the same. The kneefd has x-arguments ranging from 0 to 20, whereas the ‘knee.hip.f$yhatfd’ has x-arguments ranging from 1 to 501. ‘gaitrange <- c(0,20)’ is created for the Fourier basis. This range corresponds to the actual time duration over which ther gait data was collected. However, ‘knee.hip.f$yhatfd$fdnames$time’ returns the range of the time points at which the predicted knee angles are evaluated. The range of [1,501] means the predicted curves is evaluated at 501 equally spaced points over the range of the basis functions used in the regression analysis.

Plot the two curves in one panel, as shown in Figure 5.10. Extract the x–axis values of the predicted curve with argvals. Use legend to identify the curves.

```
plot(kneefd['boy1'], ylim = c(8, 75),lwd = 2, main = "Observed and predicted knee angle curves for boy 1")
```

```
## [1] "done"
```

```
lines(knee.hip.f$yhatfd[1], ylim = c(8, 75),lwd = 2,lty = 2)
legend("topleft", legend = c("Observed", "Predicted"), lty = 1:2, lwd = 2)
```


Observed and predicted knee angle curves for boy 1
Observed
Predicted
Knee Angle
time