Department of Statistics
Lund University

Functional Data Analysis
Spring

Project 2: Splines, functional principle component.

Smoothing by B-splines, comparison between theoretical and empirical principle components

Perform all requested task. Your work will be monitored and the credit for it will be given based on your in-lab activities. Some useful R-code that can help in completing Project 2 can be found **here**.

Part One – Smooth spline fitting a generalized additive model with one predictor

We consider the precipitation data set for which we discuss smoothing techniques. The two packages will be considered for this purpose: `splines` that is specifically designated for the smoothing purposes and `fda` that contains smoothing methods as a part of its routines. We start with smoothing a single function. We will use the acceleration of the `growth` data that come with the fda package.

1. Load the package *"splines"* to R system.
2. Check the growth data set provide some plots of its components.

```
help(growth)

BH=growth$hgtm
GH=growth$hgtf
A=growth$age

plot(A,BH[,1],type="l",col=1,ylim=c(min(BH),max(BH))) #First Boy
for(i in 2:dim(BH)[2]){
  lines(A,BH[,i],type="l",col=i)
}

# Velocity for boys

VBH=t(t(apply(BH,2,diff))%*%diag(diff(A)))
AA=A[2:length(A)]

plot(AA,VBH[,1],type="l",col=1,ylim=c(min(VBH),max(VBH))) #First Boy
for(i in 2:dim(VBH)[2]){
  lines(A,BH[,i],type="l",col=i)
```

```
}

# Acceleration for boys

ABH=t(t(apply(VBH,2,diff))%*%diag(diff(AA)))
AAA=AA[2:length(AA)]

plot(AAA,ABH[,1],type="l",col=1,ylim=c(min(ABH),max(ABH))) #First Boy
for(i in 2:dim(ABH)[2]){
  lines(AAA,ABH[,i],type="l",col=i)
}
```

3. Evaluate the acceleration of the growth data.

```
# Velocity for boys

VBH=t(t(apply(BH,2,diff))%*%diag(diff(A)))
AA=A[2:length(A)]

plot(AA,VBH[,1],type="l",col=1,ylim=c(min(VBH),max(VBH))) #First Boy
for(i in 2:dim(VBH)[2]){
  lines(A,BH[,i],type="l",col=i)
}

# Acceleration for boys

ABH=t(t(apply(VBH,2,diff))%*%diag(diff(AA)))
AAA=AA[2:length(AA)]

plot(AAA,ABH[,1],type="l",col=1,ylim=c(min(ABH),max(ABH))) #First Boy
for(i in 2:dim(ABH)[2]){
  lines(AAA,ABH[,i],type="l",col=i)
}
```

4. Perform the following sequence of commands for a single element of the acceleration data.

```
#Fitting the curve for the first boy
fit=lm(ABH[,1]~bs(AAA,knots=c(5,8,10,15)))

Al=range(AAA)
Agrid=seq(from=Al[1],to=Al[2],length.out =100)

pred=predict(fit,newdata=list(AAA=Agrid),se=T)
plot(AAA,ABH[,1],col="gray")
lines(Agrid,pred$fit,lwd=2)
```

```
lines(Agrid,pred$fit+2*pred$se,lty="dashed")
lines(Agrid,pred$fit-2*pred$se,lty="dashed")
```

Explain what tasks have been performed.

5. In the presented approach we have specified explicitly knots. Alternatively, one can consider the degree of freedom as well as using the higher order than cubic splines and we can even use different splines than the B-splines (the normal splines in the code below). The following code is performing these tasks. Explain particular steps of the code.

```
dim(bs(AAA,knots=c(5,8,10,15)))
dim(bs(AAA,df=7))

attr(bs(AAA,df=7),"knots") #These are knots returned by B-splines method

BS=bs(AAA,knots=c(5,8,10,15),intercept=TRUE)
dim(BS)

plot(AAA,BS[,1], col=1, type='l',ylab="",lwd=2)
for(i in 2:8)
{
  lines(AAA,BS[,i], col=i,lwd=2)
}

fit2=lm(ABH[,1]~ns(AAA,df=7))
pred2=predict(fit2,newdata=list(AAA=Agrid))

lines(Agrid, pred2,col="red",lwd=2)
```

   Compare fits using degrees of freedom versus specified knots. What knots have been chosen when the degree of freedom method was applied?

6. Next we turn to the smooth splines.

```
title("Smoothing Spline")
fit=smooth.spline(AAA,ABH[,1],df=50)
fit2=smooth.spline(AAA,ABH[,1], cv=T)

fit2$df

plot(AAA,ABH[,1],col="gray")
lines(fit,col="red",lwd=2)
lines(fit2,col="blue",lwd=2)
legend("topright",legend=c("50 DF","8 DF"),col=c("red","blue"),lty=1,lwd=2,cex=.8)
```

   Notice that in the first call to `smooth.spline()`, we specified `df=50`. The function then determines which value of λ leads to 50 degrees of freedom. In the second call to `smooth.spline()`, we select the smoothness level by cross-validation; this results in a value of λ that yields 2 degrees of freedom. Judging from the fit which of these two methods you prefer and why? Either something else?

Part Two – Covariance estimation

In this part you practice the estimation of the covariance using the `fda` package. We will work with precipitation data from the Canadian weather data set.

1. Extract log-precipitation data from the data set. Create some plots that give some ideas how the data look like.

2. Use that data to estimate the mean and the covariance kernel and present the graphical representation of the estimates.

3. Perform smoothing on the data as described in the textbook.

4. Make an estimate of the mean and covariance on smoothed data.

Part Three – Principle component

In this part, you are asked to perform principle component analysis on the log-precipitation data.

1. Smooth precipitation data as discussed before. Present their graphs.

2. Perform pca analysis. How many principle components seem to be meaningful.

3. Plot the eigenfunctions and corresponding eigenvalues.

4. Compute approximation of the covariance function based on the PC and compare it with the estimated covariance.