Appendix 2

GWW 12031299

2021/3/6

Smooth the fruit flies data

```
rm(list = ls())
library(fda)
library(ggplot2)
library(reshape2)
load('E://R Files/Functional Data/Data Sets/fly.Rdata')
flyname = colnames(medfly$eggcount)
day = 1:26
y = medfly$eggcount[,flyname[1]]
```

Find out the best number basis by SSE(OCV)

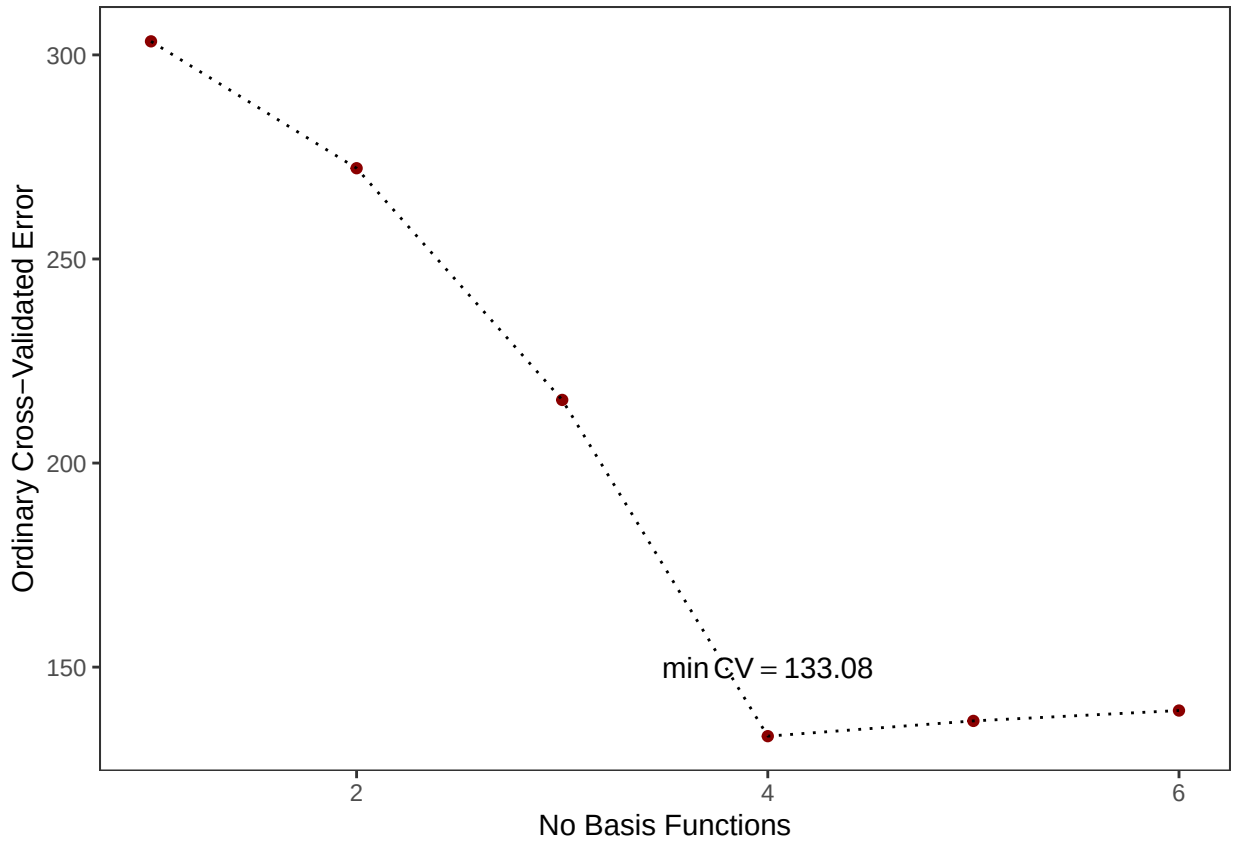
```
num1 = 6
P_bvals = matrix(day,26,num1)
for(i in 1:num1){ P_bvals[,i] = ((P_bvals[,i]-mean(y))/var(y))^(i-1)}
P_bvals[,1] = 0.01

smsse = rep(0,num1)
for(i in 1:num1){
  S = P_bvals[,1:i]%*%solve( t(P_bvals[,1:i])%*%P_bvals[,1:i])%*%t(P_bvals[,1:i])
  h = (1-diag(S))^2
  errs = y - S%*%y
  smsse[i] = sum(errs^2/h )/26
}
print(smsse)
```

```
## [1] 303.3216 272.2339 215.4667 133.0854 136.8218 139.3622
```

```
which.min(smsse)
```

```
## [1] 4
```

```
ggplot(data=NULL,aes(1:num1,smsse[1:num1]))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())+xlab("No Basis Functions") +ylab("Ordinary Cross-Validat
  annotate("text",x =4, y = 150,parse = T,label="min(CV) == 133.08")
```



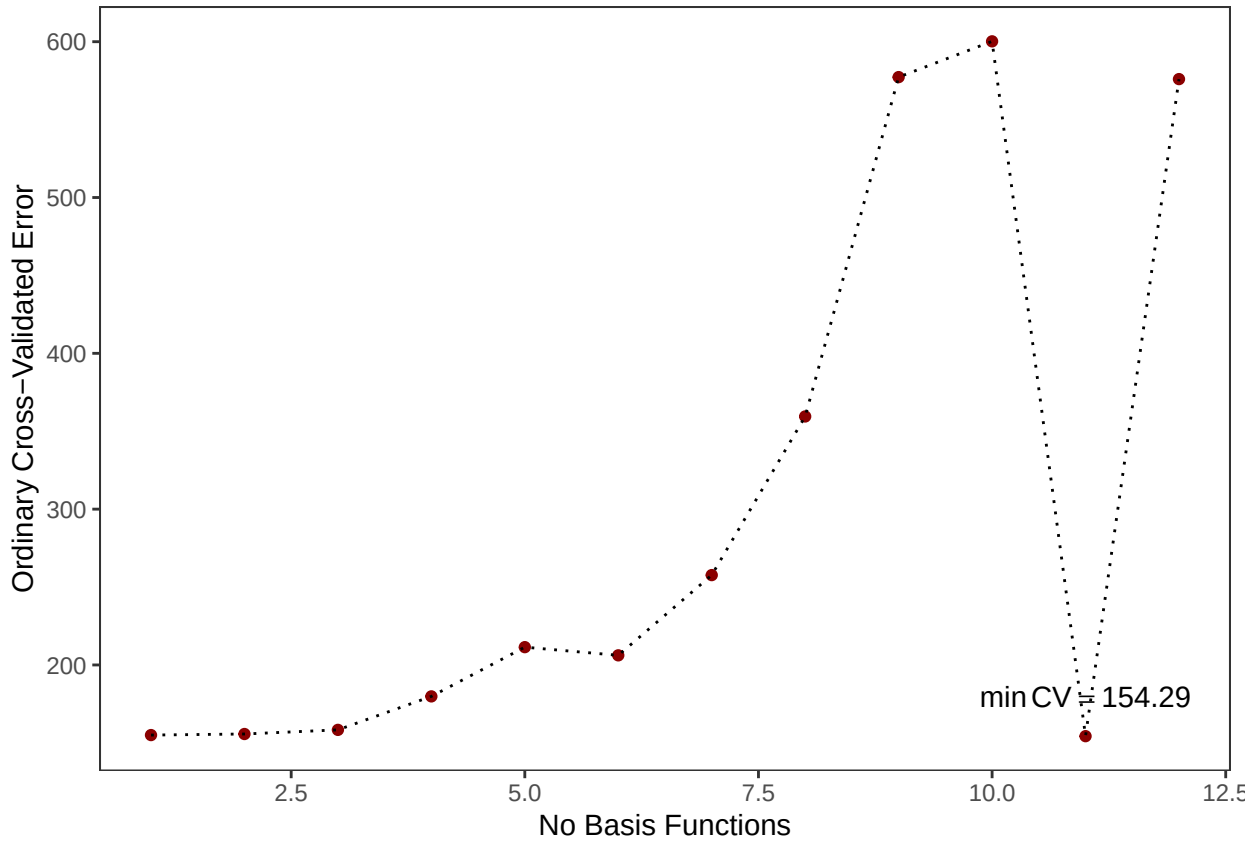

```
daybasis = create.fourier.basis(c(0, 26), 26)
bvals = eval.basis(day,daybasis)
num2 = 12
smsse = rep(0,num2)
for(i in 1:num2){
  F_bvals = bvals[,1:(2*i+1)]
  S = F_bvals%*%solve( t(F_bvals)%*%F_bvals)%*%t(F_bvals)
  h = (1-diag(S))^2
  errs = y - S%*%y
  smsse[i] = sum(errs^2/h )/26
}
print(smsse)
```

```
##  [1] 154.9942 155.6753 158.3300 179.8173 211.4156 206.1704 257.6648 359.4800
##  [9] 577.2818 600.2313 154.2935 576.0000
```

```
which.min(smsse)
```

```
## [1] 11
```

```
ggplot(data=NULL,aes(1:num2,smsse[1:num2]))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())+xlab("No Basis Functions") +ylab("Ordinary Cross-Validat
  annotate("text",x =11, y = 180,parse = T,label="min(CV) == 154.29")
```



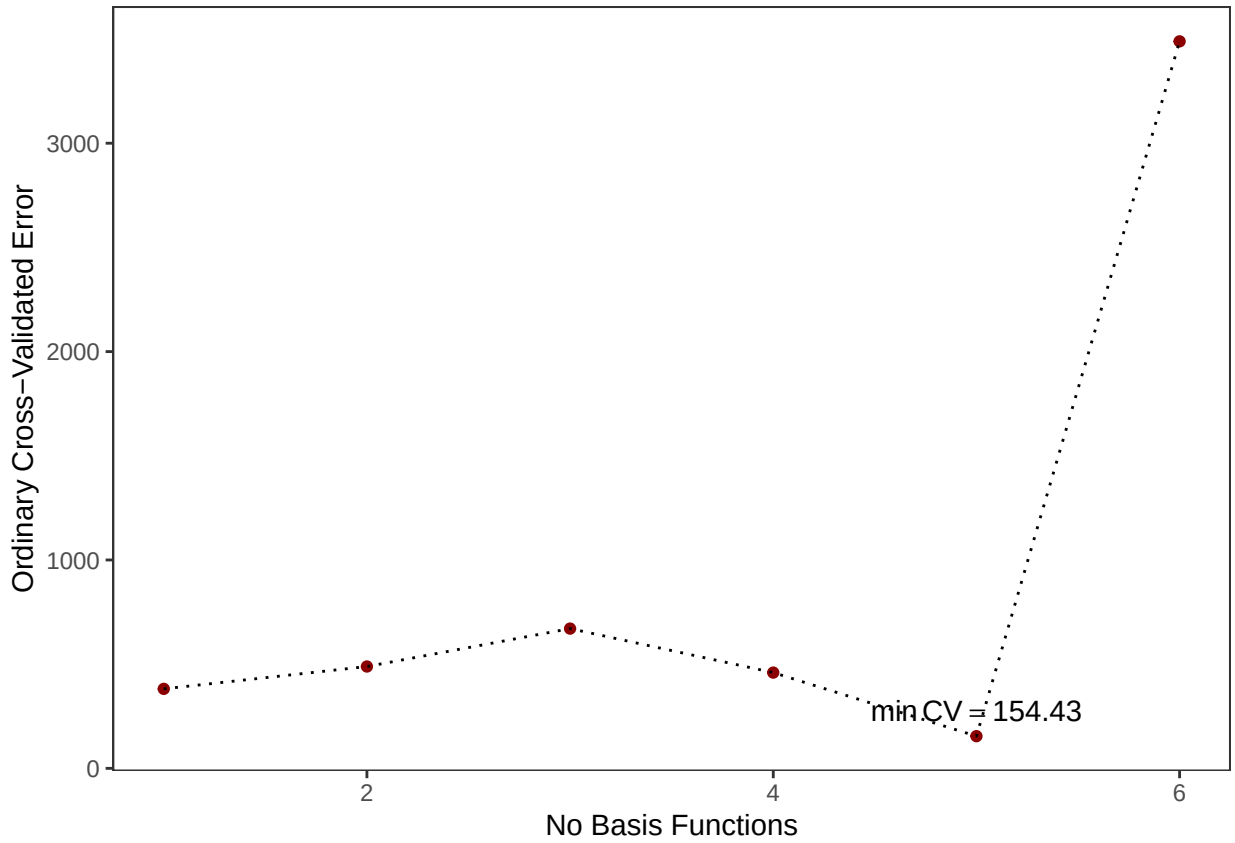

```
bbasis = list()
for (i in 1:6){
  bbasis[[i]] = create.bspline.basis(rangeval=c(0,26), nbasis = 7+i, norder=4) # nbasis=nbreaks+norder-
}
smsse = rep(0,6)
for(i in 1:6){
  bvals = eval.basis(day,bbasis[[i]])
  S = bvals%*%solve( t(bvals)%*%bvals)%*%t(bvals)
  h = (1-diag(S))^2
  errs = y - S%*%y
  smsse[i] = sum(errs^2/h )/26
}
print(smsse)
```

```
## [1]  381.4631  488.6264  670.6217  459.5390  154.4385 3488.9648
```

```
which.min(smsse)
```

```
## [1] 5
```

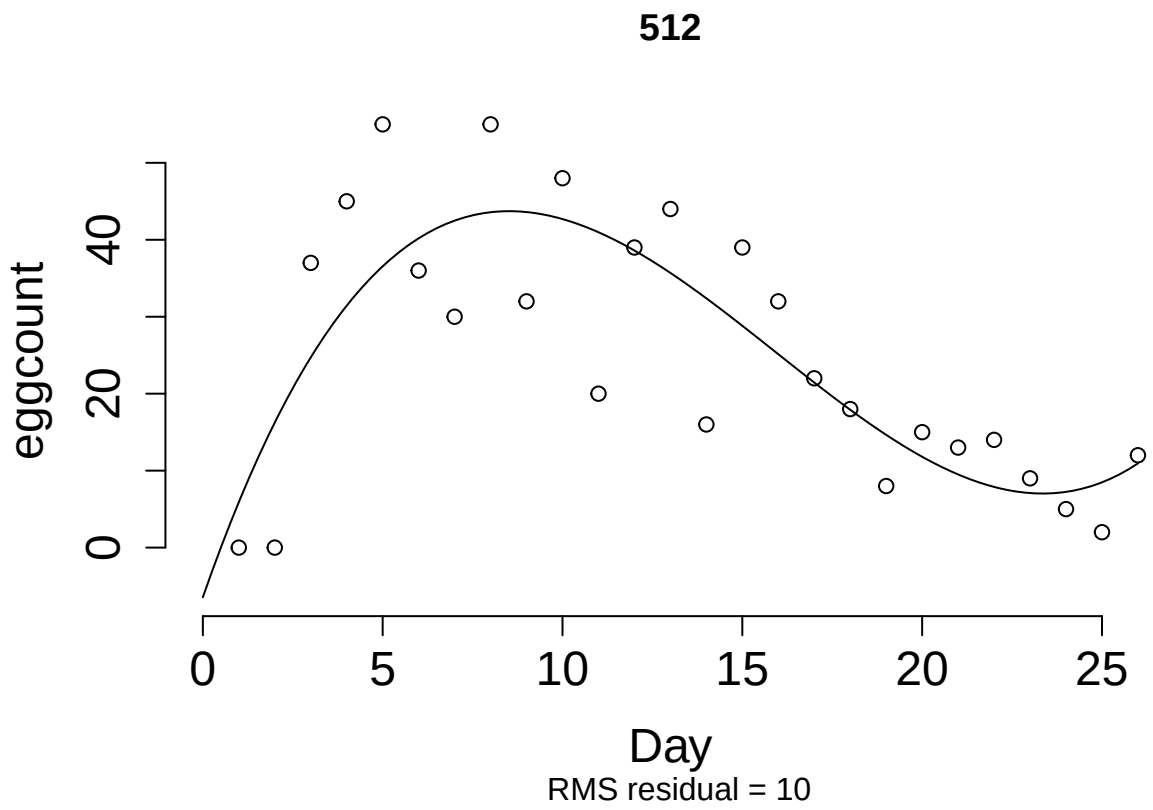
```
ggplot(data=NULL,aes(1:6,smsse[1:6]))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())+xlab("No Basis Functions") +ylab("Ordinary Cross-Validate
  annotate("text",x =5, y = 280,parse = T,label="min(CV) == 154.43")
```



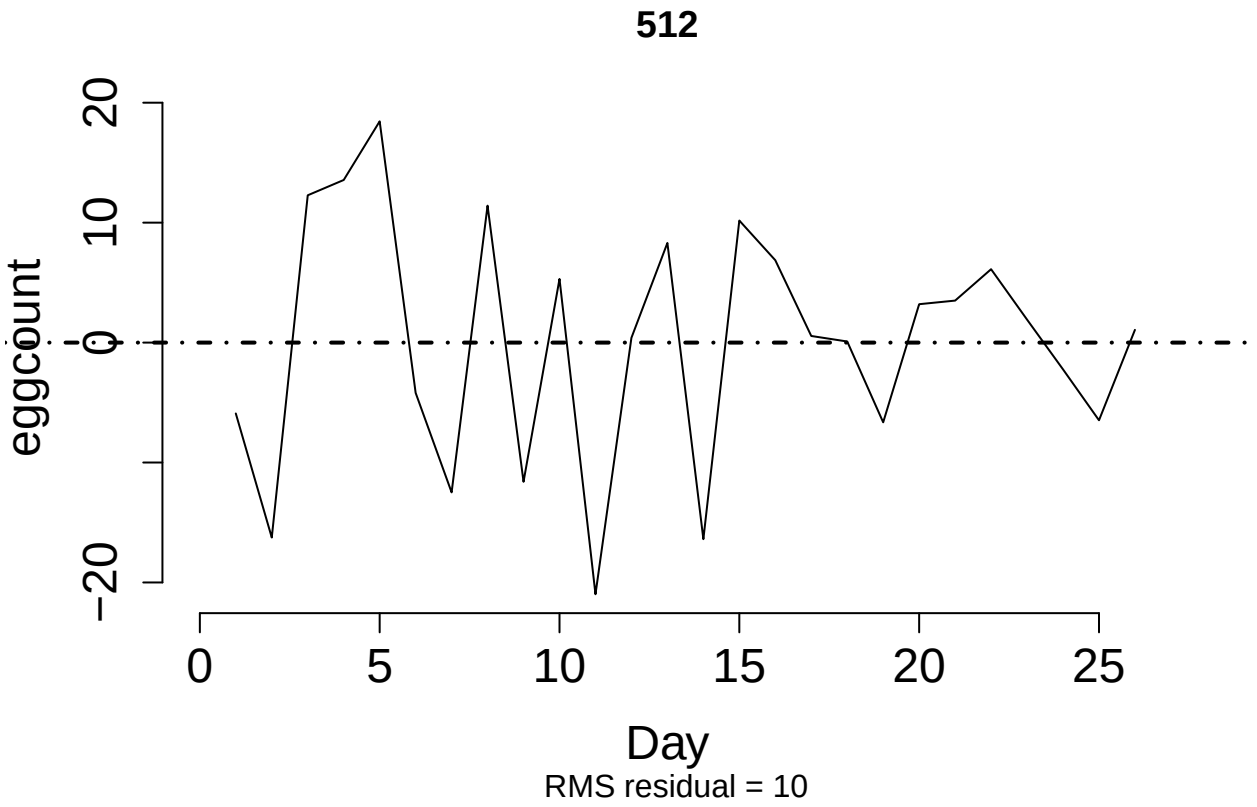

```
# bbasis = create.bspline.basis(rangeval=c(0,26), nbasis = 26, norder=4)
# D2fdPar = fdPar(bbasis,Lfdobj=int2Lfd(2),lambda=10)
# P_dayEggfd = smooth.basis(day,y,D2fdPar)
# plotfit.fd(y,day,P_dayEggfd$fd,title="lambda=10",
#            xlab="Day", ylab= "eggcount",cex.lab=1.5, cex.axis=1.5)
# L_RMSE = sqrt(mean((eval.fd(day, P_dayEggfd$fd) - y)^2))
# print(L_RMSE)
```

Least-square estimate

```
par(mfrow=c(1,1), xpd=NA, bty="n")

pbasis = create.power.basis(c(0,26),4,seq(0,3,1))
P_dayEggfd = smooth.basis(day, y, pbasis)
plotfit.fd(y, day, P_dayEggfd$fd, residual = FALSE, titles = "512",
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```



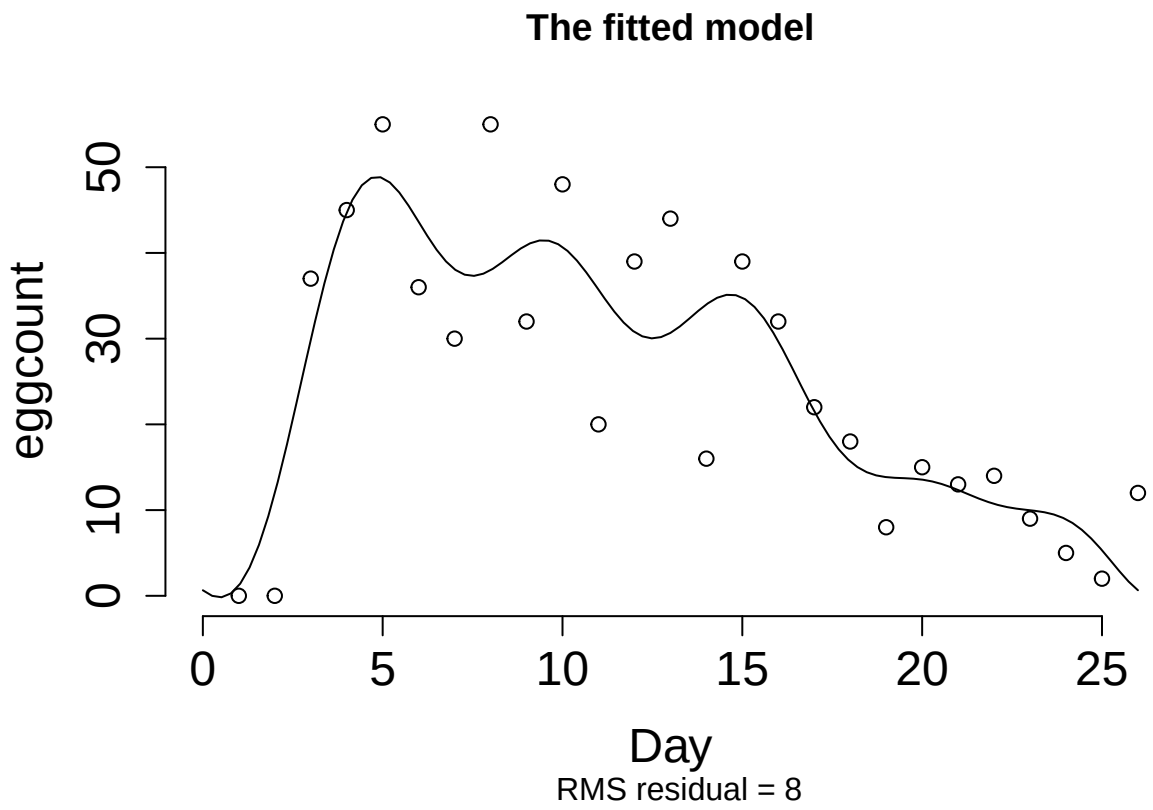

```
plotfit.fd(y, day, P_dayEggfd$fd, residual = TRUE,titles = "512",
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```



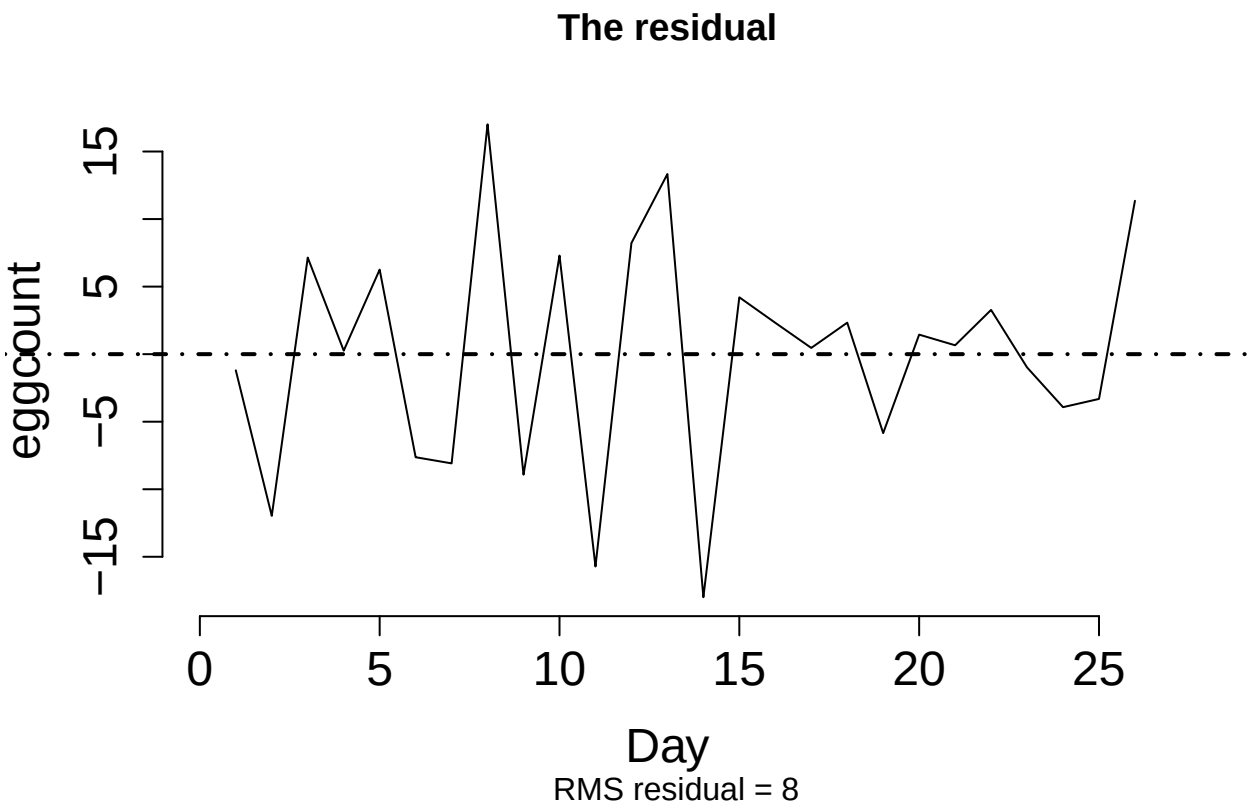

```
P_RMSE = sqrt(mean((eval.fd(day, P_dayEggfd$fd) - y)^2))
print(P_RMSE)
```

```
## [1] 9.855915
```

```
par(mfrow=c(1,1), xpd=NA, bty="n")

fbasis = create.fourier.basis(c(0, 26), 11)
F_dayEggfd = smooth.basis(day, y, fbasis)
plotfit.fd(y, day, F_dayEggfd$fd, residual = FALSE, titles = 'The fitted model',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```



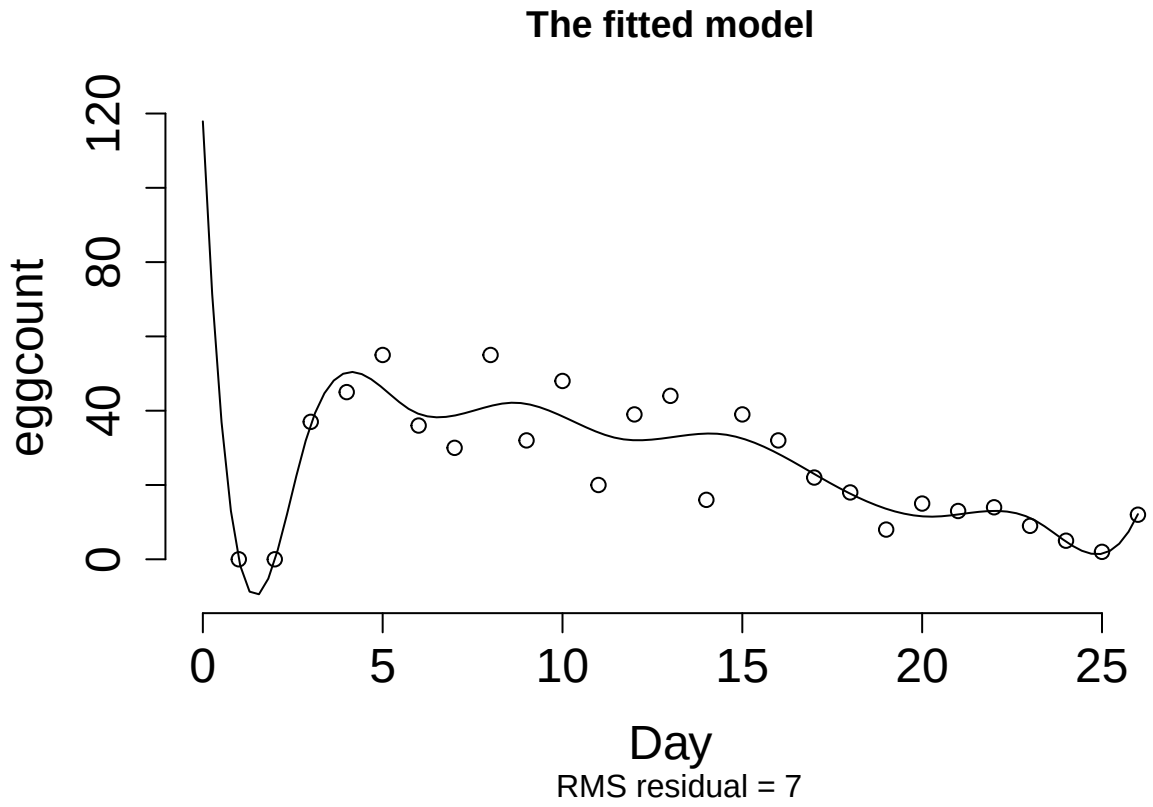

```
plotfit.fd(y, day, F_dayEggfd$fd, residual = TRUE, titles = 'The residual',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```



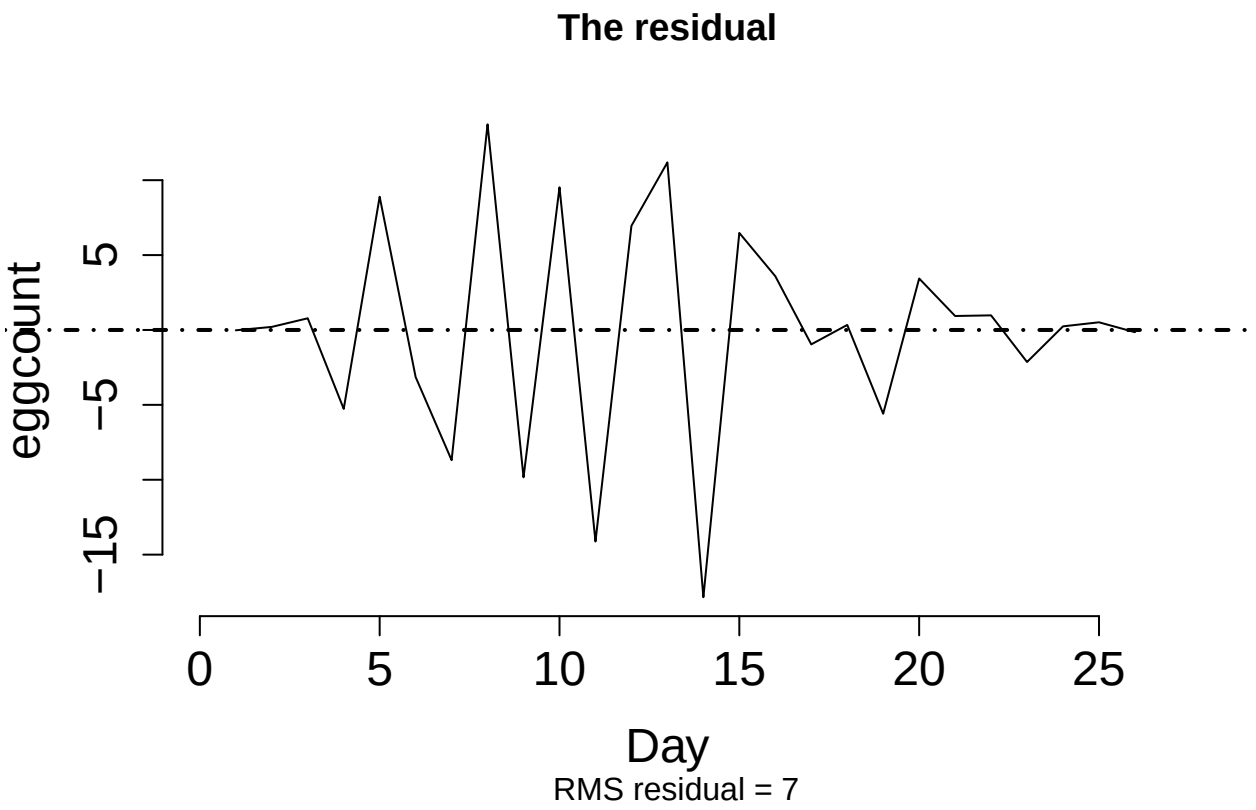

```
L_RMSE = sqrt(mean((eval.fd(day, F_dayEggfd$fd) - y)^2))
print(L_RMSE)
```

```
## [1] 8.388541
```

```
par(mfrow=c(1,1), xpd=NA, bty="n")

B_dayEggfd = smooth.basis(day, y, bbasis[[5]]) # 8 knots, 12 basis functions
plotfit.fd(y, day, B_dayEggfd$fd, residual = FALSE, titles = 'The fitted model',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5) # we can calculate the fitted value
```




```
plotfit.fd(y, day, B_dayEggfd$fd, residual = TRUE, titles = 'The residual',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```



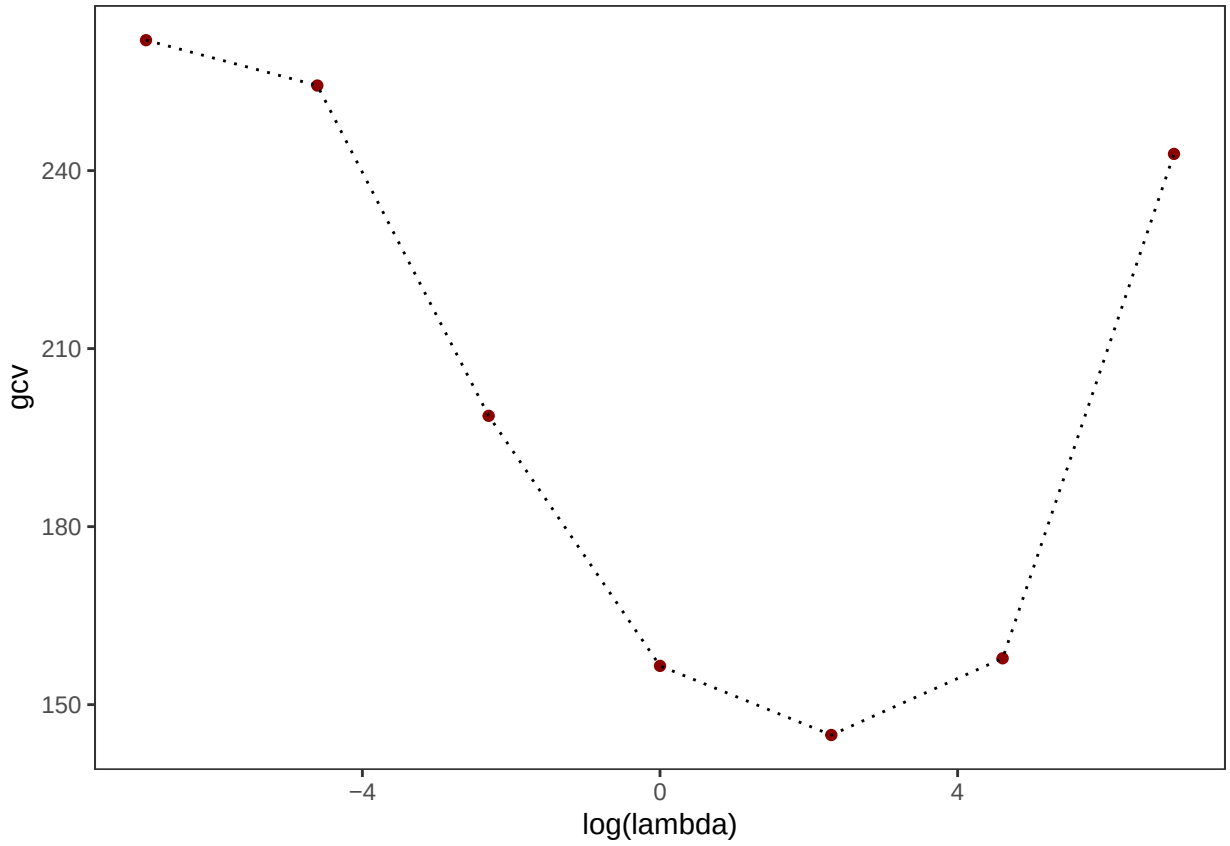

```
L_RMSE = sqrt(mean((eval.fd(day, B_dayEggfd$fd) - y)^2))
print(L_RMSE)
```

```
## [1] 7.248359
```

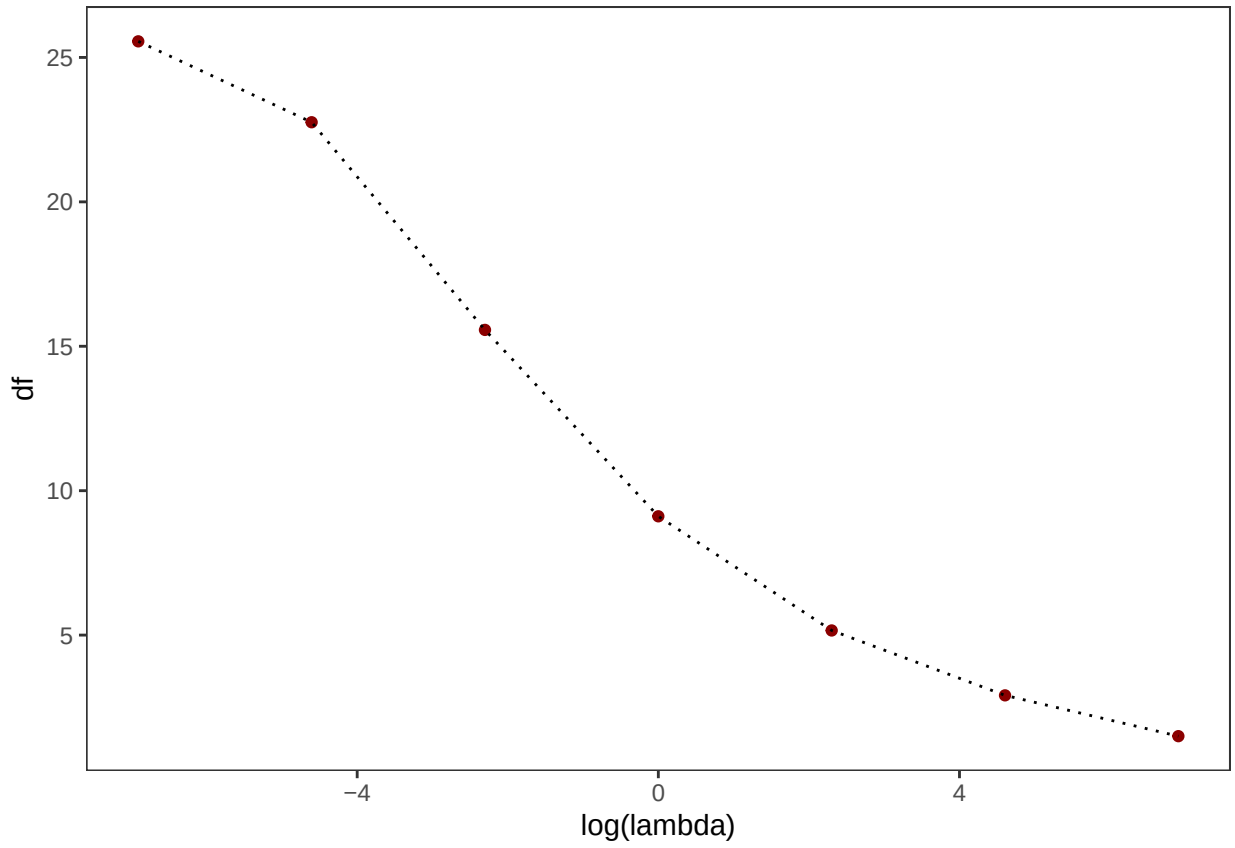
Penalty estimate

```
fbasis_F = create.fourier.basis(c(0, 26), 26)
gcv = rep(0,7)
df = rep(0,7)
sse = rep(0,7)
for(i in 1:7){
  lambda=10^{i-4}
  D2fdPar = fdPar(fbasis_F,Lfdobj=int2Lfd(2),lambda=lambda)
  P_dayeggfd = smooth.basis(day,y,D2fdPar)
  gcv[i] = P_dayeggfd$gcv
  df[i] = P_dayeggfd$df
  sse[i] = P_dayeggfd$SSE
  mainstr = paste('lambda = ',lambda,' df = ',df[i],' gcv = ',gcv[i],sep='')
  #plotfit.fd(y,day,P_dayeggfd$fd,title=mainstr,
             #xlab="Day", ylab= "eggcount",cex.lab=1.5, cex.axis=1.5)
}
par(mfrow=c(1,1),ask=F)
lambda = 10^(-3:3)
res = data.frame(lambda,gcv,df,sse)
```

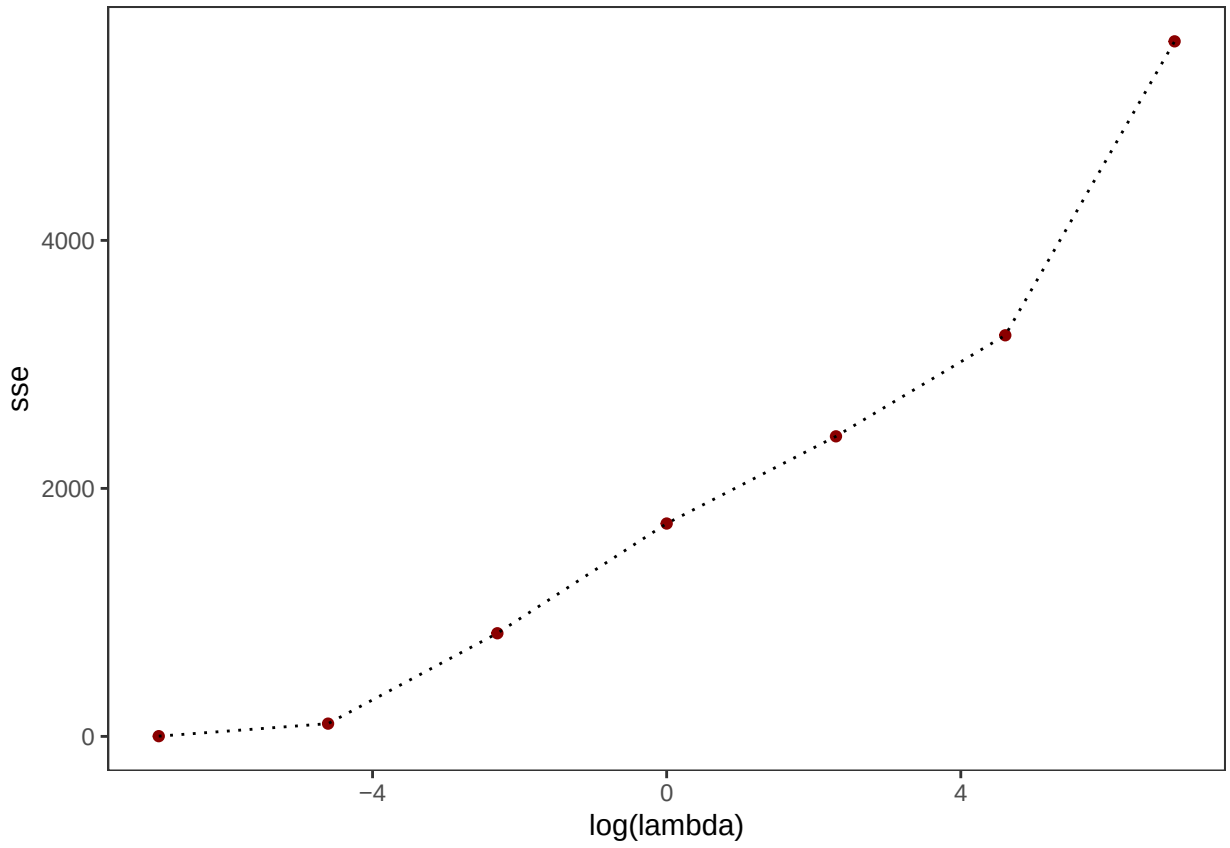
```
ggplot(data=res,aes(log(lambda),gcv))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())
```




```
ggplot(data=res,aes(log(lambda),df))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())
```



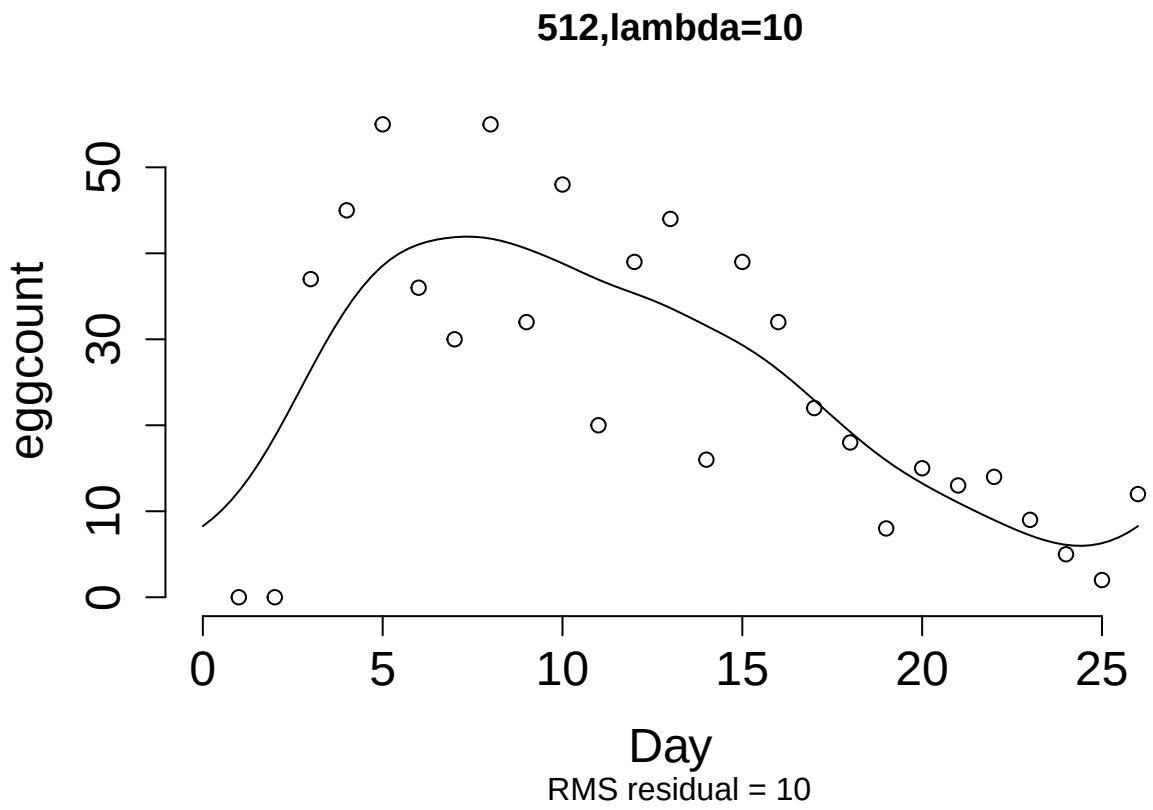

```
ggplot(data=res,aes(log(lambda),sse))+geom_point(color = "darkred")+geom_line(linetype="dotted")+
  theme_bw()+theme(panel.grid=element_blank())
```



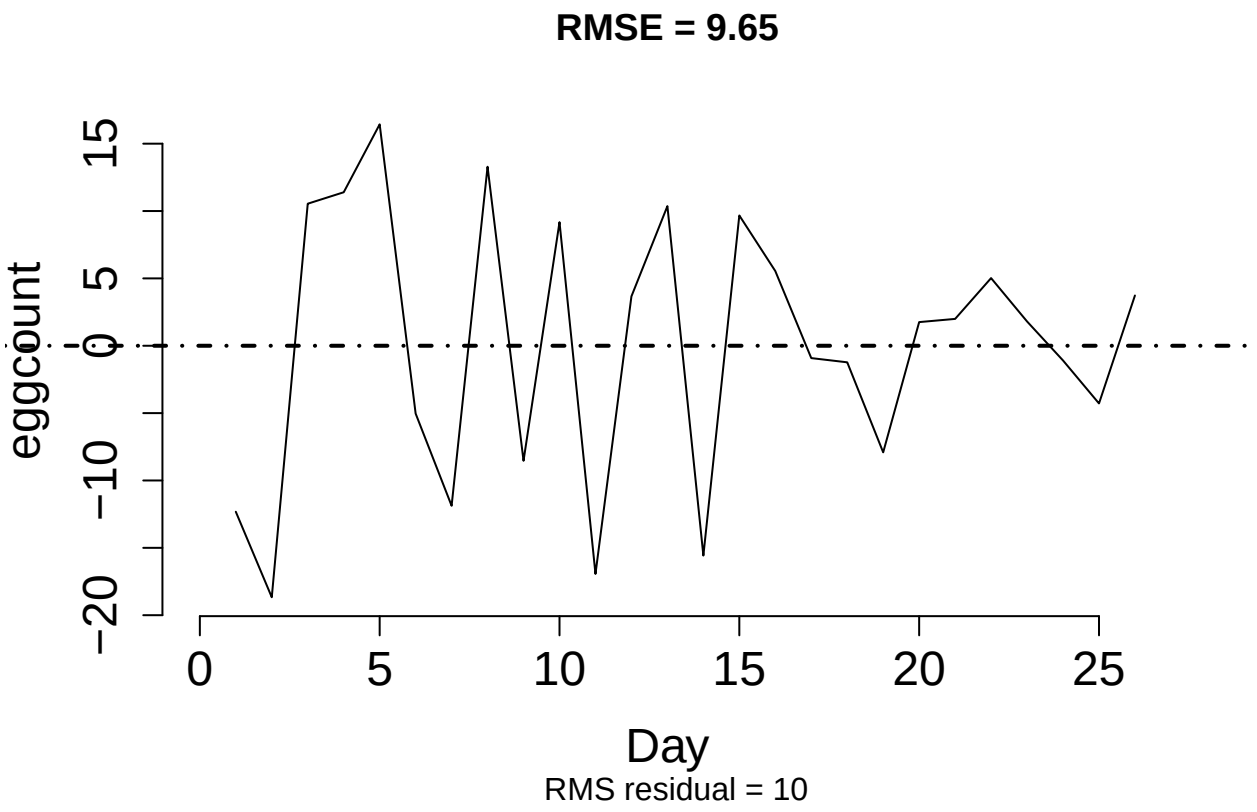

```
which.min(gcv)
```

```
## [1] 5
```

```
par(mfrow=c(1,1), xpd=NA, bty="n")
D2fdPar = fdPar(fbasis_F,Lfdobj=int2Lfd(2),lambda=lambda[5])
P_dayEggfd = smooth.basis(day,y,D2fdPar)
plotfit.fd(y,day,P_dayEggfd$fd,title="512,lambda=10",
           xlab="Day", ylab= "eggcount",cex.lab=1.5, cex.axis=1.5)
```




```
plotfit.fd(y, day, P_dayEggfd$fd, residual = TRUE, titles = 'RMSE = 9.65',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```

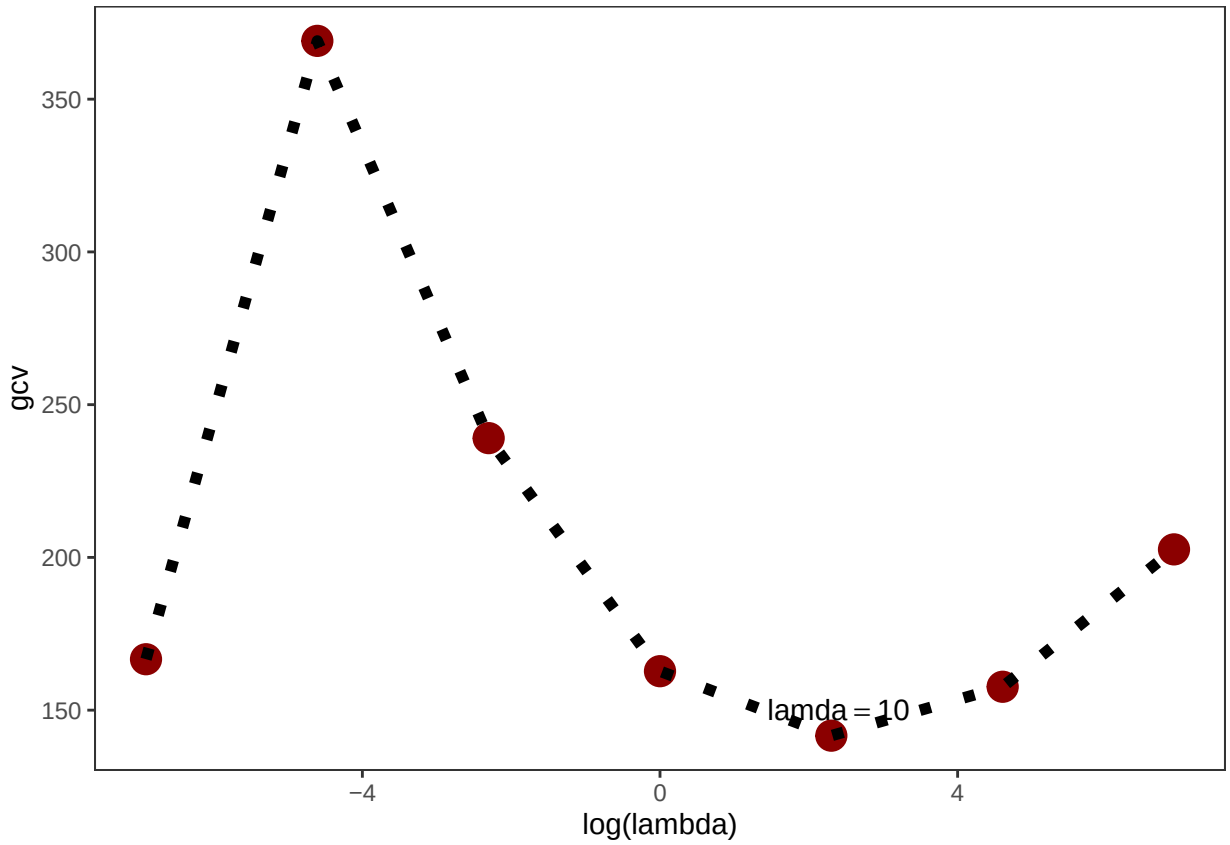



```
P_RMSE = sqrt(mean((eval.fd(day, P_dayEggfd$fd) - y)^2))
print(P_RMSE)
```

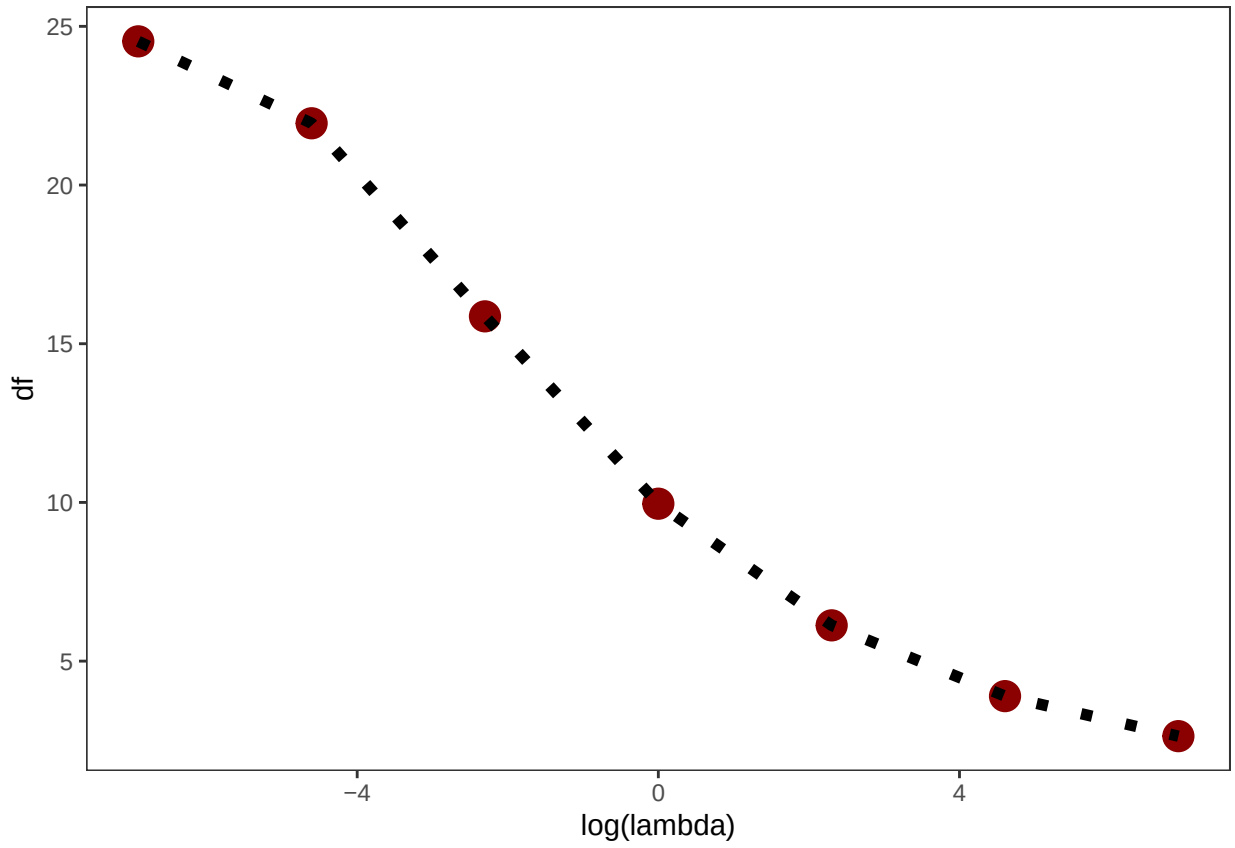
```
## [1] 9.64754
```

```
bbasis_F = create.bspline.basis(rangeval=c(0,26), nbasis = 28, norder=4)
gcv = rep(0,7)
df = rep(0,7)
sse = rep(0,7)
Lfd = int2Lfd(2) # We can use different Lfd(nderiv,a list of fd object)
for(i in 1:7){
  lambda=10^{i-4}
  D2fdPar = fdPar(bbasis_F,Lfdobj=Lfd,lambda=lambda)
  P_dayeggfd = smooth.basis(day,y,D2fdPar)
  gcv[i] = P_dayeggfd$gcv
  df[i] = P_dayeggfd$df
  sse[i] = P_dayeggfd$SSE
  mainstr = paste('lambda = ',lambda,' df = ',df[i],' gcv = ',gcv[i],sep='')
  #plotfit.fd(y,day,P_dayeggfd$fd,title=mainstr,
             # xlab="Day", ylab= "eggcount",cex.lab=1.5, cex.axis=1.5)
}
par(mfrow=c(1,1),ask=F)
lambda = 10^(-3:3)
res = data.frame(lambda,gcv,df,sse)

ggplot(data=res,aes(log(lambda),gcv))+geom_point(color = "darkred",size=5)+geom_line(linetype="dotted",
  theme_bw()+theme(panel.grid=element_blank())+
```

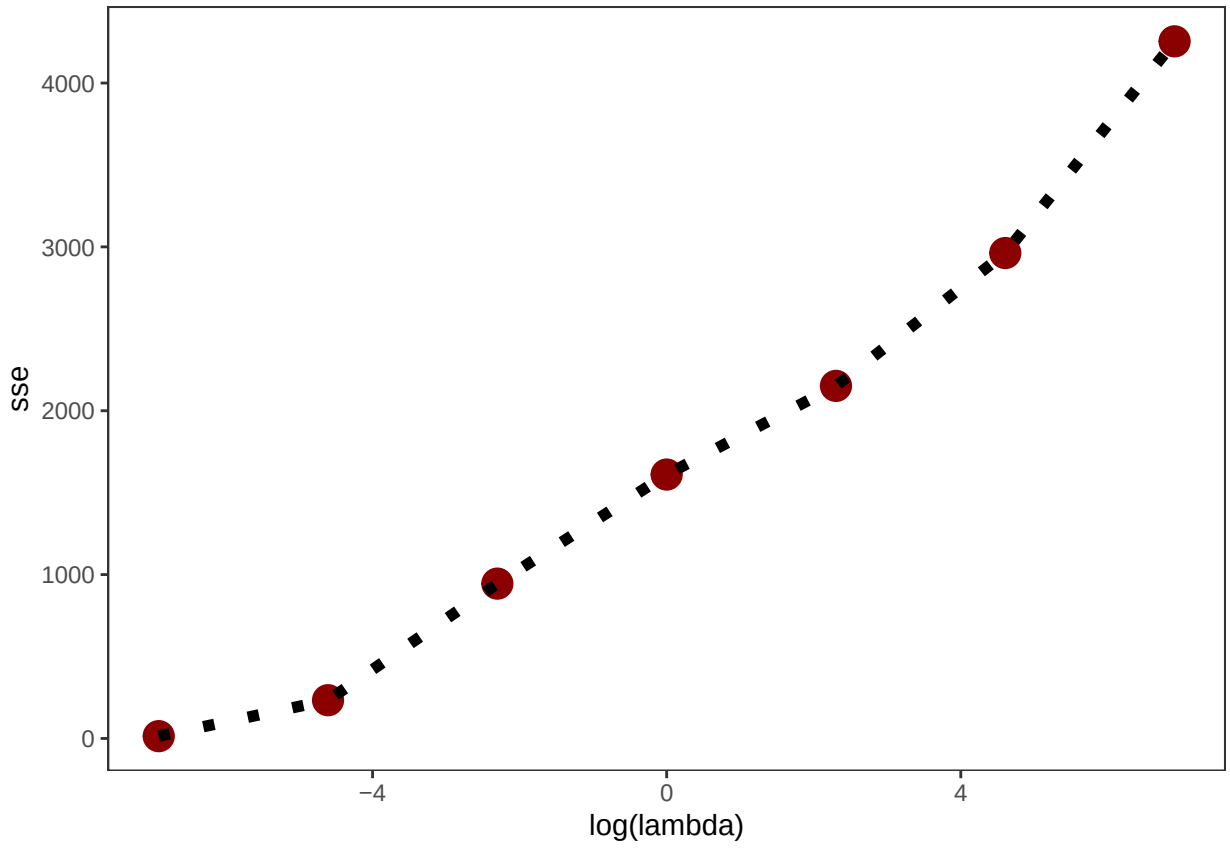
```
  annotate("text",x =2.4, y = 150,parse = T,label="lamda == 10")
```




```
ggplot(data=res,aes(log(lambda),df))+geom_point(color = "darkred",size=5)+geom_line(linetype="dotted",s
  theme_bw()+theme(panel.grid=element_blank())
```



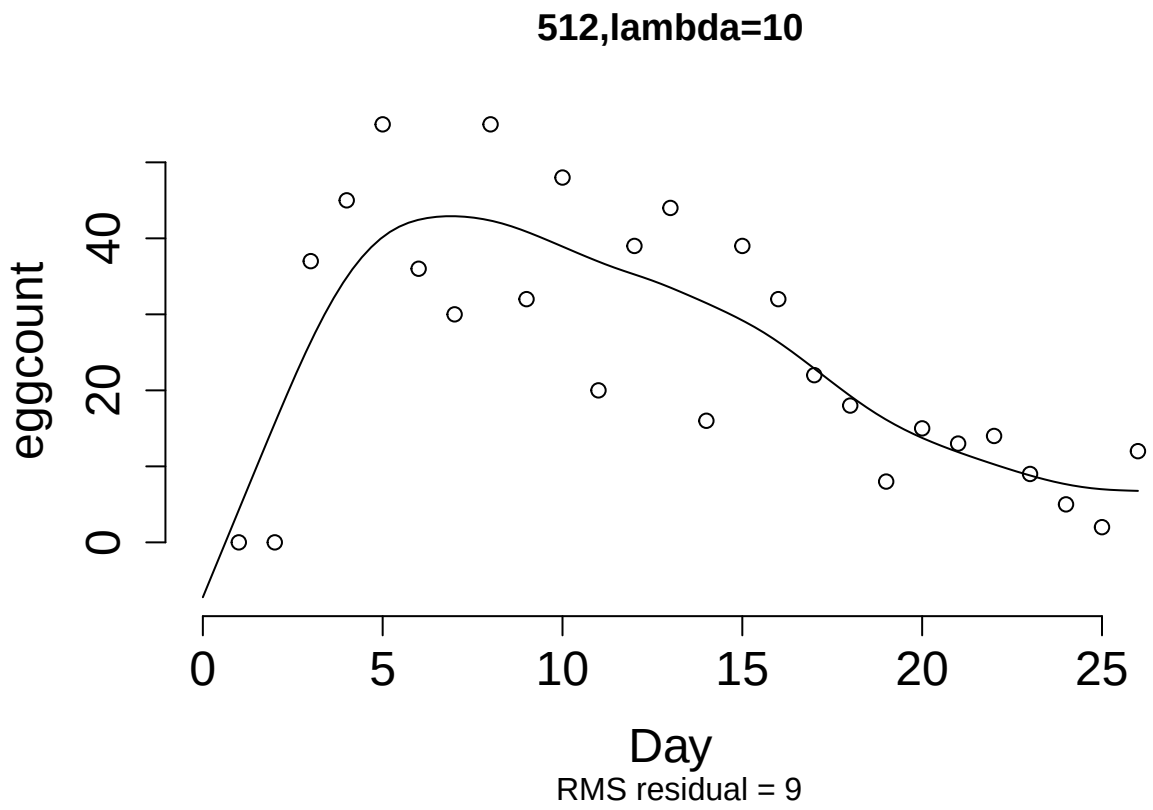

```
ggplot(data=res,aes(log(lambda),sse))+geom_point(color = "darkred",size=5)+geom_line(linetype="dotted",s
  theme_bw()+theme(panel.grid=element_blank())
```



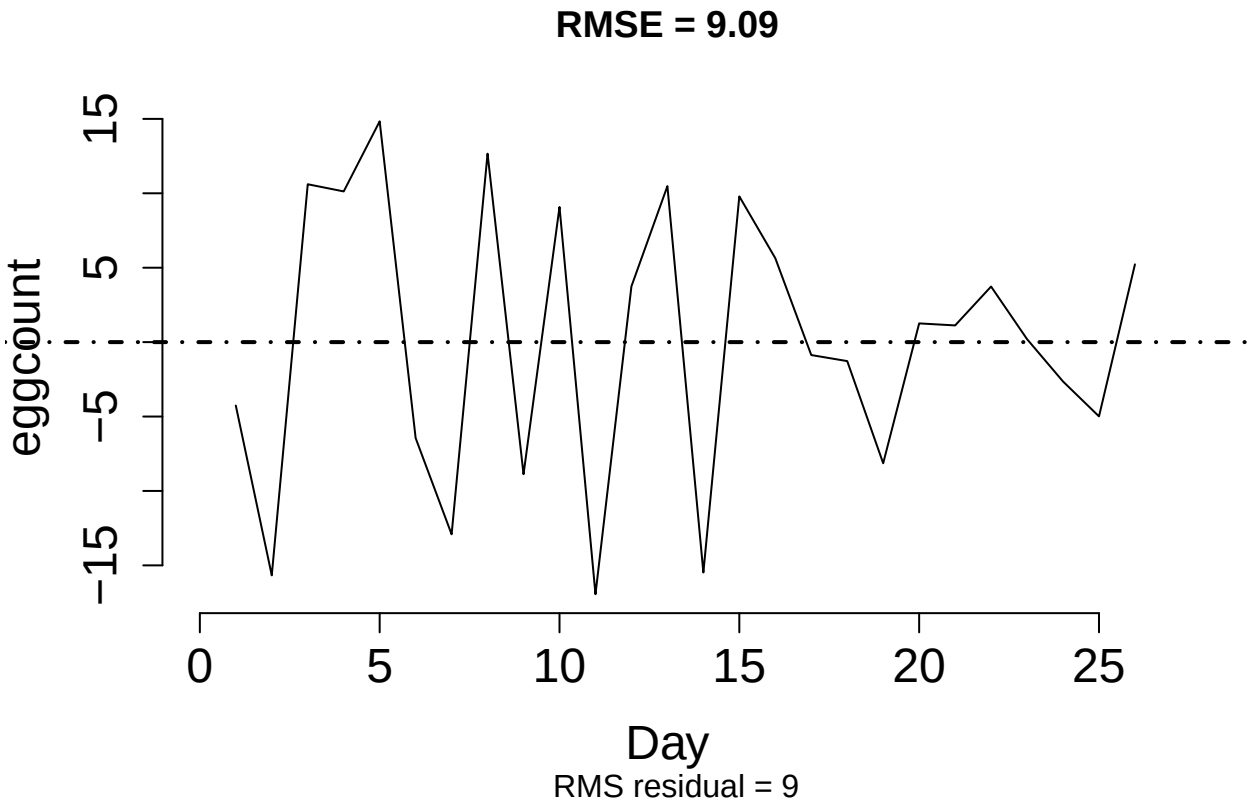

```
which.min(gcv)
```

```
## [1] 5
```

```
par(mfrow=c(1,1), xpd=NA, bty="n")
D2fdPar = fdPar(bbasis_F,Lfdobj=Lfd,lambda=lambda[5]) # functional parameter par
P_dayEggfd = smooth.basis(day,y,D2fdPar)
plotfit.fd(y,day,P_dayEggfd$fd,title="512,lambda=10",
           xlab="Day", ylab= "eggcount",cex.lab=1.5, cex.axis=1.5)
```




```
plotfit.fd(y, day, P_dayEggfd$fd, residual = TRUE, titles = 'RMSE = 9.09',
           xlab="Day", ylab= "eggcount", cex.lab=1.5, cex.axis=1.5)
```




```
P_RMSE = sqrt(mean((eval.fd(day, P_dayEggfd$fd) - y)^2))
print(P_RMSE)
```

```
## [1] 9.09699
```