# Duration of Unemployment - Cubic B-Splines

February 1, 2012

The unemployment data from catdata are loaded.

```
> library(catdata)
> data(unemployment, package="catdata")
> attach(unemployment)

The following object(s) are masked from 'addiction (position 3)':

    age
The following object(s) are masked from 'addiction2':

    age
The following object(s) are masked from 'addiction (position 29)':

    age
The following object(s) are masked from 'unemployment (position 30)':

    age, durbin
The following object(s) are masked from 'medcare':

    age
The following object(s) are masked from 'children':

    age
```

The GAM is fitted by using the library "mgcv".

```
> library(mgcv)
```

Now the response "durbin" is transformed and the GAM is fitted.

```
> durbin[durbin==2] <- 0
> gamage <- gam(durbin ~ s(age, bs="ps", m=c(2,1), k=15), family=binomial())
```

To plot the fitted probabilities for the whole range of age probabilities have
to be predicted.

```
> minage <- min(age)
> maxage <- max(age)
> ageindex <- seq(from=minage, to=maxage, by=0.01)
> pred <- predict(gamage, newdata=data.frame("age"=ageindex), type="response")
```

The following function describes the code for B–Splines.

```
> bspline<-function(x,k,i,m=2)
+ {if (m==-1)
+ {res<-as.numeric(x<k[i+1]&x>=k[i])}
+ else{
+ z0<-(x-k[i])/(k[i+m+1]-k[i])
+ z1<-(k[i+m+2]-x)/(k[i+m+2]-k[i+1])
+ res<- z0*bspline(x,k,i,m-1)+z1*bspline(x,k,i+1,m-1)}
+ res}
```

Now the knots for the B–Splines are defined, furthermore for each age the corresponding mean of durbin is computed.

```
> k <- gamage$smooth[[1]]$knots
> meanage <- c()
> for (i in minage:maxage){
+ meanage[i] <- sum(durbin[age==i])
+ if(sum(durbin[age==i])!=0){
+ meanage[i] <- mean(durbin[age==i])
+ }
+ }
```

Now the line for the predicted probabilities, the B–Splines and the corresponding means for each age are plotted.

```
> par(cex=1.3, lwd=1.5)
> plot(ageindex, pred, type="l", ylim=c(0,0.8),  xlab="age/years", ylab="short-term unempl
> for(i in 1:15)
+ {lines(ageindex,(0.5*gamage$coefficients[i+1]+0.7)*(bspline(x=ageindex,k=k,i=i+1,m=2)),t
> points(minage:maxage, meanage[minage:maxage], pch=20)
```
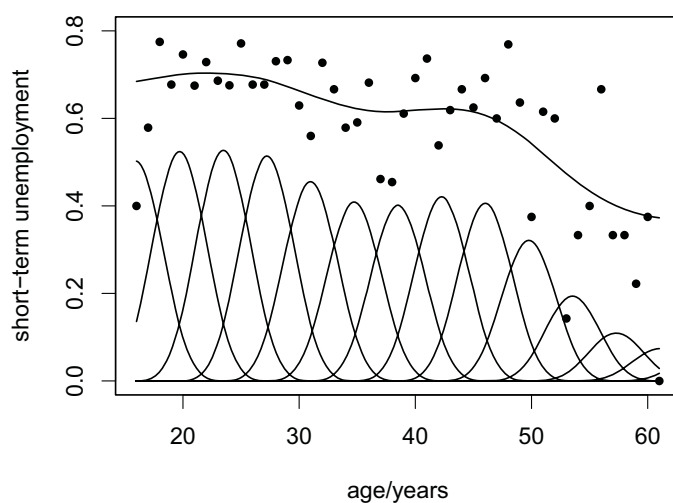
Via the option "fx=TRUE" a unpenalized gam is fitted, afterwards again the probabilities for the whole range of age are computed.

```
> gamage2 <- gam(durbin ~ s(age, bs="ps", fx=TRUE, m=c(2,1),k=15), family=binomial())
> pred2 <- predict(gamage2, newdata=data.frame("age"=ageindex), type="response")
```

Now for the unpenalized GAM the new probabilities, the new B–Splines and again the means are plotted. The fitted line for the probabilities is very wiggly now.

```
> par(cex=1.3, lwd=1.5)
> plot(ageindex, pred2, type="l", ylim=c(0,0.8), xlab="age/years", ylab="short-term unempl
> for(i in 1:15)
+ {lines(ageindex, (0.1*gamage2$coefficients[i+1]+0.3)*bspline(x=ageindex,k=k,i=i+1,m=2),t
> points(minage:maxage, meanage[minage:maxage],pch=20)
```