Chapter 10: Extensions to the GLM

10.1 *Implement a GAM for the Swedish mortality data, for males, using smooth functions for age and year.*

Age and year are standardized as described in Section 4.11, for numerical stability.

```
proc means noprint data=act.mortality;
/* Find min and max of age and year,  for standardization */
var age year;
output out=stats (drop=_type_ _freq_) min=min_age min_year max=max_age max_year;
run;

data mortality;
set act.mortality;
    if _n_=1 then set stats;
    agestd = (age-0.5*(max_age+min_age))/(0.5*(max_age-min_age));
    yearstd = (year-0.5*(max_year+min_year))/(0.5*(max_year-min_year));
    if Female_Exp = 0 then  l_female_exp = 0;
    else  l_female_exp = log(Female_Exp);
    if Male_Exp =0 then  l_male_exp =0;
    else l_male_exp = log(Male_Exp);
run;
```

Spline terms are fitted to age (standardized) and year (standardized), with 15 and 10 degrees of freedom respectively.

```
ods html;
*ods graphics on;                                /* Use this if you just want screen output */
ods graphics on / imagename="c:\glm\mortality" imagefmt=jpeg ;   /*Use this if you want the
                                                                   graphics to go to a file */
proc gam data=mortality;
model male_death = param(l_male_exp) spline(agestd,df=15) spline(yearstd,df=10)
    / dist = poisson ;
run;
ods graphics off;
ods html close;
```

```
                              The GAM Procedure
                        Dependent Variable: Male_death
                  Regression Model Component(s): L_male_exp
               Smoothing Model Component(s): spline(agestd) spline(yearstd)

                              Summary of Input Data Set

                        Number of Observations              5868
                        Number of Missing Observations       237
                        Distribution                     Poisson
                        Link Function                        Log


                           Iteration Summary and Fit Statistics

                    Number of local score iterations                        5
                    Local score convergence criterion            3.722309E-9
                    Final Number of Backfitting Iterations                  1
                    Final Backfitting Criterion                  7.720542E-9
                    The Deviance of the Final Estimate           47870.003542


                  The local score algorithm converged.


                              Regression Model Analysis
                                 Parameter Estimates

                              Parameter       Standard
                  Parameter    Estimate          Error    t Value    Pr > |t|
```

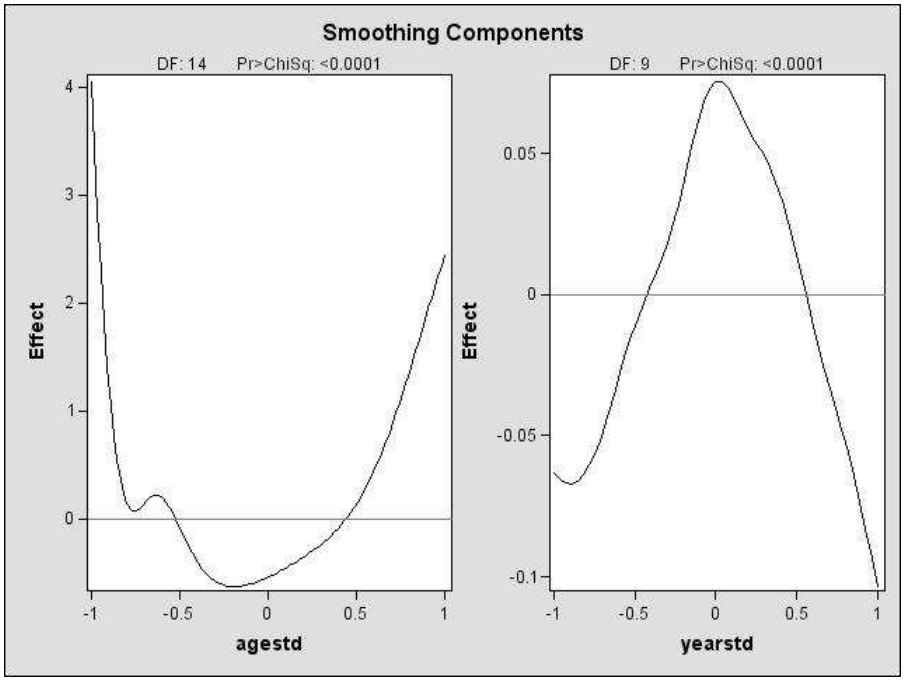
Intercept	-7.17078	0.00935	-767.21	<.0001
L_male_exp	1.25903	0.00087877	1432.73	<.0001
Linear(agestd)	4.89024	0.00295	1660.16	<.0001
Linear(yearstd)	-0.36902	0.00116	-316.84	<.0001

Smoothing Model Analysis
Fit Summary for Smoothing Components

Component	Smoothing Parameter	DF	GCV	Num Unique Obs
Spline(agestd)	0.999980	14.000000	20.575113	110
Spline(yearstd)	0.966366	9.000000	0.000263	55

Smoothing Model Analysis
Analysis of Deviance

Source	DF	Sum of Squares	Chi-Square	Pr > ChiSq
Spline(agestd)	14.00000	278785	278785.169	<.0001
Spline(yearstd)	9.00000	6629.508532	6629.5085	<.0001



The above graphs show the effects of age and year, each after correcting for the other. The y–scale is in residual units.

10.2 *Implement a GAMLSS for the Swedish mortality data.*

We use the negative binomial response distribution. The polynomial models in Chapter 6 are suggestive of smooth functions for both year and age. As there is only one observation for each year–age combination, it is not possible to fit a model for the dispersion parameter κ as well as for the mean μ. We use p–splines (`ps`):

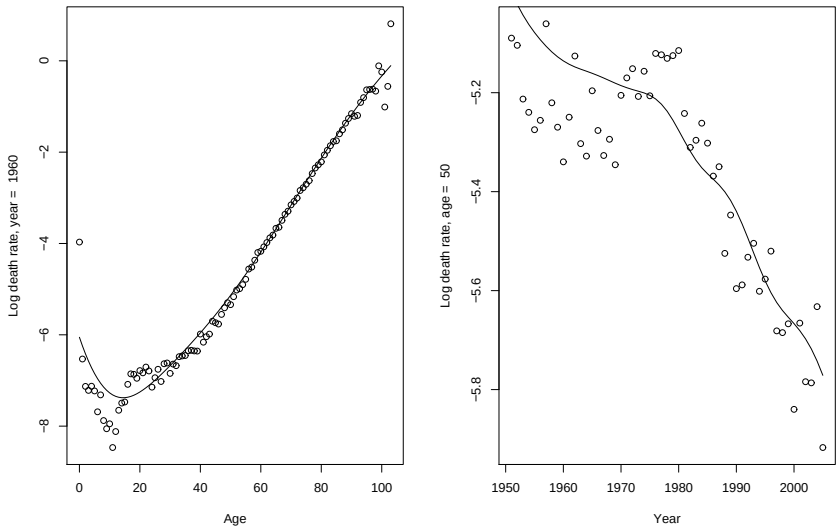
```
> ########### GAMLSS model for Swedish male mortality
>
> mortality <- read.table("mortality.csv",header=T,sep=",")
> mortality <- mortality[,-c(3,5,7,9,11)]
> mortality <- na.omit(mortality)
>
> library(gamlss)
>
> model1 <- gamlss(round(Male_death,0) ~ ps(Year,df=10) + ps(Age,df=4) + offset(L_male_exp),
+ family=NBI, data=mortality)
GAMLSS-RS iteration 1: Global Deviance = 64632.76
GAMLSS-RS iteration 2: Global Deviance = 64567.12
GAMLSS-RS iteration 3: Global Deviance = 64569.46
GAMLSS-RS iteration 4: Global Deviance = 64569.48
GAMLSS-RS iteration 5: Global Deviance = 64569.48
> summary(model1)
******************************************************************
Family:  c("NBI", "Negative Binomial type I")
Call:  gamlss(formula = round(Male_death, 0) ~ ps(Year, df = 10) + ps(Age, df = 4) + offset(L_male_exp),
family = NBI, data = mortality)
Fitting method: RS()

------------------------------------------------------------------
Mu link function:  log
Mu Coefficients:
                   Estimate  Std. Error  t value    Pr(>|t|)
(Intercept)        17.34927   0.6610921    26.24  1.322e-143
ps(Year, df = 10)  -0.01312   0.0003343   -39.26  2.024e-299
ps(Age, df = 4)     0.07632   0.0001810   421.58   0.000e+00

------------------------------------------------------------------
Sigma link function:  log
Sigma Coefficients:
             Estimate  Std. Error  t value  Pr(>|t|)
(Intercept)    -1.915    0.006624   -289.1         0

------------------------------------------------------------------
No. of observations in the fit:  5868
Degrees of Freedom for the fit:  18.00005
      Residual Deg. of Freedom:  5850
                      at cycle:  5

Global Deviance:     64569.48
            AIC:     64605.48
            SBC:     64725.67
******************************************************************
>
######## plot at year=y, age=x
######## y-scale is log(death rate)
> y <- 1960
> x <- 50
>
> par(mfrow=c(1,2))
> plot(Age[Year==y],log(q_male)[Year==y],xlab="Age",
+ ylab=paste("Log death rate, year = ",as.character(y)))
> lines(Age[Year==y],(log(fitted(model1))-L_male_exp)[Year==y])
> plot(Year[Age==x],log(q_male)[Age==x],xlab="Year",
+ ylab=paste("Log death rate, age = ",as.character(x)))
> lines(Year[Age==x],(log(fitted(model1))-L_male_exp)[Age==x])
```



- If you plot later years you will see that the fit gets worse between ages 0 and 20.
- The degrees of freedom of the splines may be chosen using the AIC or BIC (SBC).
- In general, splines are preferable to polynomials as they are far more robust.

10.3 *Investigate the use of a GAM for the third party claims data, using a smooth function for accidents (or log accidents).*

A comprehensive analysis of these data, using the R package `gamlss`, is presented in the article

Stasinopoulos DM and Rigby RA (2007). Generalized Additive Models for Location Scale and Shape (GAMLSS) in R, *Journal of Statistical Software*, 23(7), ??-??.

which is downloadable at
`http://www.jstatsoft.org/v23/i07`

The final model chosen is the negative binomial response distribution, with log link and explanatory variables for μ:

- statistical division
- cubic spline of log population density
- log killed/injured
- cubic spline of log accidents
- cubic spline of log population.

For the model for κ, the variables log population, log killed/injured and log accidents were chosen using the AIC. However, the authors felt that this was overly complex.

We replicate the model for μ using `proc gam`, in SAS. Note that `proc gam` does not have the negative binomial response distribution (in version 9.1.3), so we use the Poisson distribution.

```
ods html;
  ods graphics on / imagename="c:\lgaplot" imagefmt=ps ;
proc gam data=lgaclaims;
class sd;
model claims = param(sd l_ki) spline(l_popdens,df=5) spline(l_accidents, df=5) spline(l_population,df=5)
   / dist=POISSON;
run;
ods graphics off; ods html close;
```

The GAM Procedure
Dependent Variable: Claims
Regression Model Component(s): SD l_ki
Smoothing Model Component(s): spline(l_popdens) spline(L_Accidents) spline(L_Population)

Summary of Input Data Set

Number of Observations	176
Number of Missing Observations	0
Distribution	Poisson
Link Function	Log

Class Level Information

Class	Levels	Values
SD	13	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

Iteration Summary and Fit Statistics

Number of local score iterations	4
Local score convergence criterion	3.6237965E-9
Final Number of Backfitting Iterations	1
Final Backfitting Criterion	8.1582487E-9
The Deviance of the Final Estimate	4179.5945246

The local score algorithm converged.

Regression Model Analysis
Parameter Estimates

Parameter	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	0.09839	0.08081	1.22	0.2254
SD 1	-0.49465	0.03873	-12.77	<.0001
SD 2	-0.51202	0.03868	-13.24	<.0001
SD 3	-0.52968	0.03785	-13.99	<.0001
SD 4	-0.62151	0.03857	-16.12	<.0001
SD 5	-0.85535	0.04205	-20.34	<.0001
SD 6	-0.57857	0.03855	-15.01	<.0001
SD 7	-0.81425	0.04312	-18.88	<.0001
SD 8	-0.36320	0.07604	-4.78	<.0001
SD 9	-0.50497	0.03872	-13.04	<.0001
SD 10	-0.71342	0.04142	-17.22	<.0001
SD 11	-0.41288	0.04230	-9.76	<.0001
SD 12	-0.97334	0.04562	-21.34	<.0001
SD 13	0	.	.	.
l_ki	1.03690	0.03084	33.63	<.0001

Dependent Variable: Claims
Regression Model Component(s): SD l_ki
Smoothing Model Component(s): spline(l_popdens) spline(L_Accidents) spline(L_Population)

Regression Model Analysis
Parameter Estimates

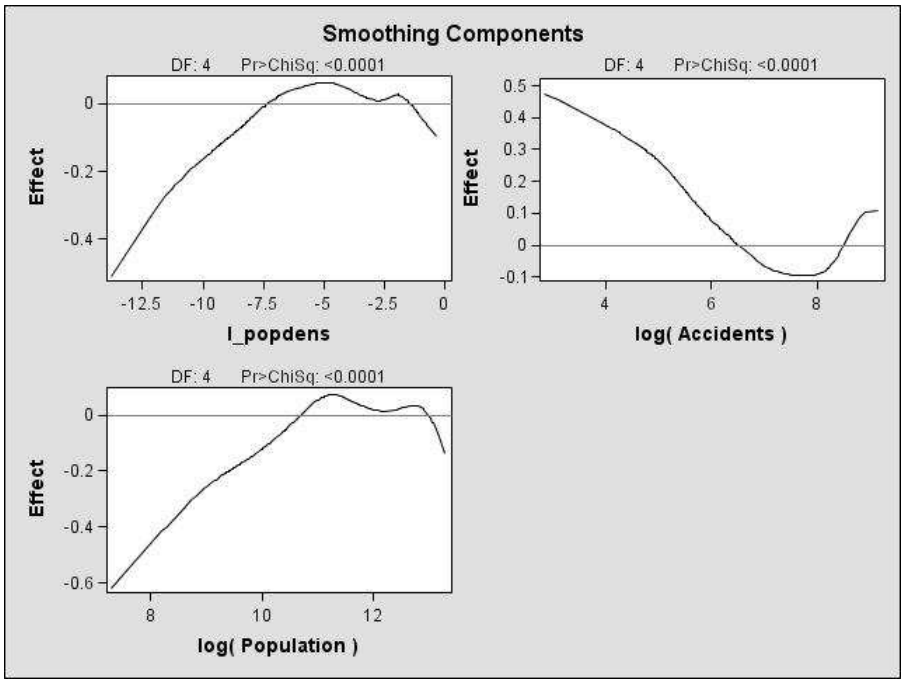
Parameter	Parameter Estimate	Standard Error	t Value	Pr > |t|
Linear(l_popdens)	0.10517	0.00317	33.23	<.0001
Linear(L_Accidents)	-0.06796	0.03093	-2.20	0.0296
Linear(L_Population)	0.07914	0.01010	7.84	<.0001

Smoothing Model Analysis
Fit Summary for Smoothing Components

Component	Smoothing Parameter	DF	GCV	Num Unique Obs
Spline(l_popdens)	0.999996	4.000000	0.770823	164
Spline(L_Accidents)	0.999998	4.000000	1.658074	164
Spline(L_Population)	0.999998	4.000000	1.617644	170

Smoothing Model Analysis
Analysis of Deviance

Source	DF	Sum of Squares	Chi-Square	Pr > ChiSq
Spline(l_popdens)	4.00000	184.733660	184.7337	<.0001
Spline(L_Accidents)	4.00000	354.733506	354.7335	<.0001
Spline(L_Population)	4.00000	412.680318	412.6803	<.0001



10.4 *In the Enterprise Miner data set, develop a statistical model for claim size, for all policies.*

The same models that were determined for claim occurrence and claim amount, in Chapter 7 and 8 solutions, have been used here. In addition, the model for claim amount specifies a formula for the dispersion parameter σ. The AIC or BIC (SBC) may be used for model selection (the process is not shown here).

```
> car <- read.table(""claims_sas_miner.csv"",sep="","",header=T)
> attach(car)
>
> ### Perform changes as in Chapter 7 and 8 solutions
>
> bluebk <- bluebook/1000-14.2
> npolicy1 <- 1*(npolicy==1)
>
> ### Need to categorise income, because of the frequency spike at zero
> cut <- c(0,25000,50000,75000,100000,500000)
> r <- length(cut)
> incomecat <- 1*(income==0)
> for(i in 2:r)incomecat <- incomecat + i*(income>cut[i-1]&income<=cut[i])
> table(incomecat)
incomecat
   1    2    3    4    5    6
 797 1423 2304 2101 1347 1761
>
> ### Need to categorise oldclaim, because of the frequency spike at zero
> cut <- c(0,5000,10000,58000)
> r <- length(cut)
> oldclaimcat <- 1*(oldclaim==0)
> for(i in 2:r)oldclaimcat <- oldclaimcat + i*(oldclaim>cut[i-1]&oldclaim<=cut[i])
> table(oldclaimcat)
oldclaimcat
   1    2    3    4
6293 1584 1463  963
>
> ### Need to categorise mvr_pts
> cut <- c(0,1,2,3,6,13)
> r <- length(cut)
> mvrcat <- 1*(mvr_pts==0)
> for(i in 2:r)mvrcat <- mvrcat + i*(mvr_pts>cut[i-1]&mvr_pts<=cut[i])
> table(mvrcat)
mvrcat
   1    2    3    4    5    6
4659 1467 1199  966 1596  416
>
> car <- cbind(car,bluebk,npolicy1,incomecat,oldclaimcat,mvrcat)
>
> library(gamlss)
Loading required package: splines
 ********** GAMLSS Version 1.5-0 **********
For more on GAMLSS look at http://www.londonmet.ac.uk/gamlss/
Type gamlssNews() to see new features/changes/bug fixes.
>
> ##### ZAIG model
> ## use models for clm_flag and clm_amt as determined in Chapters
> ## 7 and 8 solutions.
> ## Use cubic splines (cs) instead of quadratic terms.
>
> model1 <- gamlss(clm_amt ~ cs(bluebk) + npolicy1 + married ,
+ sigma.fo=~ npolicy1 + car_type +
+ factor(clm_freq)+ factor(incomecat)+max_educ  ,
+ nu.fo =~ cs(kidsdriv) + car_use + cs(bluebk) + cs(retained) + car_type + factor(oldclaimcat) + revoked +
+ factor(mvrcat) + factor(incomecat) + married + parent1 + max_educ + density + cs(travtime) + cs(age),
+ family=ZAIG,
+ data=na.omit(car))
GAMLSS-RS iteration 1: Global Deviance = 48349.1
GAMLSS-RS iteration 2: Global Deviance = 48348.29
GAMLSS-RS iteration 3: Global Deviance = 48348.32
GAMLSS-RS iteration 4: Global Deviance = 48348.32
GAMLSS-RS iteration 5: Global Deviance = 48348.32
>
```

```
> summary(model1)
******************************************************************
Family:  c(""ZAIG"", ""Zero adjusted IG"")
Call:  gamlss(formula = clm_amt ~ cs(bluebk) + npolicy1 + married,
sigma.formula = ~npolicy1 + car_type + factor(clm_freq) + factor(incomecat) + max_educ,
nu.formula = ~cs(kidsdriv) + car_use + cs(bluebk) + cs(retained) +
car_type + factor(oldclaimcat) + revoked + factor(mvrcat) +
factor(incomecat) + married + parent1 + max_educ + density +
cs(travtime) + cs(age), family = ZAIG, data = na.omit(car))
Fitting method: RS()

------------------------------------------------------------------
Mu link function:  log
Mu Coefficients:
             Estimate  Std. Error  t value   Pr(>|t|)
(Intercept)   8.80001    0.038369  229.351  0.000e+00
cs(bluebk)    0.03098    0.002435   12.722  1.000e-36
npolicy1     -0.09488    0.041008   -2.314  2.070e-02
marriedYes   -0.09660    0.039334   -2.456  1.407e-02

------------------------------------------------------------------
Sigma link function:  log
Sigma Coefficients:
                      Estimate  Std. Error    t value   Pr(>|t|)
(Intercept)          -4.330465     0.08278  -52.31196  0.000e+00
npolicy1             -0.137978     0.03053   -4.51923  6.294e-06
car_typePickup        0.242075     0.06398    3.78345  1.558e-04
car_typeSedan         0.120563     0.06554    1.83957  6.587e-02
car_typeSports Car    0.145754     0.06750    2.15923  3.086e-02
car_typeSUV           0.026372     0.06102    0.43217  6.656e-01
car_typeVan           0.044269     0.07305    0.60599  5.445e-01
factor(clm_freq)1    -0.175097     0.04287   -4.08470  4.456e-05
factor(clm_freq)2     0.080962     0.04006    2.02082  4.333e-02
factor(clm_freq)3     0.070382     0.04527    1.55464  1.201e-01
factor(clm_freq)4     0.131853     0.08516    1.54831  1.216e-01
factor(clm_freq)5    -1.024546     0.31809   -3.22092  1.283e-03
factor(incomecat)2   -0.082991     0.05790   -1.43328  1.518e-01
factor(incomecat)3   -0.034793     0.05329   -0.65295  5.138e-01
factor(incomecat)4    0.103121     0.05633    1.83076  6.717e-02
factor(incomecat)5    0.115370     0.06767    1.70489  8.825e-02
factor(incomecat)6   -0.013982     0.07312   -0.19121  8.484e-01
max_educBachelors     0.003687     0.05202    0.07088  9.435e-01
max_educHigh School  -0.041434     0.04487   -0.92348  3.558e-01
max_educMasters      -0.205087     0.06157   -3.33092  8.695e-04
max_educPhD          -0.107093     0.08526   -1.25609  2.091e-01

------------------------------------------------------------------
Nu link function:  logit
Nu Coefficients:
                        Estimate  Std. Error    t value   Pr(>|t|)
(Intercept)             1.683014    0.318167    5.28971  1.257e-07
cs(kidsdriv)           -0.533730    0.058131   -9.18146  5.313e-20
car_usePrivate          0.857910    0.076315   11.24170  4.173e-29
cs(bluebk)              0.022939    0.004737    4.84226  1.307e-06
cs(retained)            0.057972    0.007459    7.77169  8.693e-15
car_typePickup          0.032838    0.145191    0.22617  8.211e-01
car_typeSedan           0.615654    0.147494    4.17410  3.023e-05
car_typeSports Car     -0.355077    0.164456   -2.15910  3.087e-02
car_typeSUV            -0.113101    0.149858   -0.75472  4.504e-01
car_typeVan            -0.132159    0.147713   -0.89470  3.710e-01
factor(oldclaimcat)2   -0.564325    0.085870   -6.57188  5.275e-11
factor(oldclaimcat)3   -0.593060    0.088347   -6.71286  2.037e-11
factor(oldclaimcat)4   -0.085157    0.112256   -0.75860  4.481e-01
revokedYes             -0.904930    0.095863   -9.43978  4.789e-21
factor(mvrcat)2        -0.118809    0.094412   -1.25841  2.083e-01
factor(mvrcat)3        -0.206981    0.099794   -2.07409  3.810e-02
factor(mvrcat)4        -0.208464    0.106143   -1.96399  4.957e-02
factor(mvrcat)5        -0.243594    0.090113   -2.70320  6.882e-03
factor(mvrcat)6        -0.762581    0.144516   -5.27679  1.349e-07
factor(incomecat)2      0.355693    0.126362    2.81486  4.892e-03
factor(incomecat)3      0.457805    0.117650    3.89123  1.005e-04
factor(incomecat)4      0.495973    0.122281    4.05600  5.039e-05
factor(incomecat)5      0.747976    0.139230    5.37223  7.993e-08
```

```
factor(incomecat)6     0.859446    0.144436    5.95035  2.787e-09
marriedYes             0.744901    0.071448   10.42584  2.729e-25
parent1Yes            -0.273761    0.101810   -2.68894  7.182e-03
max_educBachelors      0.512845    0.106960    4.79473  1.658e-06
max_educHigh School    0.010165    0.097942    0.10379  9.173e-01
max_educMasters        0.545003    0.119502    4.56060  5.176e-06
max_educPhD            0.642489    0.155668    4.12729  3.707e-05
densityHighly Urban   -2.955383    0.211383  -13.98117  6.549e-44
densityRural          -0.005446    0.236007   -0.02308  9.816e-01
densityUrban          -1.750987    0.211516   -8.27829  1.449e-16
cs(travtime)          -0.014992    0.001982   -7.56434  4.325e-14
cs(age)                0.009217    0.003631    2.53820  1.116e-02

---------------------------------------------------------------------
No. of observations in the fit:  8163
Degrees of Freedom for the fit:  78.0033
      Residual Deg. of Freedom:  8084.997
                      at cycle:  5

Global Deviance:     48348.32
            AIC:     48504.33
            SBC:     49050.92
*********************************************************************
```