Outline

P-splines and GLAM

Specific mortality applications

MortalitySmooth: an R package for smoothing mortality

Conclusions

P-splines for Normal data

In case of a univariate response, normally distributed, we aim to estimate a smooth function, \( \hat{f} : Y = f(X) + \epsilon \)
**P-splines for Normal data**

- In case of a univariate response, normally distributed, we aim to estimate a smooth function, \( \hat{f} : Y = f(X) + \epsilon \)
- In a regression setting, following Eilers and Marx (1996)
  - construct a base with large number of \( B \)-splines with associate coefficients: \( B\theta \)
  - put a penalty to force the coefficients to vary smoothly: \( P \)

\[
\hat{\theta} = (B'B + P)^{-1}B'y \quad \text{where} \quad P = \lambda D_dD_d
\]

The smoothing parameter \( \lambda \) balances model fidelity and smoothness of the parameter estimates.

**P-splines in action**

Simulated data and \( B \)-splines

Simulated data and \( B \)-splines with unpenalized coefficients
**P-splines in action**

![Simulated data and B-splines with penalized coefficients, \( \lambda = 10 \)](image)

**P-splines for Poisson data**

- Introducing the concept of a link function, we can smoothly model Poisson counts adapting the iterative reweighted least squares algorithm:

\[
(B'\tilde{W}B + P)\tilde{\theta} = B'\tilde{W}^{-1}(y - \bar{\mu}) + B\tilde{\theta}
\]


**P-splines for Poisson data**

- Introducing the concept of a link function, we can smoothly model Poisson counts adapting the iterative reweighted least squares algorithm:

\[
(B'\tilde{W}B + P)\tilde{\theta} = B'\tilde{W}^{-1}(y - \bar{\mu}) + B\tilde{\theta}
\]

- And hat matrix:

\[
H_\lambda = B(B'\tilde{W}B + P)^{-1}B'\tilde{W}
\]
P-splines for Poisson data

Introducing the concept of a link function, we can smoothly model Poisson counts adapting the iterative reweighted least squares algorithm:

\[(B'\hat{W}B + P)\hat{\theta} = B'\hat{W}^{-1}(y - \hat{\mu}) + B\hat{\theta}\]

And hat matrix: \[H_\lambda = B(B'\hat{W}B + P)^{-1}B'\hat{W}\]

To choose the optimal \(\lambda\) we can alternatively minimize:

\[\text{AIC}(\lambda) = \text{Dev}(y; \theta, \lambda) + 2 \cdot \text{ED}(\theta, \lambda)\]
\[\text{BIC}(\lambda) = \text{Dev}(y; \theta, \lambda) + \log(n) \cdot \text{ED}(\theta, \lambda)\]

We define the effective dimension to \[\text{ED}(\theta, \lambda) = \text{tr}(H_\lambda)\] (Hastie and Tibshirani, 1990, p. 52)

P-splines on mortality data

Age 40, 1930–2010


P-splines for Poisson data

Introducing the concept of a link function, we can smoothly model Poisson counts adapting the iterative reweighted least squares algorithm:

\[(B'\hat{W}B + P)\hat{\theta} = B'\hat{W}^{-1}(y - \hat{\mu}) + B\hat{\theta}\]

And hat matrix: \[H_\lambda = B(B'\hat{W}B + P)^{-1}B'\hat{W}\]

To choose the optimal \(\lambda\) we can alternatively minimize:

\[\text{AIC}(\lambda) = \text{Dev}(y; \theta, \lambda) + 2 \cdot \text{ED}(\theta, \lambda)\]
\[\text{BIC}(\lambda) = \text{Dev}(y; \theta, \lambda) + \log(n) \cdot \text{ED}(\theta, \lambda)\]

P-splines on mortality data

![Graph showing log-mortality over years for age 40, 1930-2010]

Actual and fitted death rates over years by P-splines with AIC and BIC optimal $\lambda$, logarithmic scale. Age 40. Denmark, females, 1930-2010.

P-splines for mortality surface

- We arrange both death and exposures matrices in columns
- The regression matrix for our two-dimensional model is the Kronecker product $B = B_y \otimes B_a$, where $a$ and $y$ stand for applied age and year dimensions

P-splines for mortality surface

- We arrange both death and exposures matrices in columns
- The regression matrix for our two-dimensional model is the Kronecker product $B = B_y \otimes B_a$, where $a$ and $y$ stand for applied age and year dimensions
- $B$ has an associated vector of regression coefficients $a$, which can be arranged in a matrix $\Theta$
**P-splines for mortality surface**

- We arrange both death and exposures matrices in columns.
- The regression matrix for our two-dimensional model is the Kronecker product $B = B_y \otimes B_a$, where $a$ and $y$ stand for applied age and year dimensions.
- $B$ has an associated vector of regression coefficients $a$, which can be arranged in a matrix $\Theta$.
- We penalized the coefficients to each of the columns and rows of $\Theta$:
  $$P = \lambda_a I_c y \otimes D'_a D_a + \lambda_y D'_y D_y \otimes I_c a$$

$\lambda_a$ and $\lambda_y$ are the smoothing parameters used for age and year, respectively (Currie et al. 2004).

$\lambda_a$ and $\lambda_y$ are the smoothing parameters used for age and year, respectively (Currie et al. 2004).

- To choose the optimal $(\lambda_a, \lambda_y)$-combination we can minimize both AIC and BIC.
Outline

- P-splines and GLAM
- Specific mortality applications
- MortalitySmooth: an R package for smoothing mortality
- Conclusions

Two-dimensional basis

The GLAM algorithm: the equations

- Issues:
  - \( \mathbf{B} = \mathbf{B}_y \otimes \mathbf{B}_a \) is huge: \( n \times m \times c \times c_y \)
  - construction of \( \mathbf{B} \) is expensive
  - and so computing the inner product, \( \mathbf{B}' \mathbf{W} \mathbf{B} \)

- Solution: the Generalized Linear Array Models (Currie et al. 2006)
The GLAM algorithm: the equations

- Issues:
  - \( B = B_y \otimes B_a \) is huge: \( n m \times c_y c_a \)
  - construction of \( B \) is expensive
  - and so computing the inner product, \( B \cdot \tilde{W} \)
- Solution: the Generalized Linear Array Models (Currie et al. 2006)

We define row-tensor product of \( X, n \times c \):

\[
G(X) = \begin{bmatrix} X \otimes 1' \end{bmatrix} \cdot \begin{bmatrix} 1' \otimes X \end{bmatrix}, \quad n \times c^2
\]

The GLAM algorithm: looking at the advantages

```r
> a <- 1:50; m <- length(a)
> y <- 1:100; n <- length(y)
> Ba <- bbase(x=a, xl=min(x), xr=max(x), ndx=7, deg=3)
> By <- bbase(x=y, xl=min(y), xr=max(y), ndx=17, deg=3)
> ka <- ncol(Ba); ky <- ncol(By) ## 10 and 20
> B <- kronecker(By, Ba) # 5000 x 200
> w <- runif(m*n)
> system.time(tBWB1 <- t(B) %*% (w*B))
user  system elapsed
0.184 0.012 0.195
```

```
> Ba1 <- kronecker(matrix(1, ncol=ka, nrow=1), Ba)
> Ba2 <- kronecker(Ba, matrix(1, ncol=ka, nrow=1))
> By1 <- kronecker(matrix(1, ncol=ky, nrow=1), By)
> By2 <- kronecker(By, matrix(1, ncol=ky, nrow=1))
> Ga <- Ba1 * Ba2
> Gy <- By1 * By2
> W <- matrix(w, m, n)
> system.time(tBWB2 <- GLAMinner(Ga, Gy, ka, ky, W))
user  system elapsed
0.004 0.000 0.005
```

The GLAM algorithm: looking at the advantages

```r
> a <- 1:50 ; m <- length(a)
> y <- 1:100 ; n <- length(y)
> Ba <- bbase(x=a, xl=min(x), xr=max(x), ndx=7, deg=3)
> By <- bbase(x=y, xl=min(y), xr=max(y), ndx=17, deg=3)
> ka <- ncol(Ba) ; ky <- ncol(By) ## 10 and 20
> B <- kronecker(By, Ba) ## 5000 x 200
> w <- runif(m*n)
> system.time(tBWB1 <- t(B) %*% (w*B))

user  system   elapsed
 0.184  0.012   0.195
```

```r
> Ba1 <- kronecker(matrix(1, ncol=ka, nrow=1), Ba)
> Ba2 <- kronecker(Ba, matrix(1, ncol=ka, nrow=1))
> By1 <- kronecker(matrix(1, ncol=ky, nrow=1), By)
> By2 <- kronecker(By, matrix(1, ncol=ky, nrow=1))
> Ga <- Ba1 * Ba2
> Gy <- By1 * By2
> Wbreve <- matrix(w, m, n)
> system.time(tBWB2 <- GLAMinner(Ga, Gy, ka, ky, Wbreve))
```

```r
user  system   elapsed
 0.004  0.000   0.005
```

```r
> size1 <- object.size(list(B, w))
> size2 <- object.size(list(Ga, Gy, ka, ky, Wbreve))
> size1/size2

[20.0775316139601 bytes]
```

Actual and fitted death rates from 2D smoothing with P-splines.
Ages from 10 to 100. Denmark, females, 1960-2006

Actual death rates (left panel) and two-dimensional BIC profile for 2D smoothing with P-splines.
Ages from 10 to 100. Denmark, females, 1960-2006

Actual and fitted death rates from 2D smoothing with P-splines.
Ages from 10 to 100. Denmark, females, 1960-2006
Outline

P-splines and GLAM

Specific mortality applications

MortalitySmooth: an R package for smoothing mortality

Conclusions

Small populations: Iceland

Actual and fitted death rates from 2D smoothing with P-splines. Ages from 10 to 95. Iceland, males, 1981-2007

1D vs. 2D perspective for small population: Iceland

Actual and fitted death rates from 1D and 2D smoothing with P-splines. Age 75 over years 1981-2007 (left panel). Year 1990 over ages 10-95 (right panel). Iceland, males.

Cause-specific mortality: Suicides among French males

Suicide rates over ages and years, logarithmic scale. Five-years age-group from 17 to 97. France, males, 1970-1999. Source: INED.
Cause-specific mortality: Suicides among French males

Actual and fitted suicide rates from 2D smoothing with P-splines, logarithmic scale.

Forecasting mortality

- We treat the forecasting of future values as a missing value problem and estimate the fitted and forecast values simultaneously.

Actual and fitted suicide rates over years for selected ages, logarithmic scale.
Forecasting mortality

- We treat the forecasting of future values as a missing value problem and estimate the fitted and forecast values simultaneously.
- Keeping the same age range and forecasting over the years, we augment the data and the \( B \)-spline bases:

\[
\tilde{Y} = [Y : Y^1], \quad \tilde{E} = [E : E^1] \quad \text{and} \quad \tilde{B} = [B_y : B_y^1] \otimes B_a
\]

where the superscript 1 denotes the period we aim to forecast.

Forecasting Danish mortality

- Actual death rates over years for selected ages, logarithmic scale. Denmark, females, 1950-2006.

ISNPS, July 2015 Camarda C.G. (INED) | Smoothing on grids with applications to mortality data

ISNPS, July 2015 Camarda C.G. (INED) | Smoothing on grids with applications to mortality data
Forecasting Danish mortality

Actual, fitted and forecast death rates and 95% C.I. from 2D smoothing with P-splines, logarithmic scale. Selected ages over years. Denmark, females, 1950-2006.

What can we do with limited data?

Death rates over ages and years, logarithmic scale. Ages from 10 to 100. Sweden, females, 1930-2006. Source: HMD.
What can we do with limited data?

Actual and fitted death rates from a 2D smoothing with $P$-splines over ages for selected years, logarithmic scale. Estimation on 5% of actual data (randomly chosen). Sweden, females, age 10-100, years from 1930 to 2006.

Fitted death rates from a 2D smoothing with $P$-splines over ages for selected years, logarithmic scale. Estimation on 5% and 100% of actual data (randomly chosen). Sweden, females, age 10-100, years from 1930 to 2006.

Two-dimensional smoothing and plot:

```r
> ages <- 10:95
> years <- 1981:2007
> deaths2D <- read.table("deathsICE.txt")
> exposures2D <- read.table("exposuresICE.txt")
> fit2D <- Mort2Dsmooth(x=ages, y=years, Z=deaths2D, offset=log(exposures2D))
> fit2D
```

Bayesian Information Criterion (BIC): 2594.015

```r
> plot(fit2D)
```

MortalitySmooth: an R package for smoothing mortality

ISNPS, July 2015 Camarda C.G. (INED) Smoothing on grids with applications to mortality data 20
Outcome from Mort2Dsmooth

Actual and fitted death rates from 2D smoothing with P-splines. Ages from 10 to 95. Iceland, males, 1981-2007

Further features

- Support functions are available:
  - summary: gives a detailed summary of the outcome
  - residuals: computes diverse types of residuals (deviance, Pearson, Anscombe, working)
  - predict: predictions and standard errors, useful for forecasting
- Overdispersion can be accounted (quasi-likelihood estimation)
- Specifications for B-splines and penalty term can be modified
- Alternative criteria for the optimal smoothing are available (AIC, subjective, ED)
- Weights can be included in the fitting procedure
- Control settings can be adjusted
- Possible isotropic smoothing in the 2D setting

MortalitySmooth: optimizing smoothing parameters

- An heuristic as alternative to a full grid-search

MortalitySmooth: optimizing smoothing parameters

- An heuristic as alternative to a full grid-search
MortalitySmooth: optimizing smoothing parameters

- An heuristic as alternative to a full grid-search
MortalitySmooth: optimizing smoothing parameters

- An heuristic as alternative to a full grid-search

\[
\begin{array}{c}
\text{log10}(\lambda_a) \\
\text{log10}(\lambda_y)
\end{array}
\]

- See Dae-Jin Lee and Coté Rodriguez for alternative (and more informative) options

Concluding remarks

- We present an elegant and efficient statistical methodology for smoothing data on grids

Outline

- P-splines and GLAM
- Specific mortality applications
- MortalitySmooth: an R package for smoothing mortality

Conclusions

- We present an elegant and efficient statistical methodology for smoothing data on grids
- Array structures are the natural setting for these problems
Concluding remarks

- We present an elegant and efficient statistical methodology for smoothing data on grids
- Array structures are the natural setting for these problems
- The GLAM: Keep everything in array form, data and coefficients
- The key idea in GLAM is that of marginal processing:
  - The model matrix $B_y \otimes B_z$ is not stored
  - GLAM works with $B_y$ and $B_z$ separately
- Remember: GLAM is more than an algorithm - it is a structure for modelling

For mortality data:
- No strong model structure is assumed
- 2D approach suitable for small populations
- Forecasting and interpolation is possible
- Availability of routines for a friendly usage
Thanks for your attention.

Comments and questions?

carlo-giovanni.camarda@ined.fr

Institut National d’Etudes Démographiques, Paris, France

sites.google.com/site/carlogiovannicamarda