Computer–assisted Generalized Partial Linear Models

Marlene Müller

Humboldt University Berlin (Germany)

Computer–assisted Generalized Partial Linear Models

- Estimation
- Testing
- Simulations
- Application: Migration
- Computational Issues

Migration East→West Data:

GSOEP (spring 1991) on migration (intention), $n = 3235$

- dependent variable

$$Y = \begin{cases} 1 & \text{intention to migrate} \\ 0 & \text{otherwise} \end{cases}$$

- vector of 6 explanatory variables

		Yes	No	(in %)	
Y	migration intention	38.5	61.5		
X_1	family/friends in west	85.6	14.4		
X_2	unemployed	19.7	80.3		
X_3	city size 10-100,000	29.3	70.7		
X_4	female	51.1	48.9		
		Min	Max	Mean	S.D.
X_5	age	18	65	39.8	12.6
T	household income	200	4000	2194.3	752.4

How do the explanatory variables x (linear), t (nonlinear) influence

$$E(Y|\boldsymbol{x},\boldsymbol{t}) = P(Y=1|\boldsymbol{x},\boldsymbol{t})?$$

Analytic framework: latent-variable model

- binary response model

$$Y = \begin{cases} 1 & \text{if } Y^* = v(\boldsymbol{x},\boldsymbol{t}) - u > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Y^* = latent variable, net benefit from migrating
- $v(\bullet)$ = index function that relates $\boldsymbol{x}$, $\boldsymbol{t}$ to Y^*, e.g.

$$v(x,t) = \boldsymbol{\beta}^T\boldsymbol{x} + \boldsymbol{\gamma}^T + \gamma_0\boldsymbol{t}$$

- u = unobserved error term.

Suppose: $G = G_{u|x,t}$ known (e.g. logistic) distribution function

Generalized Linear Model (GLM)

* $P(Y=1|x,t) = G(\alpha + \boldsymbol{\beta}^T x + \boldsymbol{\gamma}^T t)$.

Generalized Additive Model (GAM)

* $P(Y=1|x,t) = G\left(\alpha + \boldsymbol{\beta}^T x + \sum_{j=1}^{q} m_j(t_j)\right)$.

Partial Linear Model

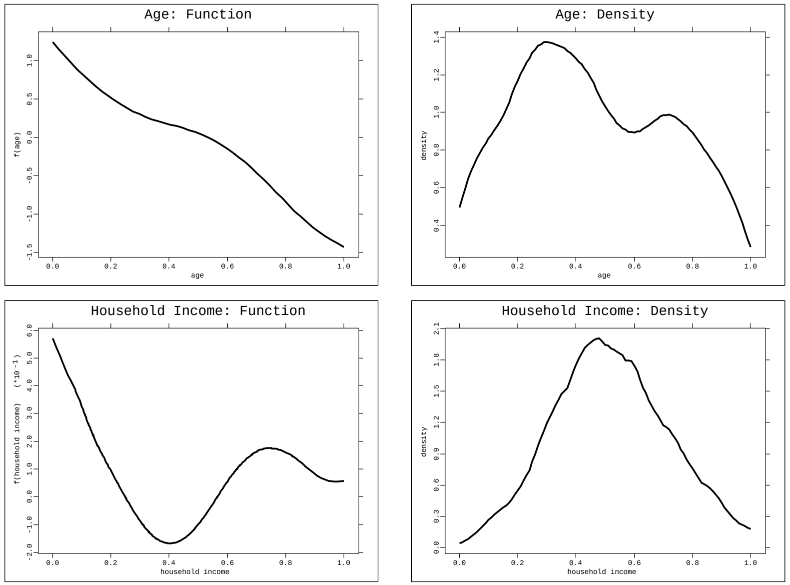
* $P(Y=1|x,t) = G\left(\boldsymbol{\beta}^T x + m(t)\right)$.

GAM fit for Migration data

Logit and GAM coefficients:

	logit (s.d.)	Coeff.
const.	-0.305 (0.177)	-0.534
family/friends in west	0.560 (0.115)	0.722
unemployed	0.221 (0.096)	0.315
city size 10-100,000	0.311 (0.083)	0.418
female	-0.240 (0.076)	-0.159
age	-2.208 (0.152)	—
household income	0.540 (0.197)	—

Estimates for continuous variables (GAM):



Semiparametric Quasi–Likelihood estimation

Quasi–likelihood (log–Likelihood!)

$$Q(\mu; y) = \int_{\mu}^{y} \frac{(s-y)}{V(s)}\, ds$$

Linear Model

$$\begin{aligned} E(Y|\boldsymbol{x}, \boldsymbol{t}) &= \mu = G\{\boldsymbol{x}^T \boldsymbol{\beta} + \boldsymbol{t}^T \boldsymbol{\gamma}\}, \\ Var(Y|\boldsymbol{x}, \boldsymbol{t}) &= \sigma^2 V(\mu). \end{aligned}$$

Partial linear model

$$\begin{aligned} E(Y|\boldsymbol{x}, \boldsymbol{t}) &= \mu = G\{\boldsymbol{x}^T \boldsymbol{\beta} + m(\boldsymbol{t})\}, \\ Var(Y|\boldsymbol{x}, \boldsymbol{t}) &= \sigma^2 V(\mu). \end{aligned}$$

Estimation in a Partial Linear Model

- $\hat{\boldsymbol{\beta}}$ can be found for known m,
- $\hat{m}$ can be found for known $\boldsymbol{\beta}$.

Profile likelihood (P)

maximize usual likelihood

$$0 = \sum_{i=1}^{n} \ell_i'\{x_i^T\boldsymbol{\beta} + m_{\boldsymbol{\beta}}(t_i)\}\left\{x_i + \frac{\partial}{\partial\beta} m_{\boldsymbol{\beta}}(t_i)\right\}$$

maximize smoothed quasi–likelihood

$$0 = \sum_{i=1}^{n} \ell_i'\{x_i^T\boldsymbol{\beta} + m_{\boldsymbol{\beta}}(t_j)\} K_h(t_i - t_j)$$

References:
Severini & Staniswalis (1994), Severini & Wong (1992),
Hastie & Tibshirani (1990), Speckman (1988)

Initialization

- Start with $\widetilde{\boldsymbol{\beta}}$, $\widetilde{m}_j$ from a parametric (GLM) fit. Higher order polynomial terms in $\boldsymbol{t}$ may be included to allow for a nonlinear function $\widetilde{m}_j$.
- Alternatively, it is possible to start with $\boldsymbol{\beta} = 0$ and $m_j = G^{-1}(y_j)$ as for the index in GLM (with the adjustment $m_j = G^{-1}\{(y_j + 0.5)/2\}$ for binary responses).

Algorithm (P)

- *updating step for* β

$$
\begin{aligned}
\boldsymbol{\beta}^{new} &= \boldsymbol{\beta} - \mathcal{B}^{-1} \sum_{i=1}^{n} \ell_i'(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_i)\,\widetilde{\boldsymbol{x}}_i \\
\mathcal{B} &= \sum_{i=1}^{n} \ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_i)\,\widetilde{\boldsymbol{x}}_i\widetilde{\boldsymbol{x}}_i^T \\
\widetilde{\boldsymbol{x}}_j &= \boldsymbol{x}_j - \frac{\sum\limits_{i=1}^{n} \ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\,\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)\,\boldsymbol{x}_i}{\sum\limits_{i=1}^{n} \ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\,\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)}.
\end{aligned}
$$

- *updating step for* m_j

$$
m_j^{new} = m_j - \frac{\sum\limits_{i=1}^{n} \ell_i'(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\,\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)}{\sum\limits_{i=1}^{n} \ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\,\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)}.
$$

Define $\mathcal{S}^P$ the smoother matrix

$$
\mathcal{S}_{ij}^P = \frac{\ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)}{\sum\limits_{i=1}^{n} \ell_i''(\boldsymbol{x}_i^T\boldsymbol{\beta} + m_j)\mathcal{K}_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)}
$$

Algorithm (P)

- *updating step for* β

$$
\boldsymbol{\beta}^{new} = (\widetilde{\mathcal{X}}^T\mathcal{W}\widetilde{\mathcal{X}})^{-1}\widetilde{\mathcal{X}}^T\mathcal{W}\widetilde{\boldsymbol{z}}
$$

with

$$
\begin{aligned}
\widetilde{\mathcal{X}} &= (\mathcal{I} - \mathcal{S}^P)\mathcal{X}, \\
\widetilde{\boldsymbol{z}} &= \widetilde{\mathcal{X}}\boldsymbol{\beta} - \mathcal{W}^{-1}\boldsymbol{v}.
\end{aligned}
$$

$\mathcal{X}$ design, $\mathcal{I}$ identity, $\boldsymbol{v} = \ell_i'$, $\mathcal{W} = \ell_i''$

Backfitting

- ordinary partial linear model (identity G)

$$E(Y|\boldsymbol{x},\boldsymbol{t}) = \boldsymbol{x}^T\boldsymbol{\beta} + m(\boldsymbol{t})$$

$\Rightarrow$ $\mathcal{P} = \mathcal{X}(\mathcal{X}^T\mathcal{X})^{-1}\mathcal{X}^T$ and $\mathcal{S}$ a smoother matrix, then backfitting means to solve

$$\begin{aligned} \mathcal{X}\boldsymbol{\beta} &= \mathcal{P}(\boldsymbol{y}-\boldsymbol{m}) \\ \boldsymbol{m} &= \mathcal{S}(\boldsymbol{y}-\mathcal{X}\boldsymbol{\beta}). \end{aligned}$$

- generalized partial linear model

$$E(Y|\boldsymbol{x},\boldsymbol{t}) = G\{\boldsymbol{x}^T\boldsymbol{\beta} + m(\boldsymbol{t})\}$$

$\Rightarrow$ backfitting for adjusted dependent variable

Reference: Hastie & Tibshirani (1990)

Algorithm (B)

- *updating step for $\boldsymbol{\beta}$*

$$\boldsymbol{\beta}^{new} = (\mathcal{X}^T\mathcal{W}\widetilde{\mathcal{X}})^{-1}\mathcal{X}^T\mathcal{W}\widetilde{\boldsymbol{z}},$$

- *updating step for $\boldsymbol{m}$*

$$\boldsymbol{m}^{new} = \mathcal{S}(\boldsymbol{z}-\mathcal{X}\boldsymbol{\beta}),$$

using the notations

$$\begin{aligned} \widetilde{\mathcal{X}} &= (\mathcal{I}-\mathcal{S})\mathcal{X}, \\ \widetilde{\boldsymbol{z}} &= (\mathcal{I}-\mathcal{S})\boldsymbol{z} = \widetilde{\mathcal{X}}\boldsymbol{\beta} - \mathcal{W}^{-1}\boldsymbol{v}. \end{aligned}$$

$\mathcal{X}$ design, $\mathcal{I}$ identity, $\boldsymbol{v} = \ell_i'$, $\mathcal{W} = \ell_i''$

Note that the update of the index $\mathcal{X}\boldsymbol{\beta} + \boldsymbol{m}$ can be expressed by a linear estimation matrix $\mathcal{R}^B$:

$$\mathcal{X}\boldsymbol{\beta}^{new} + \boldsymbol{m}^{new} = \mathcal{R}^B \boldsymbol{z}$$

with

$$\mathcal{R}^B = \widetilde{\mathcal{X}}\{\mathcal{X}^T\mathcal{W}\widetilde{\mathcal{X}}\}^{-1}\mathcal{X}^T\mathcal{W}(\mathcal{I} - \mathcal{S}) + \mathcal{S}.$$

Generalized Speckman Estimator

- ordinary partial linear model (identity G)

$$E(Y|\boldsymbol{x},\boldsymbol{t}) = \boldsymbol{x}^T\boldsymbol{\beta} + m(\boldsymbol{t})$$

$\Rightarrow$ update for m and β

$$\begin{aligned} \boldsymbol{m}^{new} &= \mathcal{S}(\boldsymbol{y} - \mathcal{X}\boldsymbol{\beta}) \\ \boldsymbol{\beta}^{new} &= (\widetilde{\mathcal{X}}^T\mathcal{W}\widetilde{\mathcal{X}})^{-1}\widetilde{\mathcal{X}}^T\mathcal{W}\widetilde{\boldsymbol{y}} \end{aligned}$$

- generalized partial linear model

$$E(Y|\boldsymbol{x},\boldsymbol{t}) = G\{\boldsymbol{x}^T\boldsymbol{\beta} + m(\boldsymbol{t})\}$$

$\Rightarrow$ above for adjusted dependent variable

Reference: Speckman (1988), Hastie & Tibshirani (1990)

Algorithm (S)

- *updating step for* $\boldsymbol{\beta}$

$$\boldsymbol{\beta}^{new} = (\widetilde{\mathcal{X}}^T \mathcal{W} \widetilde{\mathcal{X}})^{-1} \mathcal{X}^T \mathcal{W} \widetilde{\boldsymbol{z}},$$

- *updating step for* $\boldsymbol{m}$

$$\boldsymbol{m}^{new} = \mathcal{S}(\boldsymbol{z} - \mathcal{X}\boldsymbol{\beta})$$

using the notations

$$\begin{aligned} \widetilde{\mathcal{X}} &= (\mathcal{I} - \mathcal{S})\mathcal{X}, \\ \widetilde{\boldsymbol{z}} &= (\mathcal{I} - \mathcal{S})\boldsymbol{z} = \widetilde{\mathcal{X}}\boldsymbol{\beta} - \mathcal{W}^{-1}\boldsymbol{v}. \end{aligned}$$

$\mathcal{X}$ design, $\mathcal{I}$ identity, $\boldsymbol{v} = \ell_i'$, $\mathcal{W} = \ell_i''$

This method (S) shares the property of being linear on the variable $\boldsymbol{z}$:

$$\mathcal{X}\boldsymbol{\beta}^{new} + \boldsymbol{m}^{new} = \mathcal{R}^S \boldsymbol{z}$$

with

$$\mathcal{R}^S = \widetilde{\mathcal{X}}\{\widetilde{\mathcal{X}}^T \mathcal{W} \widetilde{\mathcal{X}}\}^{-1} \widetilde{\mathcal{X}}^T \mathcal{W}(\mathcal{I} - \mathcal{S}) + \mathcal{S}.$$

Note that here in contrast to (B) always $\widetilde{\mathcal{X}}$ is used.

Likelihood ratio test and approximate degrees of freedom

LR test statistic

$$R = 2\sum_{i=1}^{n} L(\widehat{\mu}_i, y_i) - L(\widetilde{\mu}_i, y_i)$$

semiparametric: $\widehat{\mu}_i = G\{\boldsymbol{x}_i^T\widehat{\boldsymbol{\beta}} + \widehat{m}(\boldsymbol{t}_i)\}$

parametric: $\widetilde{\mu}_i = G\{\boldsymbol{x}_i^T\widetilde{\boldsymbol{\beta}} + \boldsymbol{t}^T\widetilde{\boldsymbol{\gamma}} + \widetilde{\gamma}_0\}$

Deviance

$$D(\boldsymbol{y}, \widehat{\boldsymbol{\mu}}) = 2\sum_{i=1}^{n} L(\mu_i^{max}, y_i) - L(\widehat{\mu}_i, y_i)$$

$\Rightarrow$

$$R = D(\boldsymbol{y}, \widetilde{\boldsymbol{\mu}}) - D(\boldsymbol{y}, \widehat{\boldsymbol{\mu}})$$

If at convergence of iterative estimation:

$$\widehat{\boldsymbol{\eta}} = \mathcal{R}\boldsymbol{z} = \mathcal{R}(\widehat{\boldsymbol{\eta}} - \mathcal{W}^{-1}\boldsymbol{v}.)$$

then

$$D(\boldsymbol{y}, \widehat{\boldsymbol{\mu}}) \approx (\boldsymbol{z} - \widehat{\boldsymbol{\eta}})^T\mathcal{W}^{-1}(\boldsymbol{z} - \widehat{\boldsymbol{\eta}})$$

approximate degrees of freedom

$$df^{err}(\widehat{\boldsymbol{\mu}}) = n - \text{tr}\left(2\mathcal{R} - \mathcal{R}^T\mathcal{W}\mathcal{R}\mathcal{W}^{-1}\right)$$

or

$$df^{err}(\widehat{\boldsymbol{\mu}}) = n - \text{tr}\left(\mathcal{R}\right)$$

Reference: Hastie & Tibshirani (1990)

For backfitting (B) and algorithm (S)

$$\widehat{\boldsymbol{\eta}} = \mathcal{R}\boldsymbol{z}$$

For profile likelihood (P) approximately

$$\mathcal{R}^P = \widetilde{\mathcal{X}}\{\widetilde{\mathcal{X}}^T\mathcal{W}\widetilde{\mathcal{X}}\}^{-1}\widetilde{\mathcal{X}}^T\mathcal{W}(\mathcal{I}-\mathcal{S}^P)+\mathcal{S}$$

where $\widetilde{\mathcal{X}}$ denotes $(\mathcal{I}-\mathcal{S}^P)\mathcal{X}$.

Modified likelihood ratio test

bias–corrected parametric estimate

$$\overline{m}(\boldsymbol{t}_j)$$

from

$$\{G(\boldsymbol{x}_i^T\widetilde{\boldsymbol{\beta}}+\boldsymbol{t}_i^T\widetilde{\boldsymbol{\gamma}}+\widetilde{\gamma}_0),\boldsymbol{x}_i,\boldsymbol{t}_i\},\quad i=1,\ldots,n$$

modified LR statistic

$$R^\mu = 2\sum_{i=1}^n L(\widehat{\mu}_i,\widehat{\mu}_i) - L(\overline{\mu}_i,\widehat{\mu}_i)$$

References: Härdle, Mammen & Müller (1996) asymptotically equivalent

$$\widetilde{R}^{\mu} = \sum_{i=1}^{n} w_i \left\{ \boldsymbol{x}_i^T (\widehat{\boldsymbol{\beta}} - \widetilde{\boldsymbol{\beta}}) + \widehat{m}(\boldsymbol{t}_i) - \overline{m}(\boldsymbol{t}_i) \right\}^2$$

with

$$w_i = \frac{[G'\{\boldsymbol{x}_i^T \widehat{\boldsymbol{\beta}} + \widehat{m}(\boldsymbol{t}_i)\}]^2}{V[G\{\boldsymbol{x}_i^T \widehat{\boldsymbol{\beta}} + \widehat{m}(\boldsymbol{t}_i)\}]}.$$

Asymptotic Normality

Under linearity hypothesis

(i) $R^{\mu} = \widetilde{R}^{\mu} + o_p(v_n)$,

(ii) $v_n^{-1}(R^{\mu} - e_n) \xrightarrow[N]{D} (0,1)$,

where

$$\begin{aligned} e_n &= \left\{ \lambda_T \cdot \int K(u)^2 \, du \right\} \{h_1 \dots h_q\}^{-1}, \\ v_n^2 &= 2 \left[\lambda_T \int \{K \star K(u)\}^2 \, du \right] \{h_1 \dots h_q\}^{-1}, \end{aligned}$$

Bootstrap works

It holds

$$d_K(R_j^*, R^{\mu}) \xrightarrow[0]{P}$$

where d_K denotes the Kolmogorov distance.

1. Generate samples $y_1^*, \dots, y_n^*$ with

$$\begin{aligned} E^*(y_i^*) &= G(\boldsymbol{x}_i^T \widetilde{\boldsymbol{\beta}} + \boldsymbol{t}_i^T \widetilde{\boldsymbol{\gamma}} + \gamma_0) \\ \text{var}^*(y_i^*) &= \widehat{\sigma}^2 \, V\{G(\boldsymbol{x}_i^T \widetilde{\boldsymbol{\beta}} + \boldsymbol{t}_i^T \widetilde{\boldsymbol{\gamma}} + \gamma_0)\}. \end{aligned}$$

2. Calculate estimates based on the bootstrap samples and finally the test statistics R^*. The quantiles of the distribution of R are estimated by the quantiles of the conditional distributions of R^*.

Simulations

A logit model was used to simulate data:

$$E(Y|\boldsymbol{X},T) = P(Y=1|\boldsymbol{X},T) = F\{\boldsymbol{X}^T\boldsymbol{\beta} + m(T)\}$$

- $\boldsymbol{\beta} = (\beta_1,\beta_2)^T = (2,-1)^T$
- under the hypothesis: $m(t) = t$,

  under the alternative: $m(t) = \cos(\pi t)$,

T and X_1 are independent and uniform on $[-1,1]$,
X_2 is discretization of $\cos\{\pi(\rho T + (1-\rho)U\}$

independent design: $\rho = 0$,

dependent design: $\rho = 0.7$.

	$\widehat{\mu}$	$\widehat{m}$	$\widehat{\boldsymbol{\beta}}$	D	df^{err}
GPLM (P)	1.323	0.255	0.187	101.726	94.47
GPLM (S)	1.271	0.243	0.164	101.947	94.52
GPLM (B)	1.229	0.242	0.143	102.122	94.53
GPLM (MB)	1.229	0.242	0.143	102.124	94.50
	$n=100,\ h=0.6$				
GPLM (P)	0.582	0.115	0.064	261.143	243.83
GPLM (S)	0.577	0.117	0.060	261.468	243.88
GPLM (B)	0.559	0.121	0.056	261.698	243.90
GPLM (MB)	0.559	0.121	0.056	261.699	243.87
	$n=250,\ h=0.5$				
GPLM (P)	0.316	0.064	0.035	523.840	492.89
GPLM (S)	0.318	0.065	0.033	524.233	492.93
GPLM (B)	0.316	0.068	0.032	524.473	493.95
GPLM (MB)	0.316	0.068	0.032	524.473	493.93
	$n=500,\ h=0.4$				

Table 1: Mean ASE's for μ ($\times 100$), m and $\boldsymbol{\beta}$, mean deviances and mean degrees of freedom. Model under alternative, independent design, 500 Monte–Carlo's.

	$\widehat{\mu}$	$\widehat{m}$	$\widehat{\boldsymbol{\beta}}$	D	df^{err}
GPLM (P)	1.416	0.280	0.277	115.986	94.44
GPLM (S)	1.351	0.273	0.273	116.190	94.55
GPLM (B)	1.403	0.344	0.345	116.823	94.66
GPLM (MB)	1.403	0.344	0.345	116.823	94.48
	$n = 100,\ h = 0.6$				
GPLM (P)	0.654	0.128	0.096	295.133	243.77
GPLM (S)	0.641	0.131	0.091	295.442	243.88
GPLM (B)	0.697	0.193	0.168	296.618	243.99
GPLM (MB)	0.697	0.193	0.167	296.607	243.84
	$n = 250,\ h = 0.5$				
GPLM (P)	0.342	0.068	0.050	594.150	492.82
GPLM (S)	0.340	0.070	0.049	594.497	492.92
GPLM (B)	0.382	0.106	0.094	595.873	493.02
GPLM (MB)	0.382	0.106	0.093	595.864	492.89
	$n = 500,\ h = 0.4$				

Table 2: Mean ASE's for μ $(\times 1000)$, m and $\boldsymbol{\beta}$, mean deviances and mean degrees of freedom. Model under alternative, dependent design, 500 Monte–Carlo's.

α	0.01	0.05	0.10	0.20
R parametric	0.016	0.056	0.108	0.224
R (P)	0.016	0.084	0.180	0.348
R (S)	0.016	0.080	0.168	0.348
R (B)	0.016	0.076	0.184	0.376
R (MB)	0.008	0.072	0.156	0.300
R^{μ} (P) bootstrap	0.032	0.056	0.108	0.212
	$n = 100,\ h = 0.6$			
R parametric	0.020	0.052	0.104	0.188
R (P)	0.020	0.056	0.144	0.316
R (S)	0.016	0.060	0.144	0.312
R (B)	0.016	0.060	0.144	0.324
R (MB)	0.012	0.052	0.140	0.292
R^{μ} (P) bootstrap	0.028	0.044	0.100	0.196
	$n = 250,\ h = 0.5$			

Table 3: Percentage of rejections. Model under hypothesis, dependent design, 250 Monte–Carlo's.

α	0.01	0.05	0.10	0.20
R parametric	0.836	0.944	0.972	0.988
R (P)	0.764	0.916	0.960	0.984
R (S)	0.768	0.920	0.964	0.992
R (B)	0.760	0.928	0.964	0.996
R (MB)	0.728	0.904	0.952	0.984
R^{μ} (P) bootstrap	0.832	0.936	0.952	0.976
	$n = 100,\ h = 0.6$			
R parametric	1.000	1.000	1.000	1.000
R (P)	0.996	1.000	1.000	1.000
R (S)	0.996	1.000	1.000	1.000
R (B)	0.996	1.000	1.000	1.000
R (MB)	0.996	1.000	1.000	1.000
R^{μ} (P) bootstrap	0.996	1.000	1.000	1.000
	$n = 250,\ h = 0.5$			

Table 4: Percentage of rejections. Model under alternative, dependent design, 250 Monte–Carlo's.

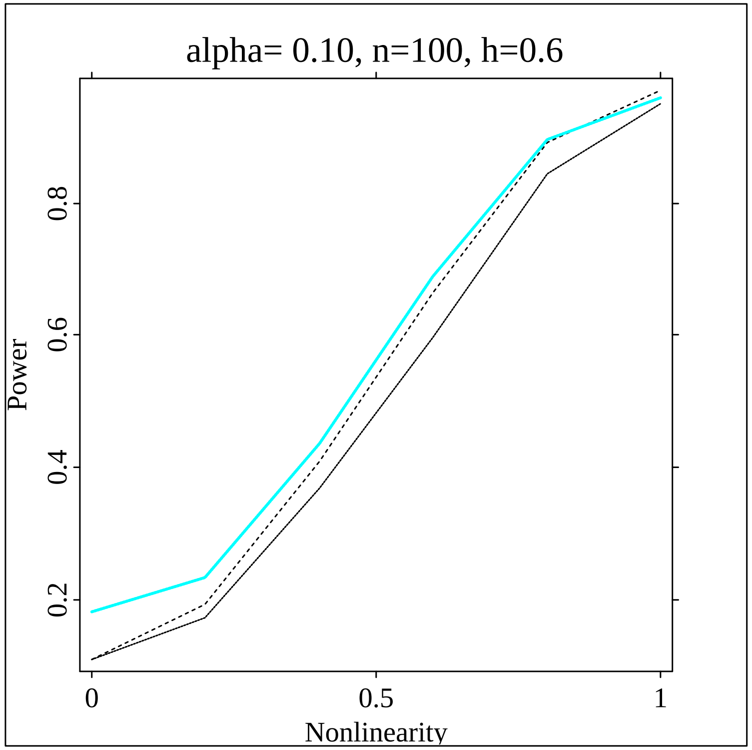


Figure 1: Power of likelihood ratio statistics R (P), R^{μ} (P) and R parametric (grey, black and dashed). $n = 100$, dependent design, 250 Monte–Carlo replications.

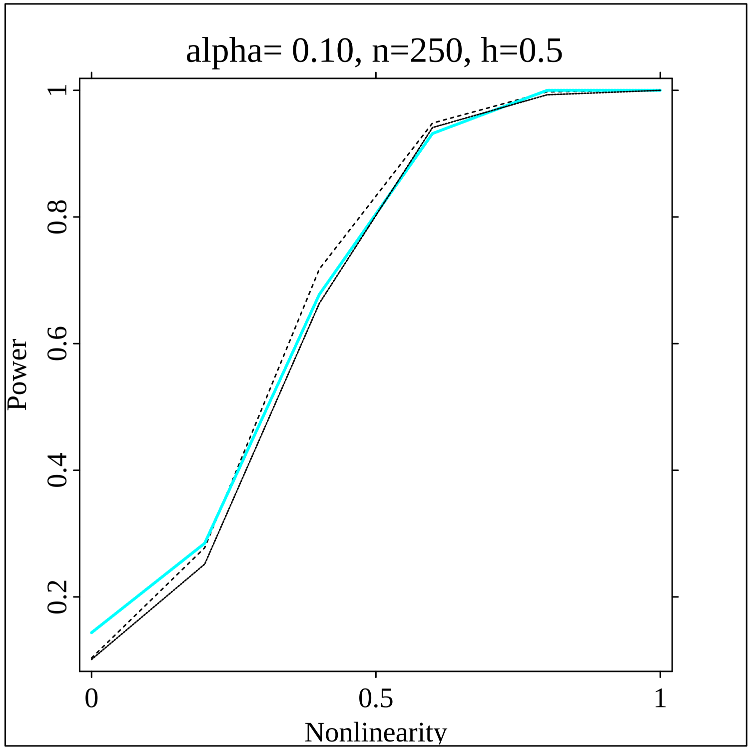


Figure 2: Power of likelihood ratio statistics R (P), R^{μ} (P) and R parametric (grey, black and dashed). $n = 250$, dependent design, 250 Monte–Carlo replications.

Example: Migration

		Yes	No	(in %)	
Y	**migration**	39.9	60.1		
X_1	**family/friends**	88.8	11.2		
X_2	**unemployed**	21.1	78.9		
X_3	**city size**	35.8	64.2		
X_4	**female**	50.2	49.8		
		Min	Max	Mean	S.D.
X_5	**age (years)**	18	65	39.93	12.89
T	**income (DM)**	400	4000	2262.22	769.82

Table 5: Descriptive statistics for migration data. Sample from Mecklenburg–Vorpommern, $n = 402$.

Estimated Coefficients

	Logit (t value)	(P)	(S)	(B)
constant	-0.358 (-0.68)	–	–	–
X_1	0.589 (1.54)	0.600	0.599	0.395
X_2	0.780 (2.81)	0.800	0.794	0.765
X_3	0.822 (3.39)	0.842	0.836	0.784
X_4	-0.388 (-1.68)	-0.402	-0.400	-0.438
X_5	-3.364 (-6.92)	-3.329	-3.313	-3.468
T	1.084 (1.90)	–	–	–
	Linear (GLM)	Part. Linear (GPLM)		

Table 6: Logit coefficients and coefficients in GPLM for migration data. Sample from Mecklenburg–Vorpommern. $n = 402$, $h = 0.3$ for the GPLM.

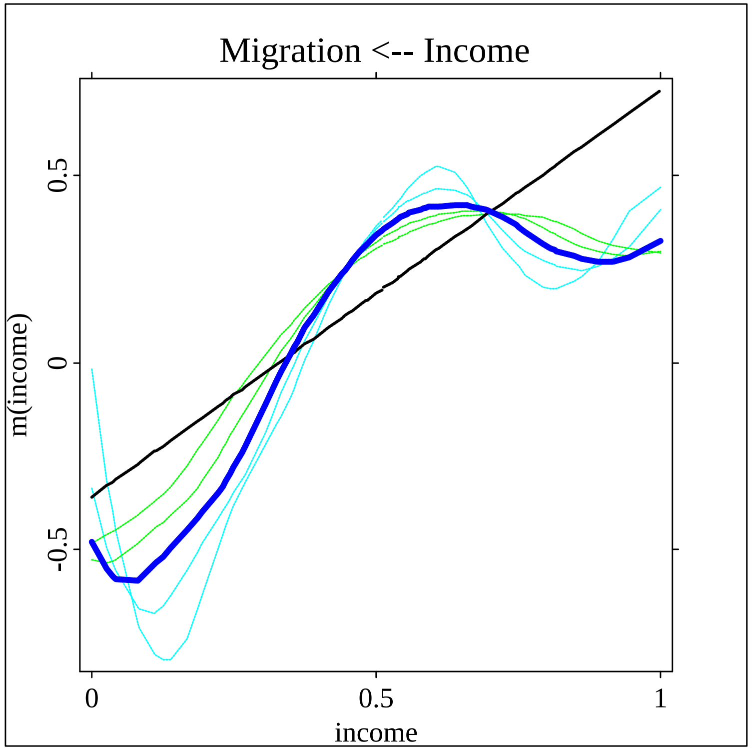


Figure 3: GPLM logit fit for migration data. Profile likelihood estimator (P) for m, with $h = 0.3$ (thick curve), $h = 0.2$, $h = 0.25$, $h = 0.35$, $h = 0.4$ (thin curves) and parametric logit fit (medium line).

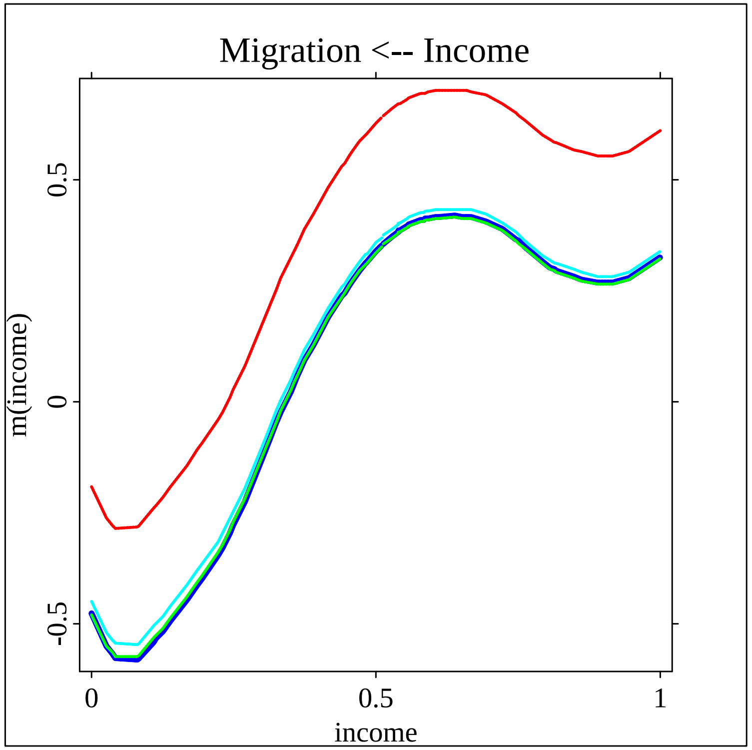


Figure 4: GPLM logit fit for migration data. Profile likelihood and simple profile likelihood (lower curves), backfitting (upper curve), $h = 0.3$.

Test results

h	0.20	0.25	0.30	0.35	0.40
R (P)	0.066	0.054	0.048	0.045	0.035
R (S)	0.068	0.055	0.047	0.044	0.033
R (B)	0.073	0.064	0.062	0.069	0.068
R (MB)	0.068	0.056	0.048	0.045	0.035
R^{μ} (P) bootstrap	0.065	0.054	0.042	0.042	0.045

Table 7: Observed significance levels for linearity test for migration data, $n = 402$. 500 bootstrap replications.

Computational Issues

updating step for $m_j = m_{\boldsymbol{\beta}}(\boldsymbol{t}_j)$

Algorithm (P):

$$\sum_{i=1}^{n} \delta_{ij} K_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)$$

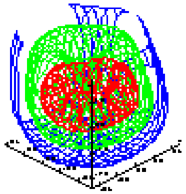
Algorithm (B), (S):

$$\sum_{i=1}^{n} \delta_{i} K_{\mathbf{H}}(\boldsymbol{t}_i - \boldsymbol{t}_j)$$

$O(n^2)$ operations

Conclusions

- Backfitting (B) is best under independence. (S) seems best otherwise.
- For large n: (P) $\approx$ (S)
- In testing parametric versus nonparametric (P) and (S) work well with approximate degrees of freedom. Bootstrapping R^M improves.
- (S) seems a good compromise between accuracy and computational efficiency in estimation and specification testing.



XploRe

http://www.xplore-stat.de

- **Download**
  local, local network, Java client & server
- **Use**
  Java netscape
- **Example collection**
  http://www.xplore-stat.de/∼marlene
  ⇒ XploRe macros for teaching