Flexible Regression and Smoothing
Model Selection

Bob Rigby    Mikis Stasinopoulos

Dipartimento di Scienze Cliniche a de Comunita, Universita degli studi di Milano, Italy, January 2018
Statistical models are built to:

- explore the data where no theory exists, *exploratory* models,
- explain or verify a theory, *explanatory* models,
- predict future values, *predictive* models

or any combination of the above situations.
GAMLSS components

Let $\mathcal{M} = \{D, G, T, \lambda\}$ represent the GAMLSS model

- $D$: distribution
- $G$: the link function for distributional parameters
- $T$: predictor terms for ($\eta$’s) i.e. $\eta = X\beta + \sum_j h_j(x_j)$
- $\lambda$: the hyperparameters
Problems:

- which distribution
- which the link function for distributional parameters
- which x-variables for $\mu$
- which x-variables for $\sigma$
- which x-variables for $\nu$
- which x-variables for $\tau$
- choosing the smoothing hyper parameters for terms in $\mu$, $\sigma$, $\nu$ and $\tau$
- selection between different (GAMLSS or not) models
Questions to answer

- Do we need the extra complexity of GAMLSS?
- How we compare the different models? This depends also on the purpose of the study
  - is it for prediction (forecasting)?
  - are we exploring relationship between variables?
- Do we choose one or average between models?
GAMLSS and GLM which to choose?

- is GAMLSS models an improvement on the standard GLM?
- Bohl et al., (2013) compared GLM and GAMLSS estimators and concluded that "the GLM gamma was the most consistent" in their simulations.
- All their simulations involved a single explanatory categorical variable with two levels and equal sample sizes.
GAMLSS statistical properties

For a parametric GAMLSS model ML estimation is used:

- if the model is correct
  - parameter estimators are (weakly) consistent, with correct asymptotic standard errors,
  - correct confidence interval coverage and test size,
  - parameter estimators are robust to outliers in specific cases

- if the model is not correct
  - the parameter estimators may not be consistent estimators of true population parameters.
GLM statistical properties

- if the mean model is correct
  - the estimators of the mean model parameters are always strongly consistent (property of the exponential family)
- if the mean model is correct but the variance and/or distribution model is wrong
  - mean model parameter estimators are (strongly) consistent but asymptotically inefficient
  - estimated SE are in general, asymptotically incorrect unless robust SE are used,
- GLM mean model parameter estimators are not robust to outliers,
Simulation studies

- Simulation 1:
  - $Y \sim GA(100, 0.5)$, for $x = 0$
  - $Y \sim GA(130, 1)$ for $x = 1$

- Simulation 2:
  - $Y \sim GIG(100, 5, -2)$ for $x = 0$
  - $Y \sim GIG(130, 0.5, 2)$ for $x = 1$
Simulation study 1

**Table:** Simulation 1 results: \( Y \sim GA(100, 0.5) \), for \( x = 0 \), \( Y \sim GA(130, 1) \) for \( x = 1 \).

<table>
<thead>
<tr>
<th>sample size</th>
<th>Method</th>
<th>MSE (^a)</th>
<th>% CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) [1000, 1000]</td>
<td>GLM</td>
<td>1.20</td>
<td>95.7</td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>1.20</td>
<td>95.9</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>95.8</td>
</tr>
<tr>
<td>(b) [500, 1500]</td>
<td>GLM</td>
<td>1.21</td>
<td>98.9</td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>1.21</td>
<td>94.8</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>94.8</td>
</tr>
<tr>
<td>(c) [1500, 500]</td>
<td>GLM</td>
<td>2.16</td>
<td>85.1</td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>2.16</td>
<td>95.2</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>95.5</td>
</tr>
</tbody>
</table>
## Simulation study 2

**Table:** Simulation 2: $Y \sim GIG(100, 5, -2)$ for $x = 0$, $Y \sim GIG(130, 0.5, 2)$ for $x = 1$.

<table>
<thead>
<tr>
<th>sample size</th>
<th>Method</th>
<th>MSE(^a)</th>
<th>% CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) [1000,1000]</td>
<td>GLM</td>
<td>3.77</td>
<td>92.0</td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>3.11</td>
<td>89.5</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>94.9</td>
</tr>
<tr>
<td>(b) [500, 1500]</td>
<td>GLM</td>
<td>6.98</td>
<td><strong>72.3</strong></td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>5.24</td>
<td>89.0</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>94.8</td>
</tr>
<tr>
<td>(c) [1500, 500]</td>
<td>GLM</td>
<td>2.72</td>
<td><strong>98.4</strong></td>
</tr>
<tr>
<td></td>
<td>GAMLSS</td>
<td>2.51</td>
<td>90.1</td>
</tr>
<tr>
<td></td>
<td>GAMLSS (profile CI)</td>
<td>-</td>
<td>93.1</td>
</tr>
</tbody>
</table>

\(^a\) Mean squared error multiplied by 1000

Bob Rigby, Mikis Stasinopoulos
Flexible Regression and Smoothing
2016
Fitted distributions: GLM (dashed red), GAMLSS (dotted blue), true (black)
Conclusions

- GAMLSS model generally outperformed the GLM gamma model in terms of mean square error and coverage (the % of confidence intervals including the true parameter) for the parameter of interest.
- With different sample sizes the GLM coverage was very poor.
- Robust GLM SE for mean model parameters should be used when the GLM incorrectly specifies the true variance model.
- Misspecification of the model can lead to seriously misleading standard errors, confidence intervals and tests for mean model parameters.
Nested models

- **parametric**: use the likelihood ratio test
- **Non-parametric**: use likelihood ratio test as a guide

The function `LR.test()` can be used for example:

`LR.test(m1, m2)`
Non-nested models

Non-nested: use the Generalised Akaike Information Criterion

- Use the functions `AIC(\ldots, k)` or `GAIC(\ldots, k)` where $k$ is the penalty.
- \( GAIC(\ldots, k=2) \) is the standard AIC.
- \( GAIC(\ldots, k=\log(\text{length}(y))) \) is the SBC or BIC.
Selection between models

Validation and Cross validation methods

K-fold Cross Validation

In data rich situations

- Training data set
- Validation data set
- Test data test

Use the "predictive" global deviance for selecting a model
Selection between models

Validation/Test Global Deviance

\[
\hat{\eta}_i = X_i \hat{\beta}_i + Z_i1 \hat{\gamma}_{i1} + \ldots + Z_i k_1 \hat{\gamma}_{iJ_i}
\]

and \( \hat{\theta}_i = g_i(\hat{\eta}) \) for \( i = 1, 2, 3, 4 \):

\[
GDEV = -2\hat{\ell}(y, |\lambda_0).
\]

\[
\tilde{\eta}_i = \tilde{X}_i \tilde{\beta}_i + \tilde{Z}_i1 \tilde{\gamma}_{i1} + \ldots + \tilde{Z}_i k_1 \tilde{\gamma}_{iJ_i}
\]

\( \tilde{\theta}_i = g_i(\tilde{\eta}) \) for \( i = 1, 2, 3, 4 \).

The validation (or test) global deviance is defined as:

\[
VDEV = TDEV = -2\tilde{\ell}(\tilde{y}, |\lambda_0).
\]
Selection between models

The different function for model selection

<table>
<thead>
<tr>
<th>Comp.</th>
<th>All data</th>
<th>K-fold CV</th>
<th>Val./Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D$</td>
<td>GAIC()</td>
<td>gamlssCV()</td>
<td>gamlssVGD()</td>
</tr>
<tr>
<td></td>
<td>wp()</td>
<td>CV()</td>
<td>VGD()</td>
</tr>
<tr>
<td></td>
<td></td>
<td>getTGV()</td>
<td>TGD()</td>
</tr>
<tr>
<td>$G$</td>
<td>deviance() *</td>
<td>gamlssCV()</td>
<td>as above</td>
</tr>
<tr>
<td>$T$</td>
<td>drop1(), add1(),</td>
<td>gamlssCV()</td>
<td>drop1TGD()</td>
</tr>
<tr>
<td></td>
<td>add1ALL(),</td>
<td>CV()</td>
<td>add1TGD()</td>
</tr>
<tr>
<td></td>
<td>drop1ALL(),</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>stepGAIC()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>stepGAICA11.A()</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>stepGAICA11.B()</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\Lambda$ global</td>
<td>findhyper()</td>
<td>optim() *</td>
<td>optim() *</td>
</tr>
</tbody>
</table>
Choosing the distribution

How to select a distribution?

<table>
<thead>
<tr>
<th></th>
<th>NO</th>
<th>PE</th>
<th>TF</th>
<th>SHASH</th>
<th>SEP3</th>
<th>SST</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>$\nu$</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>$\tau$</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

- Start with a simple appropriate distribution e.g. NO
- Check the residuals
- Increase the complexity until the residuals looking good
Choosing terms in the predictor

The US pollution data

**Data summary:** US pollution data

\[ y \quad : \quad \text{sulphur dioxide concentration in air in mgs. per c.m.} \]

\[ x_1 \quad : \quad \text{average annual temperature in degrees F} \]

\[ x_2 \quad : \quad \text{number of manufacturers employing} \quad > \quad 20 \quad \text{workers} \]

\[ x_3 \quad : \quad \text{population size in thousands} \]

\[ x_4 \quad : \quad \text{average annual wind speed in miles per hour} \]

\[ x_5 \quad : \quad \text{average annual rainfall in inches} \]

\[ x_6 \quad : \quad \text{average number of days rainfall per year} \]
Choosing terms in the predictor

\texttt{drop1()}

\begin{verbatim}
data(usair)
mod1 <- gamlss(y ~ ., data = usair, family = GA)
drop1(mod1)
drop1(mod1, parallel="snow", ncpus=4)
\end{verbatim}
Choosing terms in the predictor

\textbf{drop1()}

Single term deletions for mu

Model:
\[ y \sim x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \]

<table>
<thead>
<tr>
<th>Df</th>
<th>AIC</th>
<th>LRT</th>
<th>Pr(Chi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;none&gt;</td>
<td>319.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x1</td>
<td>1 327.58</td>
<td>10.4245</td>
<td>0.001244 **</td>
</tr>
<tr>
<td>x2</td>
<td>1 326.92</td>
<td>9.7557</td>
<td>0.001788 **</td>
</tr>
<tr>
<td>x3</td>
<td>1 321.39</td>
<td>4.2299</td>
<td>0.039717 *</td>
</tr>
<tr>
<td>x4</td>
<td>1 324.08</td>
<td>6.9247</td>
<td>0.008501 **</td>
</tr>
<tr>
<td>x5</td>
<td>1 320.57</td>
<td>3.4141</td>
<td>0.064642 .</td>
</tr>
<tr>
<td>x6</td>
<td>1 317.16</td>
<td>0.0017</td>
<td>0.966960</td>
</tr>
</tbody>
</table>

---

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1
Choosing terms in the predictor

add1()

add1(mod1, scope = ~(x1 + x2 + x3 + x4 + x5 + x6)^2)
add1(mod1, scope = ~(x1 + x2 + x3 + x4 + x5 + x6)^2, parallel="snow", ncpus=4)
Choosing terms in the predictor

add1()

Single term additions for
mu
Model:
y ~ x1 + x2 + x3 + x4 + x5 + x6

<table>
<thead>
<tr>
<th>Term</th>
<th>Df</th>
<th>AIC</th>
<th>LRT</th>
<th>Pr(Chi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;none&gt;</td>
<td></td>
<td>319.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x1:x2</td>
<td>1</td>
<td>320.09</td>
<td>1.0689</td>
<td>0.3012</td>
</tr>
<tr>
<td>x1:x3</td>
<td>1</td>
<td>319.40</td>
<td>1.7626</td>
<td>0.1843</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x4:x5</td>
<td>1</td>
<td>307.07</td>
<td>14.0870</td>
<td>0.000174***</td>
</tr>
<tr>
<td>x4:x6</td>
<td>1</td>
<td>320.33</td>
<td>0.8346</td>
<td>0.3609322</td>
</tr>
<tr>
<td>x5:x6</td>
<td>1</td>
<td>318.74</td>
<td>2.4188</td>
<td>0.1198894</td>
</tr>
</tbody>
</table>

---
Choosing terms in the predictor

\texttt{stepGAIC()}

\begin{verbatim}
mod2 <- stepGAIC(mod1, parallel="snow", ncpus=4)
mod21 <- stepGAIC(mod1, k=length(usair$y))
mod3 <- stepGAIC(mod1, scope=list(lower=~1,
                   upper=~(x1+x2+x3+x4+x5+x6)^2))
mod2$anova
mod4 <- stepGAIC(mod1, parameter="sigma",
                   scope=~x1+x2+x3+x4+x5+x6)
\end{verbatim}
Different Strategies

How to select explanatory variables?

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\sigma$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\nu$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\tau$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Strategy A
- Strategy B
- Other strategies?
- Boosting
Strategy A:

- It starts with a **forward stepwise** selection using GAIC.
- Each $x$ variables is set for selection first for $\mu$ then for $\sigma$, $\nu$ and $\tau$
- then it does a **backward** elimination for $\nu$, $\sigma$ and $\mu$.

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>$\sigma$</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\nu$</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\tau$</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Different Strategies Selecting Terms

\texttt{stepGAICAll.A()}

\begin{verbatim}
m1 <- gamlss(y~1, data=usair, famly=GA, trace=FALSE)
m2 <- stepGAICAll.A(m1, scope=list(lower=~1, upper=~x1+x2+x3+x4+x5+x6))
m3 <- stepGAICAll.A(m1, scope=list(lower=~1, upper=~pb(x1)+pb(x2)+pb(x3)+pb(x4)+pb(x5)+pb(x6)), k=log(41))
\end{verbatim}
Strategy B: forward stepwise selection using GAIC in which an $x$ variable is selected for all the parameters

**Table:** selecting explanatory variables

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>$\nu$</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>$\tau$</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
Different Strategies Selecting Terms

```r
stepGAICAll.B()

m4 <- stepGAICAll.B(m1, scope=list(lower=~1,
                        upper=~x1+x2+x3+x4+x5+x6),
                        k=log(41))
```
GAMLSS for high-dimensional data – a flexible approach based on boosting

Andreas Mayr¹, Nora Fenske², Benjamin Hofner¹, Thomas Kneib³, Matthias Schmid¹

¹ Institut für Medizininformatik, Biometrie und Epidemiologie, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
² Institut für Statistik, Ludwig-Maximilians-Universität München, Germany
³ Institut für Mathematik, Carl von Ossietzky Universität Oldenburg, Germany
```r
rand1 <- sample (10 , 610, replace=TRUE)
# detecting how many cores exist in the machine
nC <- detectCores()
# no parallel
g1 <- gamlssCV(y~pb(x,df=2), sigma.formula=~pb(x,df=1),
   data=abdom, family=NO, rand=rand1)
# using multicore

g2 <- gamlssCV(y~pb(x,df=2), sigma.formula=~pb(x,df=1),
   data=abdom, family=LO, rand=rand1,
   parallel = "multicore", ncpus = nC )
# using snow

g3 <- gamlssCV(y~pb(x,df=2), sigma.formula=~pb(x,df=1),
   data=abdom, family=TF, rand=rand1,
   parallel = "snow", ncpus = nC )
CV(g1,g2,g3)
```
Different Strategies Selecting Terms

gamlssVGD()

rand <- sample(2, 610, replace=TRUE, prob=c(0.6,0.4))
#---------------------------------------------------------------
# using the argument rand
#---------------------------------------------------------------
v1 <- gamlssVGD(y~pb(x,df=2), sigma.fo=~pb(x,df=1),
    data=abdom, family=NO, rand=rand)

v2 <- gamlssVGD(y~pb(x,df=2), sigma.fo=~pb(x,df=1),
    data=abdom, family=LO, rand=rand)

v3 <- gamlssVGD(y~pb(x,df=2), sigma.fo=~pb(x,df=1),
    data=abdom, family=TF, rand=rand)

VGD(v1,v2,v3)
# fit models first
g1 <- gamlss(y~pb(x,df=2),sigma.fo=~pb(x,df=1),
data=olddata, family=NO,)
g2 <- gamlss(y~pb(x,df=2),sigma.fo=~pb(x,df=1),
data=olddata, family=LO)
g3 <- gamlss(y~pb(x,df=2),sigma.fo=~pb(x,df=1),
data=olddata, family=TF)

# and then use
gg1 <- getTGD(g1, newdata=newdata)
gg2 <- getTGD(g2, newdata=newdata)
gg3 <- getTGD(g3, newdata=newdata)
TGD(gg1,gg2,gg3)
Different Strategies Selecting Terms

```r
# complete model
v1 <- gamlss(R~pb(Fl)+pb(A)+H+loc,
             sigma.fo=~pb(Fl)+pb(A)+H+loc,
             data=oldrent, family=GA, trace=FALSE)

# drop1TGDP
nC <- detectCores()
v2<- drop1TGD(v1, newdata=newrent, parallel="snow", ncpus=nC)
v4<- stepTGD(v0, scope=~pb(Fl)+pb(A)+H+loc,
             newdata=newrent,
             parallel="snow", ncpus=nC)
```
Methods of Choosing Smoothing Parameters

<table>
<thead>
<tr>
<th>Global</th>
<th>Method</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global</td>
<td>ML /REML (e.g. Laplace)</td>
<td>Rigby and Stasinopoulos (2005)</td>
</tr>
<tr>
<td>Global</td>
<td>GAIC (e.g. SBC)</td>
<td>Rigby and Stasinopoulos (2004,2006)</td>
</tr>
<tr>
<td>Global</td>
<td>Validation Global Deviance (VGD)</td>
<td>Stasinopoulos and Rigby (2007)</td>
</tr>
<tr>
<td>Local</td>
<td>ML</td>
<td>Rigby and Stasinopoulos (2013) i) alternate method ii) The Q-function</td>
</tr>
<tr>
<td>Local</td>
<td>GAIC</td>
<td>Rigby and Stasinopoulos (2013)</td>
</tr>
<tr>
<td>Local</td>
<td>Generalized Cross</td>
<td>Wood (2006)</td>
</tr>
</tbody>
</table>
The AIDS data

**Data summary:** The AIDS data cases in the U.K. from January 1983 to March 1994

- \( y \) : the number of quarterly aids cases in England and Wales
- \( x \) : time in quarters from January 1983
- \( qrt \) : a factor for the quarterly seasonal effect
Methods of Choosing Smoothing Parameters

find.hyper()

- **find.hyper() function**
- Needs a model
  
  ```r
  mod1 <- quote(gamlss(y ~ cs(x, df = p[1]) + qrt, family = NBI, data = aids, trace = FALSE))
  ```

  ```r
  op <- find.hyper(model = mod1, par = c(3), lower = c(1), steps = c(0.1), pen = 2.8)
  ```
Methods of Choosing Smoothing Parameters

```
find.hyper()

op
$par
[1] 7.652983
$value
[1] 398.3985
$counts
function gradient
 12 12
$convergence
[1] 0
$message
[1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"
```
Methods of Choosing Smoothing Parameters

```r
find.hyper()
```

![Profile Global Deviance](image1)

![Profile GAIC](image2)

![Profile GAIC](image3)

![Profile GAIC](image4)
find.hyper()
Local ML

```r
> data(aids)
> m1 <- gamlss(y~pb(x)+qrt, data=aids, family=NBI)
GAMLSS-RS iteration 1: Global Deviance = 373.1785
...
GAMLSS-RS iteration 5: Global Deviance = 366.9258
> edf(m1)
Effective df for mu model
   pb(x)
7.588861
```
Methods of Choosing Smoothing Parameters

Local GCV

```r
> m2 <- gamlss(y~pb(x, method="GCV")+qrt, data=aids, family=NBI)
GAMLSS-RS iteration 1: Global Deviance = 365.5964
...
GAMLSS-RS iteration 5: Global Deviance = 360.6098
> edf(m2)
Effective df for mu model
pb(x, method = "GCV")
  9.252291
```
Local GAIC

> m3 <- gamlss(y~pb(x, method="GAIC", k=2)+qrt, data=aids, family=NBI)
GAMLSS-RS iteration 1: Global Deviance = 395.9969
...
GAMLSS-RS iteration 7: Global Deviance = 358.7914
GAMLSS-RS iteration 8: Global Deviance = 358.791
> edf(m3)
Effective df for mu model
pb(x, method = "GAIC", k = 2)
  9.990218
Local methods

- Methods of Choosing Smoothing Parameters

Bob Rigby, Mikis Stasinopoulos
Flexible Regression and Smoothing 2016 46 / 48
Conclusions

- Selecting a GAMLSS model is not easy
- Selecting using GAIC seems to be a reasonable solution (but which $k$?)
- Selecting a distributions is not trivial
- Selecting terms in a model for large data sets can be slow
- Selecting smoothing parameters using local methods seems OK
END

for more information see

www.gamlss.com