Generalized additive models in R

Magne Aldrin, Norwegian Computing Center and the University of Oslo

Sharp workshop, Copenhagen, October 2012

Generalized Linear Models - GLM

- $y \sim$ Distributed with mean μ and perhaps an additional parameter
- $h(\mu) = \eta$
- $\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$
- η: linear predictor
- h: link fuction

Generalized Additive Models - GAM

- η is additive, but each term can be non-linear
- $\eta = \beta_0 + s_1(x_1) + s_2(x_2) + \cdots + s_p(x_p)$
- η: additive predictor
- $s_i(\cdot)$ is a smooth function, estimated from the data

Estimation of s

- Find s such that
  - $\star$ fit to data is as best as possible and
  - $\star$ s is as smooth as possible
- Can for instance be formulated as
  - $\star$ Minimize $-likelihood + \sum_i \int_x \lambda_i [s_i''(x)]^2 dx$
  - $\star$ where $s''(x) = d^2 s(x)/d^2 x$ is the second derivative of s
  - $\star$ and λ's are constants that control the degree of smoothing

Effective number of parameters

- Very high λ_i gives maximal smoothness
  - a straight line
  - described by 1 parameter (per explanatory variable)
- Smaller values of λ_i gives more flexible functions
  - effective number of parameters > 1

Choice of smoothness in the mgcv package

- Default is to estimate the smoothness of each s-function, controlled by λ_i or a corresponding sp[i] by generalized cross-validation, minimizing
  - 2 log likelihood $\cdot n/(n-p)^2$ where

  ⋆ n $=$ number of observations

  ⋆ p $=$ effective number of parameters
- This tends to give too unsmooth curves

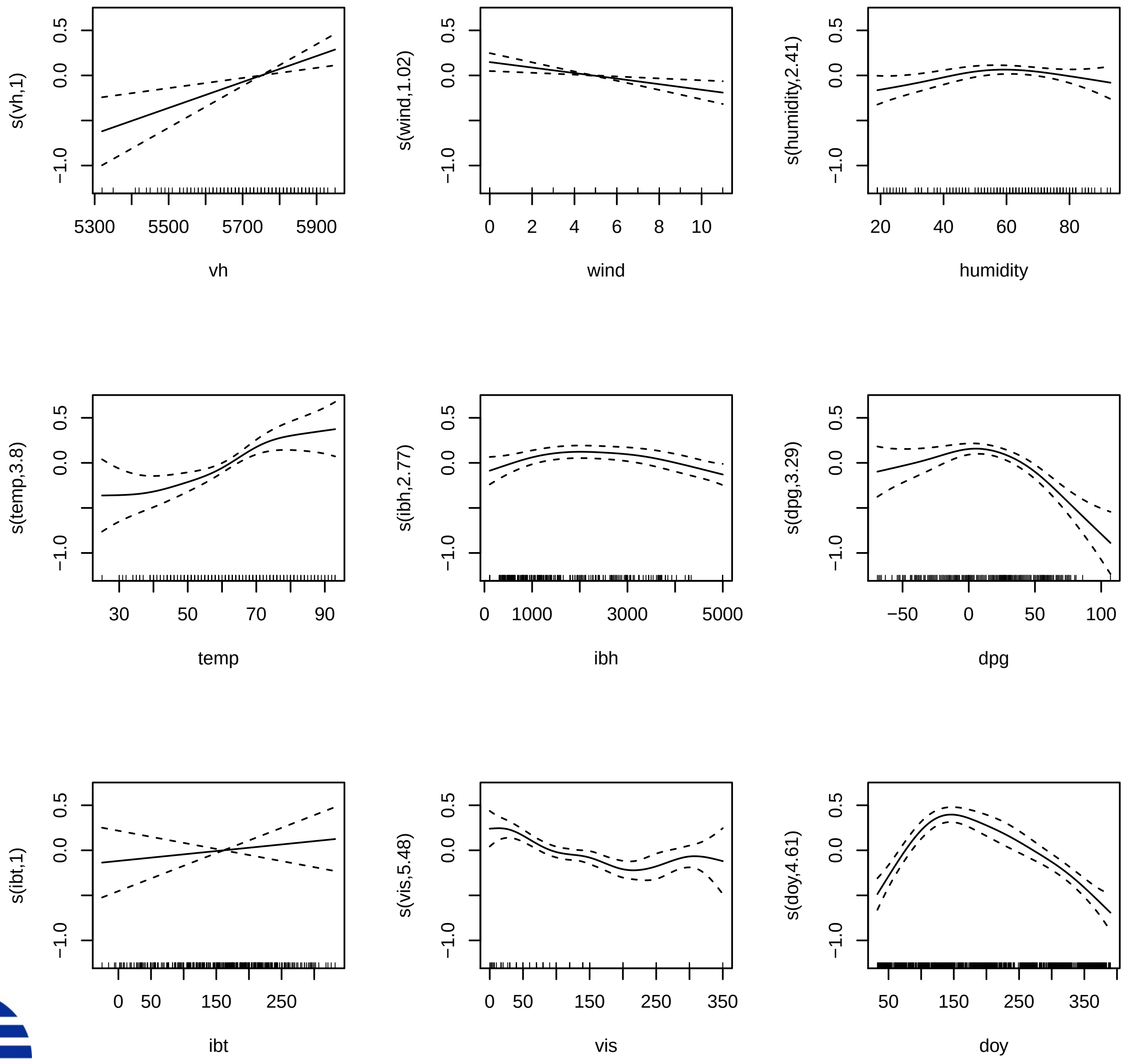
Ex: Ozone data $y = \log(NO_3) \sim$ Gaussian

```
> require(faraway)
> data(ozone)
>
> require(mgcv)
> gamobj<-gam(log(O3)~s(vh)+s(wind)+s(humidity)+s(temp)+s(ibh)+
s(dpg)+s(ibt)+s(vis)+s(doy),
family=gaussian(link=identity),data=ozone)
> summary(gamobj)

Family: gaussian
Link function: identity
Formula:
log(O3) ~ s(vh) + s(wind) + s(humidity) + s(temp) + s(ibh) +
    s(dpg) + s(ibt) + s(vis) + s(doy)
Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.21297    0.01717   128.9   <2e-16 ***
```

```
Approximate significance of smooth terms:
              edf Ref.df      F  p-value
s(vh)       1.000  1.000 10.780 0.001146 **
s(wind)     1.021  1.040  8.713 0.003036 **
s(humidity) 2.406  3.025  2.567 0.054130 .
s(temp)     3.801  4.740  4.161 0.001418 **
s(ibh)      2.774  3.393  5.341 0.000797 ***
s(dpg)      3.285  4.176 14.247 5.27e-11 ***
s(ibt)      1.000  1.000  0.495 0.482255
s(vis)      5.477  6.635  6.023 2.30e-06 ***
s(doy)      4.612  5.738 25.162  < 2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.826   Deviance explained =   84%
GCV score = 0.10569  Scale est. = 0.097247  n = 330
```

Plot results

```
> pdf("GAMozone.pdf")
> par(mfrow=c(3,3))
> plot(gamobj)
> dev.off()
null device
          1
```


s(vh,1)
vh
5300 5500 5700 5900
s(wind,1.02)
wind
0 2 4 6 8 10
s(humidity,2.41)
humidity
20 40 60 80
s(temp,3.8)
temp
30 50 70 90
s(ibh,2.77)
ibh
0 1000 3000 5000
s(dpg,3.29)
dpg
−50 0 50 100
s(ibt,1)
ibt
0 50 150 250
s(vis,5.48)
vis
0 50 150 250 350
s(doy,4.61)
doy
50 150 250 350
−1.0 0.0 0.5

Ex: Precipitation - binary logistic regression

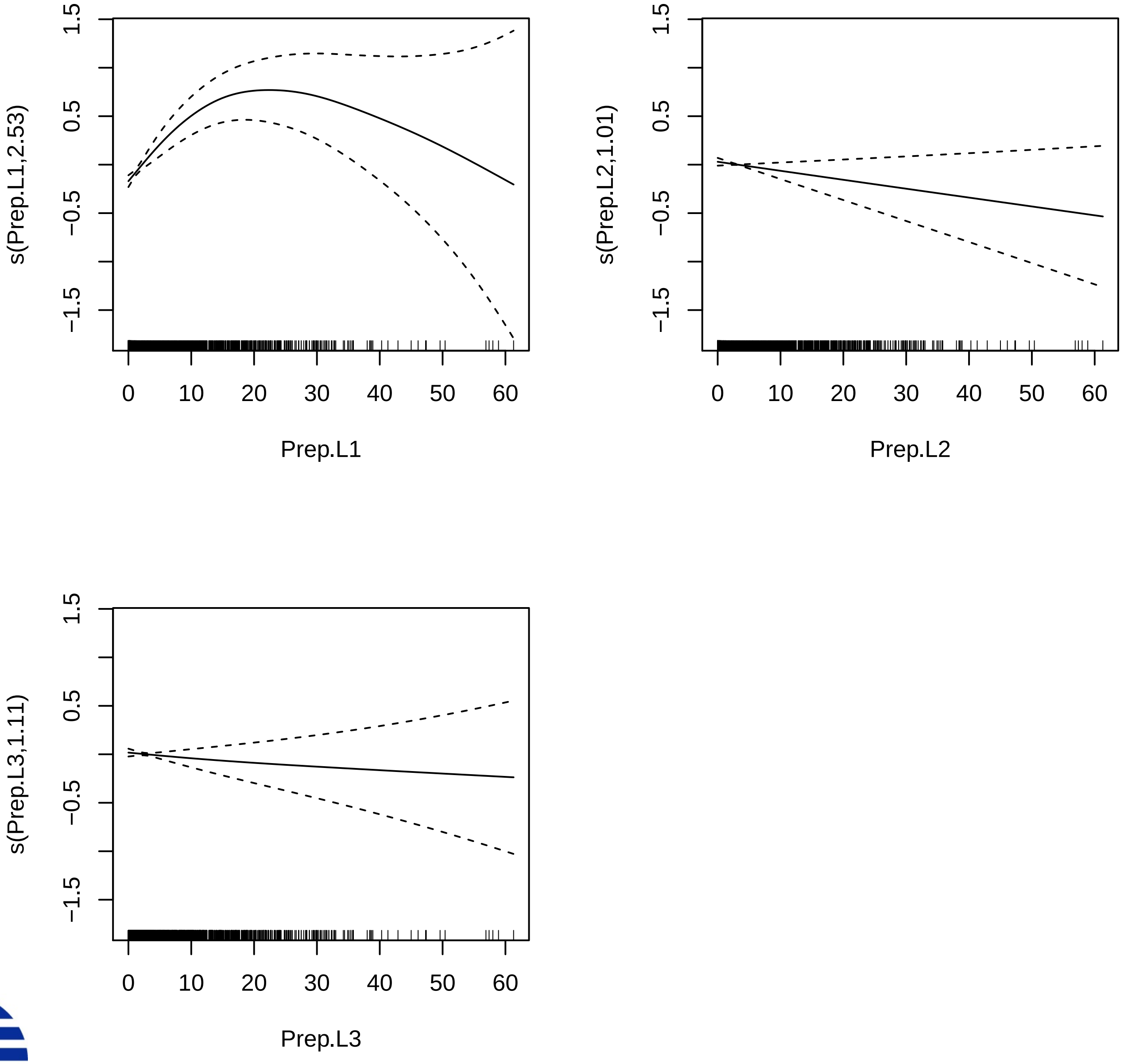
```
> Tryvann.dat<-read.table("/nr/user/aldrin/Sharp/data/Tryvann.dat")
>
> ### load the mgcv library
> require(mgcv)
>
> gamobj<-gam(P01~P01.L1+P01.L2+P01.L3+
s(Prep.L1)+s(Prep.L2)+s(Prep.L3),
data=Tryvann.dat,family=binomial(link=logit))
> summary(gamobj)

Family: binomial
Link function: logit

Formula:
P01 ~ P01.L1 + P01.L2 + P01.L3 + s(Prep.L1) + s(Prep.L2) + s(Prep.L3)
```

```
Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.64629    0.08114  -7.965 1.65e-15 ***
P01.L1       0.95902    0.08893  10.784  < 2e-16 ***
P01.L2       0.31057    0.08590   3.615    3e-04 ***
P01.L3       0.41355    0.08487   4.873 1.10e-06 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
Approximate significance of smooth terms:
             edf Ref.df Chi.sq  p-value
s(Prep.L1) 2.530  3.154 34.803 1.67e-07 ***
s(Prep.L2) 1.008  1.015  2.208    0.140
s(Prep.L3) 1.110  1.212  0.564    0.532
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.128   Deviance explained = 9.81%
UBRE score = 0.2394  Scale est. = 1         n = 3420
>
```

Plot results

```
> pdf("GAMlogistic.pdf")
> par(mfrow=c(2,2))
> plot(gamobj)
> dev.off()
null device
          1
```


s(Prep.L1,2.53)
1.5
0.5
−0.5
−1.5
0
10
20
30
40
50
60
Prep.L1
s(Prep.L2,1.01)
1.5
0.5
−0.5
−1.5
0
10
20
30
40
50
60
Prep.L2
s(Prep.L3,1.11)
1.5
0.5
−0.5
−1.5
0
10
20
30
40
50
60
Prep.L3

Ex: Simulated data y $\sim$ Poisson

```
> n<-200
> x1<-rnorm(n)       # N(0,1)
> x2<-runif(n,-10,10)  # Uniform(-10,10)
> x3<-rnorm(n)
>
> eta<- 1 + 2*x1 - 0.2*x2^2 + 0*x3
> mu<-exp(eta)
>
> y<-rpois(n=n,lambda=mu)
>
> data.obj<-data.frame(y=y,x1=x1,x2=x2,x3=x3)
>
> gamobj<-gam(y~s(x1)+s(x2)+s(x3),family=poisson(link=log),
data=data.obj)
```

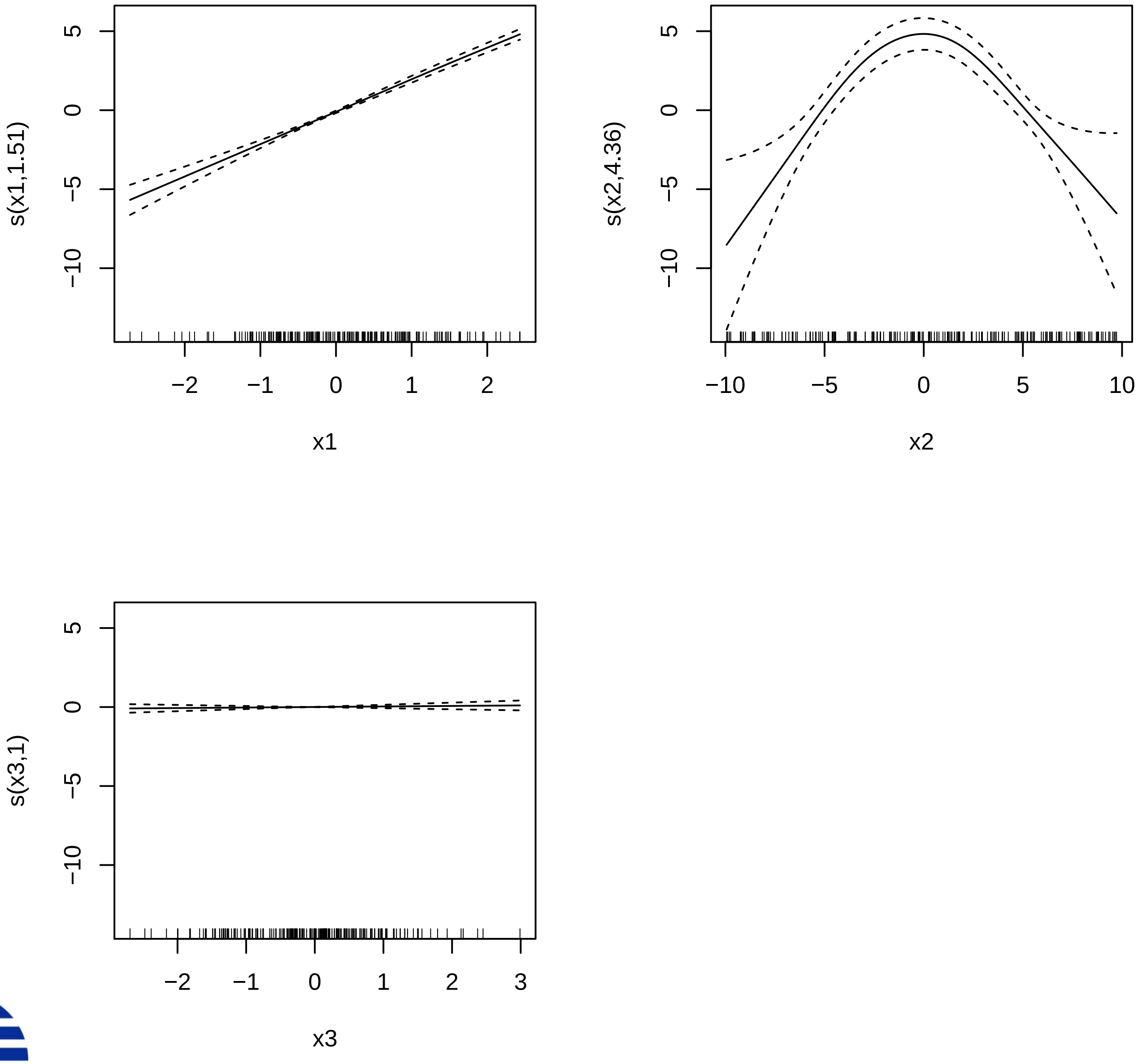
Results

```
> summary(gamobj)

Family: poisson
Link function: log

Formula:
y ~ s(x1) + s(x2) + s(x3)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.7679     0.5131  -7.343 2.08e-13 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
Approximate significance of smooth terms:
        edf Ref.df  Chi.sq p-value
s(x1) 1.508  1.849 657.653  <2e-16 ***
s(x2) 4.357  5.138 119.824  <2e-16 ***
s(x3) 1.000  1.000   0.427   0.514
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.992   Deviance explained = 97.4%
UBRE score = -0.50231  Scale est. = 1         n = 200
```

Plot results

```
> pdf("GAMPoisson.pdf")
> par(mfrow=c(2,2))
> plot(gamobj)
> dev.off()
null device
          1
```


s(x1,1.51)
5
0
−5
−10
−2
−1
0
1
2
x1
s(x2,4.36)
5
0
−5
−10
−10
−5
0
5
10
x2
s(x3,1)
5
0
−5
−10
−2
−1
0
1
2
3
x3

Option: Terms can be forced to be linear

```
> ### The terms can be forced to be linear
> gamobj<-gam(log(O3)~vh+wind+s(humidity)+s(temp)+s(ibh)+
s(dpg)+ibt+s(vis)+s(doy),
family=gaussian(link=identity),data=ozone)
> summary(gamobj)

Family: gaussian
Link function: identity
Formula:
log(O3) ~ vh + wind + s(humidity) + s(temp) + s(ibh) + s(dpg) +
    ibt + s(vis) + s(doy)
Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.0874893  2.4699664  -2.465  0.01427 *
vh           0.0014501  0.0004384   3.308  0.00105 **
wind        -0.0311454  0.0101294  -3.075  0.00230 **
ibt          0.0006986  0.0010388   0.673  0.50177
```

```
Approximate significance of smooth terms:
              edf Ref.df      F  p-value
s(humidity) 2.398  3.017  2.476 0.061153 .
s(temp)     3.811  4.753  4.081 0.001638 **
s(ibh)      2.761  3.378  5.431 0.000711 ***
s(dpg)      3.354  4.259 14.320 3.23e-11 ***
s(vis)      4.559  5.627  6.689 2.15e-06 ***
s(doy)      4.571  5.692 25.575  < 2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.826   Deviance explained = 83.9%
GCV score = 0.10575  Scale est. = 0.097591  n = 330
```

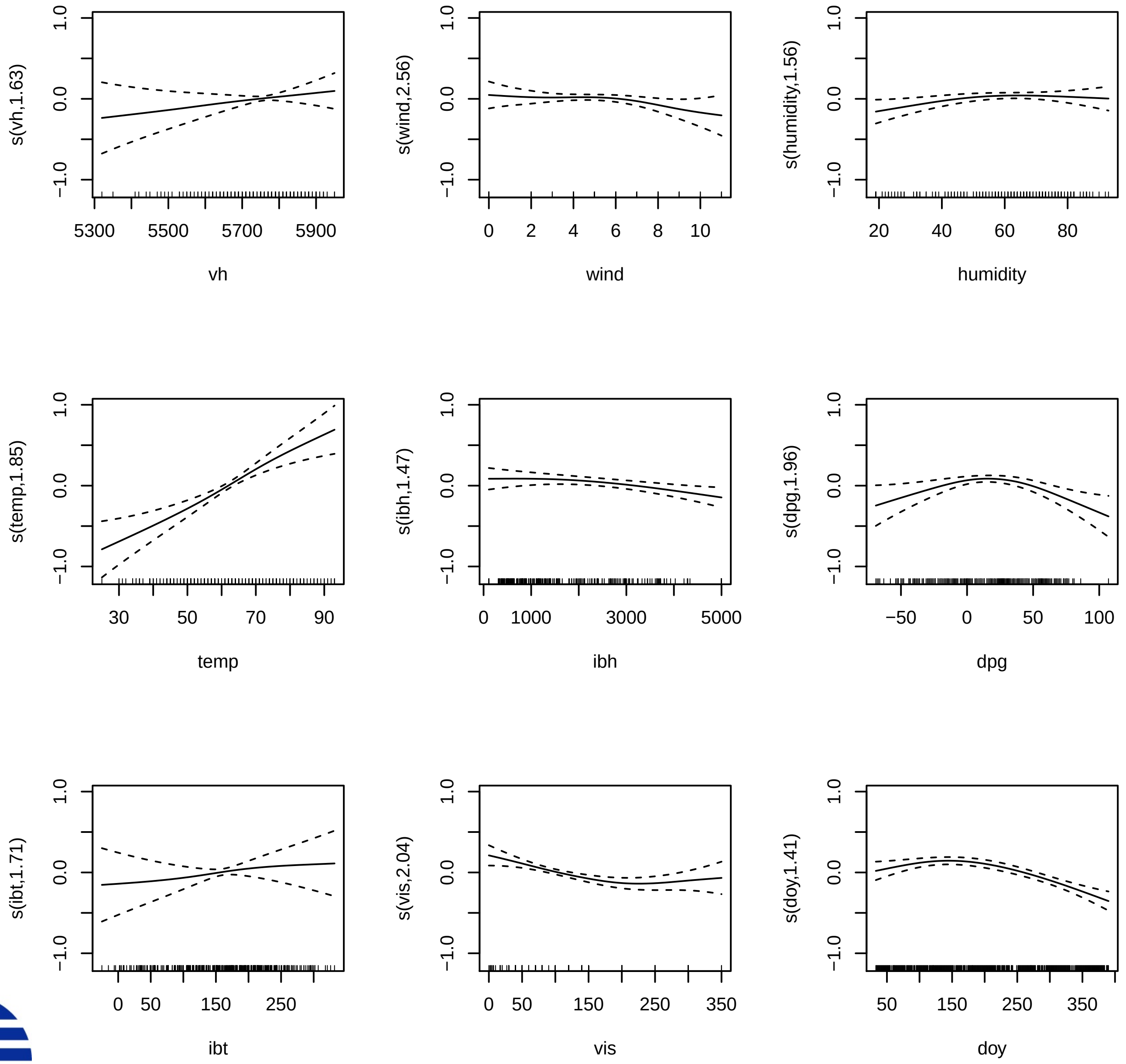
Controlling the smoothness by the sp option

```
> gamobj<-gam(log(O3)~s(vh)+s(wind)+s(humidity)+s(temp)+s(ibh)+
s(dpg)+s(ibt)+s(vis)+s(doy),sp=rep(0.5,9),
family=gaussian(link=identity),data=ozone)
> summary(gamobj)

Family: gaussian
Link function: identity

Formula:
log(O3) ~ s(vh) + s(wind) + s(humidity) + s(temp) + s(ibh) +
    s(dpg) + s(ibt) + s(vis) + s(doy)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.21297    0.01954   113.2   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
Approximate significance of smooth terms:
              edf Ref.df      F  p-value
s(vh)       1.633  2.022  0.679 0.509227
s(wind)     2.562  3.241  1.189 0.314667
s(humidity) 1.559  1.928  2.478 0.087614 .
s(temp)     1.846  2.299 13.333 7.16e-07 ***
s(ibh)      1.467  1.771  4.698 0.012703 *
s(dpg)      1.957  2.506 10.365 8.06e-06 ***
s(ibt)      1.705  2.107  0.437 0.656654
s(vis)      2.044  2.547  7.610 0.000172 ***
s(doy)      1.407  1.727 14.618 3.45e-06 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.775   Deviance explained = 78.6%
GCV score = 0.13294  Scale est. = 0.12602   n = 330
```

Plot results

```
> pdf("GAMsmoothozone.pdf")
> par(mfrow=c(3,3))
> plot(gamobj)
> dev.off()
null device
          1
```

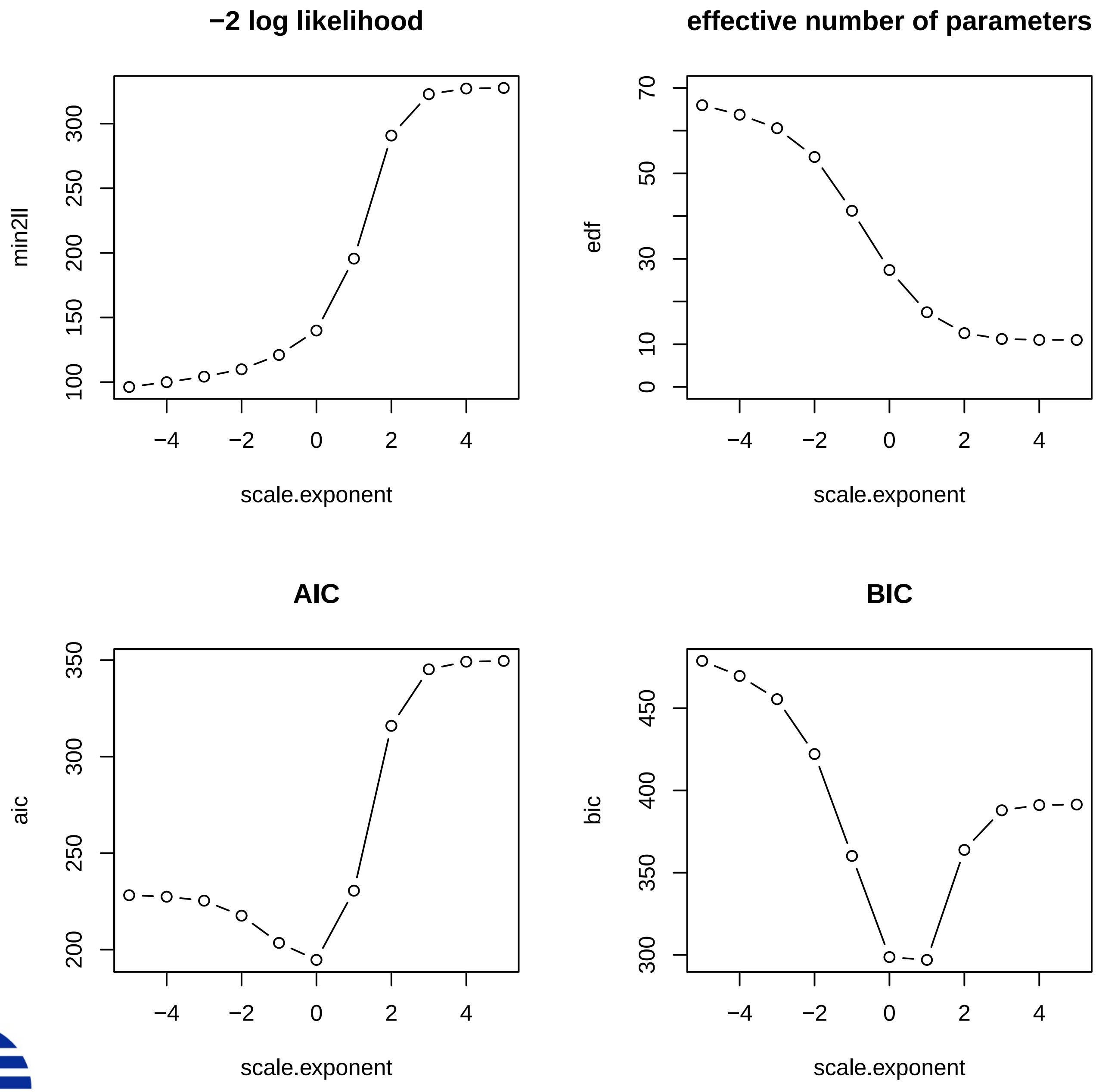

s(vh,1.63)
1.0
0.0
−1.0
5300
5500
5700
5900
vh
s(wind,2.56)
0
2
4
6
8
10
wind
s(humidity,1.56)
20
40
60
80
humidity
s(temp,1.85)
30
50
70
90
temp
s(ibh,1.47)
0
1000
3000
5000
ibh
s(dpg,1.96)
−50
0
50
100
dpg
s(ibt,1.71)
0
50
150
250
ibt
s(vis,2.04)
0
50
150
250
350
vis
s(doy,1.41)
50
150
250
350
doy

AIC or BIC to choose degree of smoothness

```
> gamobj<-gam(log(O3)~s(vh)+s(wind)+s(humidity)+s(temp)+s(ibh)+
s(dpg)+s(ibt)+s(vis)+s(doy),
family=gaussian(link=identity),data=ozone)
> sp<-gamobj$sp
>
> tuning.scale<-c(1e-5,1e-4,1e-3,1e-2,1e-1,1e0,1e1,1e2,1e3,1e4,1e5)
> ###tuning.scale<-c(0.00001,0.0001,0.001,0.01,0.1,1,10,100,1000,10000
> scale.exponent<-log10(tuning.scale)
> n.tuning<-length(tuning.scale)
> edf<-rep(NA,n.tuning)
> min2ll<-rep(NA,n.tuning)
> aic<-rep(NA,n.tuning)
> bic<-rep(NA,n.tuning)
> for (i in 1:n.tuning) {
+   gamobj<-gam(log(O3)~s(vh)+s(wind)+s(humidity)+s(temp)+s(ibh)+
s(dpg)+s(ibt)+s(vis)+s(doy),
family=gaussian(link=identity),data=ozone,
sp=tuning.scale[i]*sp)
+   min2ll[i]<--2*logLik(gamobj)
+   edf[i]<-sum(gamobj$edf)+1
+   aic[i]<-AIC(gamobj)
+   bic[i]<-BIC(gamobj)
+ }
```

Plot results

```
> pdf("AICBICozone.pdf")
> par(mfrow=c(2,2))
> plot(scale.exponent,min2ll,type="b",main="-2 log likelihood")
> plot(scale.exponent,edf,ylim=c(0,70),type="b",main="effective number
> plot(scale.exponent,aic,type="b",main="AIC")
> plot(scale.exponent,bic,type="b",main="BIC")
> dev.off()
null device
          1
```


−2 log likelihood
min2ll
100
150
200
250
300
−4
−2
0
2
4
scale.exponent
effective number of parameters
edf
0
10
30
50
70
−4
−2
0
2
4
scale.exponent
AIC
aic
200
250
300
350
−4
−2
0
2
4
scale.exponent
BIC
bic
300
350
400
450
−4
−2
0
2
4
scale.exponent

Scaling corresponding to minimum BIC

```
> min.bic<-1e100
> opt.tuning.scale<-NULL
> for (i in 1:n.tuning) {
+   if (bic[i]<min.bic) {
+     min.bic<-bic[i]
+     opt.tuning.scale<-tuning.scale[i]
+   }
+ }
> opt.sp<-opt.tuning.scale*sp
```

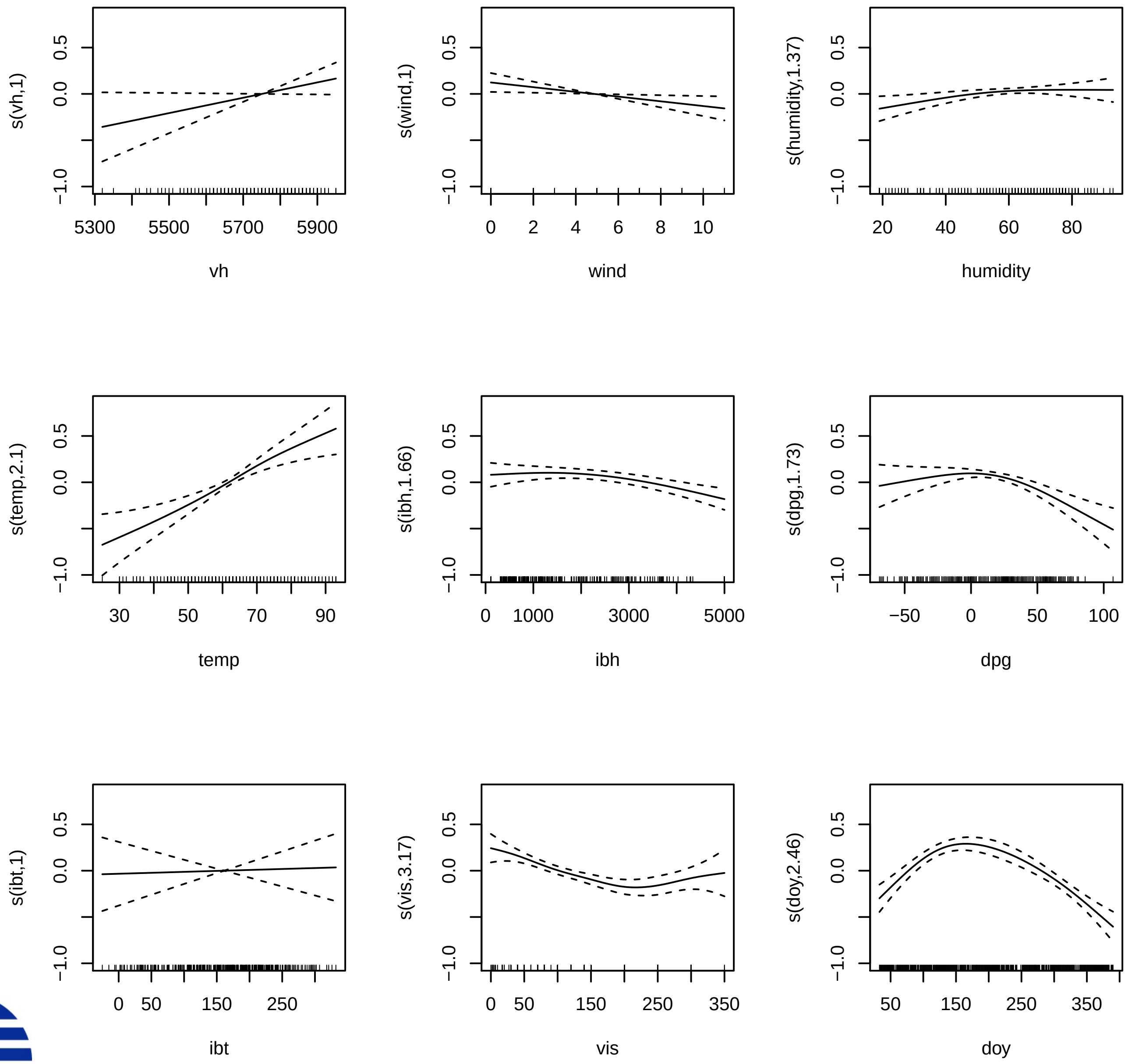
Estimate final model with optimal value of sp

```
> gamobj<-gam(log(O3)~s(vh)+s(wind)+s(humidity)+s(temp)+s(ibh)+
s(dpg)+s(ibt)+s(vis)+s(doy),
family=gaussian(link=identity),data=ozone,sp=opt.sp)
> summary(gamobj)

Family: gaussian
Link function: identity

Formula:
log(O3) ~ s(vh) + s(wind) + s(humidity) + s(temp) + s(ibh) +
    s(dpg) + s(ibt) + s(vis) + s(doy)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.21297    0.01838   120.4   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
Approximate significance of smooth terms:
              edf Ref.df      F  p-value
s(vh)       1.000  1.000  3.650  0.05699 .
s(wind)     1.002  1.004  5.834  0.01617 *
s(humidity) 1.369  1.646  2.349  0.10744
s(temp)     2.098  2.649  9.436 1.52e-05 ***
s(ibh)      1.657  2.030  6.658  0.00139 **
s(dpg)      1.727  2.192 15.471 1.39e-07 ***
s(ibt)      1.000  1.000  0.037  0.84715
s(vis)      3.171  3.965  7.033 2.10e-05 ***
s(doy)      2.462  3.196 27.265  < 2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

R-sq.(adj) =  0.801   Deviance explained =   81%
GCV score = 0.11734  Scale est. = 0.11147   n = 330
```

Plot results

```
> pdf("GAMoptsmoothozone.pdf")
> par(mfrow=c(3,3))
> plot(gamobj)
> dev.off()
null device
          1
```


s(vh,1)
−1.0
0.0
0.5
5300
5500
5700
5900
vh
s(wind,1)
−1.0
0.0
0.5
0
2
4
6
8
10
wind
s(humidity,1.37)
−1.0
0.0
0.5
20
40
60
80
humidity
s(temp,2.1)
−1.0
0.0
0.5
30
50
70
90
temp
s(ibh,1.66)
−1.0
0.0
0.5
0
1000
3000
5000
ibh
s(dpg,1.73)
−1.0
0.0
0.5
−50
0
50
100
dpg
s(ibt,1)
−1.0
0.0
0.5
0
50
150
250
ibt
s(vis,3.17)
−1.0
0.0
0.5
0
50
150
250
350
vis
s(doy,2.46)
−1.0
0.0
0.5
50
150
250
350
doy