

Generalized Additive Models (GAMs)

CMDA 4654 Project 2

Group 19: Brady Bolton, Eryk Jesse, Charles Lee, Dan Schlicht

Generalized Additive Models



Dr Gavin Simpson 🙌🇪🇺

@ucfagls

Replying to @millerdl

140 char vrsn

- 1 GAMs are just GLMs
- 2 GAMs fit wiggly terms
- 3 use + s(foo) not foo in frmla
- 4 use method = "REML"
- 5 gam.check()

2:37 PM · Mar 16, 2017 · TweetDeck

Generalized Additive Models

- Type of generalized linear model
- Response variable depends on smooth functions $f_i(x_i)$
- General structure of a GAM:
$$g(E(Y)) = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$
- Smooth functions can be many different things (polynomials, splines, weighted means, etc)

Generalized Additive Models

Generalized Additive Models

Generalized Additive Models

Generalized Additive Models

Generalized Additive Models

Generalized Additive Models

Generalized Additive Models in R

- Two options for packages, `mgcv` and `gam`
- `mgcv` is more commonly used and better supported

Description

`mgcv` provides functions for generalized additive modelling (`gam` and `bam`) and generalized additive mixed modelling (`gamm`, and `random.effects`). The term GAM is taken to include any model dependent on unknown smooth functions of predictors and estimated by quadratically penalized (possibly quasi-) likelihood maximization. Available distributions are covered in `family.mgcv` and available smooths in `smooth.terms`.

Particular features of the package are facilities for automatic smoothness selection (Wood, 2004, 2011), and the provision of a variety of smooths of more than one variable. User defined smooths can be added. A Bayesian approach to confidence/credible interval calculation is provided. Linear functionals of smooths, penalization of parametric model terms and linkage of smoothing parameters are all supported. Lower level routines for generalized ridge regression and penalized linearly constrained least squares are also available. In addition to the main modelling functions, `jagam` provided facilities to ease the set up of models for use with JAGS, while `ginla` provides marginal inference via a version of Integrated Nested Laplace Approximation.

Example - Simulated Data

```
library(mgcv)
set.seed(0)
sim_data <- gamSim(1, n = 400, dist="normal", scale=2)
```

Gu & Wahba 4 term additive model

```
head(sim_data)
```

	y	x0	x1	x2	x3	f	f0
1	5.114211	0.8966972	0.1478457	0.34826473	0.04572472	7.962274	0.6377368
2	2.175828	0.2655087	0.6588776	0.85868745	0.36652658	5.514517	1.4814113
3	6.334878	0.3721239	0.1850700	0.03443876	0.74139303	3.576406	1.8407682
4	6.853276	0.5728534	0.9543781	0.97099715	0.93350625	8.692625	1.9478442
5	7.743879	0.9082078	0.8978485	0.74511014	0.67320995	8.752859	0.5687870
6	13.920886	0.2016819	0.9436971	0.27325524	0.70135711	16.190349	1.1841037
	f1	f2	f3				
1	1.344055	5.980482e+00	0				
2	3.735028	2.980780e-01	0				
3	1.447937	2.877006e-01	0				
4	6.744695	8.611364e-05	0				
5	6.023672	2.160400e+00	0				
6	6.602142	8.404104e+00	0				

Example - Simulated Data

```
fit <- gam(y ~ s(x0) + s(x1) + s(x2) + s(x3), data = sim_data)
summary(fit)
```

Family: gaussian
Link function: identity

Formula:
y ~ s(x0) + s(x1) + s(x2) + s(x3)

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.9150	0.1049	75.44	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:

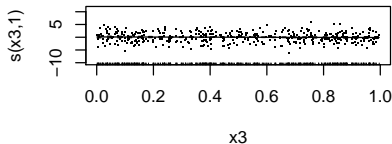
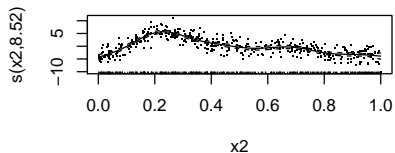
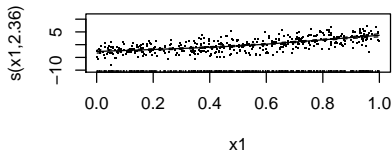
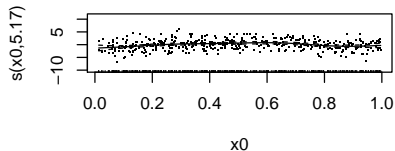
	edf	Ref.df	F	p-value
s(x0)	5.173	6.287	4.564	0.000138 ***
s(x1)	2.357	2.927	103.053	< 2e-16 ***
s(x2)	8.517	8.931	84.308	< 2e-16 ***
s(x3)	1.000	1.000	0.441	0.506929

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.726 Deviance explained = 73.7%
GCV = 4.611 Scale est. = 4.4029 n = 400

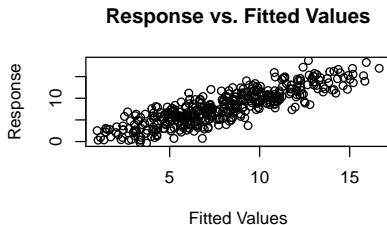
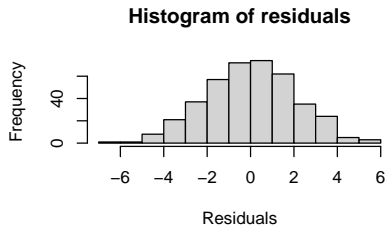
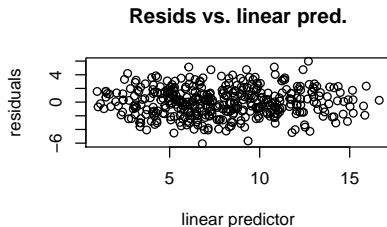
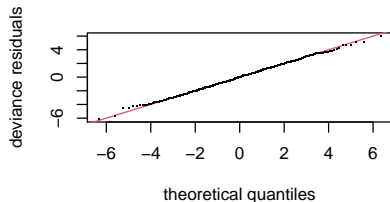
Example - Simulated Data

```
plot(fit, pages=1, residuals=TRUE)
```



Example - Simulated Data

```
gam.check(fit)
```



Example - mtcars

```
data("mtcars")
mtcars_gam <-
  gam(mpg ~ s(displ), data = mtcars, method = "REML")
summary(mtcars_gam)
```

Family: gaussian
Link function: identity

Formula:
mpg ~ s(displ)

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	20.0906	0.3788	53.04	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:

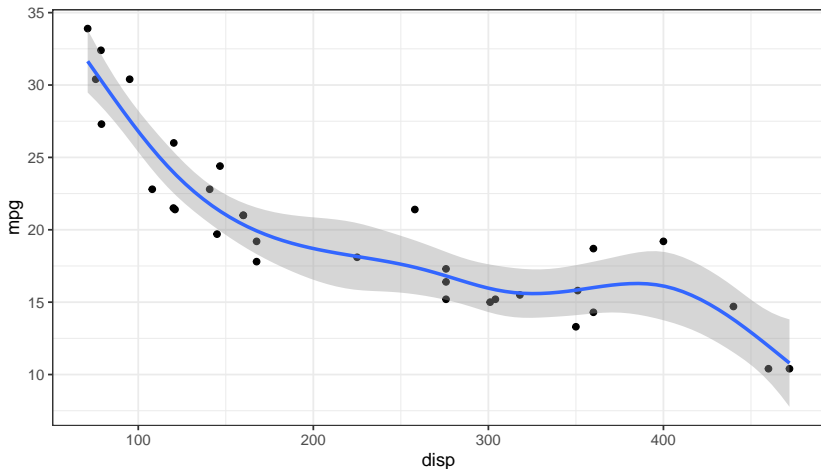
	edf	Ref.df	F	p-value
s(displ)	4.884	5.904	36.3	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.874 Deviance explained = 89.4%
-REML = 74.101 Scale est. = 4.5918 n = 32

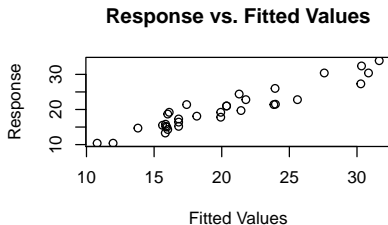
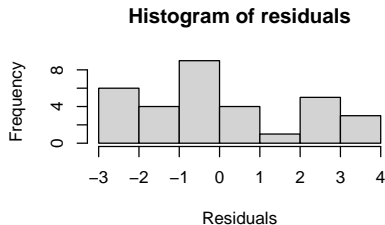
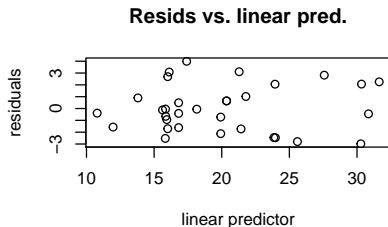
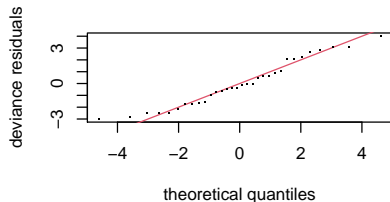
Example - mtcars

```
library(ggplot2)
ggplot(data = mtcars, aes(x = disp, y = mpg)) +
  theme_bw() + geom_point() +
  geom_smooth(method = "gam", formula = y ~ s(x))
```



Example - mtcars

```
gam.check(mtcars_gam)
```



Example - mtcars

Example - iris

```
data("iris")
```

Example - iris

Example - iris

Example - iris

Example - Health Data

```
# df <- read.csv('health_data.csv')  
# show the data
```

Example - Health Data

Example - Health Data

Example - Health Data

Example - Health Data

Example - Health Data

Conclusion

Conclusion

Conclusion

References