Introduction to Generalized Additive Models

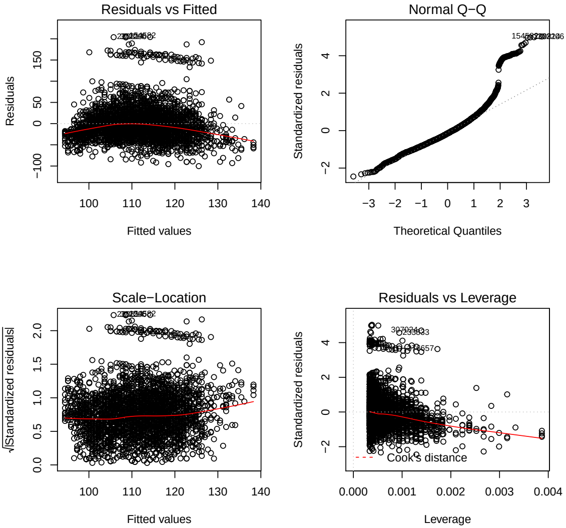
ISLR Chapter 7

April 3, 2017

Moving Beyond Linearity

Wage data from ISLR `data(Wage)`

Residual plots from Simple Linear Regression of Wage on Age:



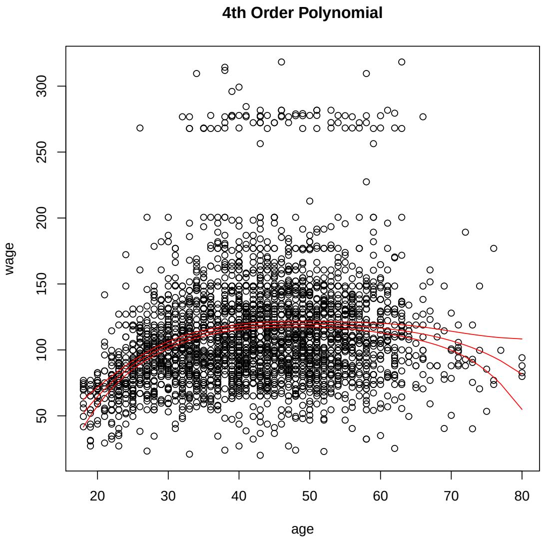
Polynomial Regression

```
> summary(lm(wage ~ age + I(age^2) + I(age^3) + I(age^4),
            data=Wage))
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.842e+02  6.004e+01  -3.067 0.002180 **
age          2.125e+01  5.887e+00   3.609 0.000312 ***
I(age^2)    -5.639e-01  2.061e-01  -2.736 0.006261 **
I(age^3)     6.811e-03  3.066e-03   2.221 0.026398 *
I(age^4)    -3.204e-05  1.641e-05  -1.952 0.051039 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 39.91 on 2995 degrees of freedom
Multiple R-squared:  0.08626,^^IAdjusted R-squared:  0.085
F-statistic: 70.69 on 4 and 2995 DF,  p-value: < 2.2e-16
```

Orthogonal Polynomial

```
> summary(lm(wage ~ poly(age,4),data=Wage))

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    111.7036     0.7287 153.283  < 2e-16 ***
poly(age, 4)1  447.0679    39.9148  11.201  < 2e-16 ***
poly(age, 4)2 -478.3158    39.9148 -11.983  < 2e-16 ***
poly(age, 4)3  125.5217    39.9148   3.145  0.00168 **
poly(age, 4)4  -77.9112    39.9148  -1.952  0.05104 .
---

Residual standard error: 39.91 on 2995 degrees of freedom
Multiple R-squared:  0.08626,^^IAdjusted R-squared:  0.085
F-statistic: 70.69 on 4 and 2995 DF,  p-value: < 2.2e-16
```

Fitted Values



Problems

- Higher order terms may be needed to fit data globally

Problems

- Higher order terms may be needed to fit data globally
- generally do not go above 3rd order or 4th order polynomial - may be too flexible

Problems

- Higher order terms may be needed to fit data globally
- generally do not go above 3rd order or 4th order polynomial - may be too flexible
- fit piece-wise polynomials over different ranges

Problems

- Higher order terms may be needed to fit data globally
- generally do not go above 3rd order or 4th order polynomial - may be too flexible
- fit piece-wise polynomials over different ranges
- is function continuous where they join?

Problems

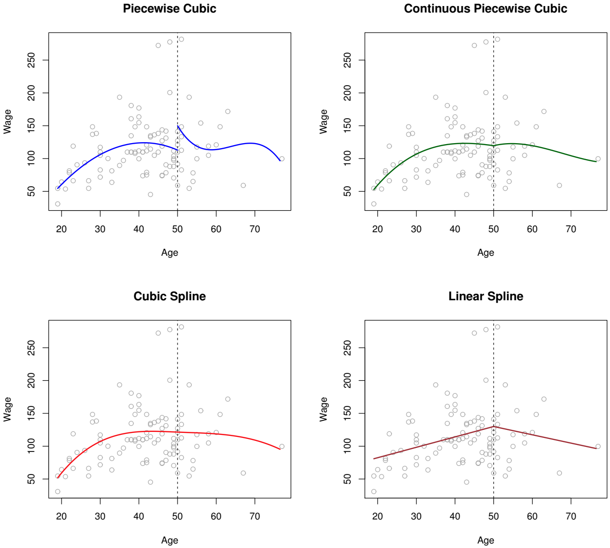
- Higher order terms may be needed to fit data globally
- generally do not go above 3rd order or 4th order polynomial - may be too flexible
- fit piece-wise polynomials over different ranges
- is function continuous where they join?
- is function differentiable where they join?

Problems

- Higher order terms may be needed to fit data globally
- generally do not go above 3rd order or 4th order polynomial - may be too flexible
- fit piece-wise polynomials over different ranges
- is function continuous where they join?
- is function differentiable where they join?

Add constraints - lose degrees of freedom

Piecewise Polynomials



Spline Basis

Alternative way to represent the model so that we have continuity, continuous first and second derivatives is

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_1^2 + \beta_3 x_i^3 + h(x_i, \xi)\beta_4 + \epsilon_i$$

where ξ is a "knot"' in a truncated cubic basis function

$$h(x_i, \xi) \equiv (x_i - \xi)_+^3 = \begin{cases} (x_i - \xi)^3 & \text{if } x_i > \xi \\ 0 & \text{otherwise} \end{cases}$$

Spline Basis

Alternative way to represent the model so that we have continuity, continuous first and second derivatives is

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_1^2 + \beta_3 x_i^3 + h(x_i, \xi)\beta_4 + \epsilon_i$$

where ξ is a "knot"' in a truncated cubic basis function

$$h(x_i, \xi) \equiv (x_i - \xi)_+^3 = \begin{cases} (x_i - \xi)^3 & \text{if } x_i > \xi \\ 0 & \text{otherwise} \end{cases}$$

We can add additional terms that each with 1 degree of freedom

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_1^2 + \beta_3 x_i^3 + \sum_k^K h(x_i, \xi_k)\beta_4 + \epsilon_i$$

Spline Basis

Alternative way to represent the model so that we have continuity, continuous first and second derivatives is

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_1^2 + \beta_3 x_i^3 + h(x_i, \xi)\beta_4 + \epsilon_i$$

where ξ is a "knot"' in a truncated cubic basis function

$$h(x_i, \xi) \equiv (x_i - \xi)_+^3 = \begin{cases} (x_i - \xi)^3 & \text{if } x_i > \xi \\ 0 & \text{otherwise} \end{cases}$$

We can add additional terms that each with 1 degree of freedom

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_1^2 + \beta_3 x_i^3 + \sum_k^K h(x_i, \xi_k)\beta_4 + \epsilon_i$$

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis
- natural splines (add more constraints so that function is linear outside range of data)

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis
- natural splines (add more constraints so that function is linear outside range of data)
- smoothing splines

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis
- natural splines (add more constraints so that function is linear outside range of data)
- smoothing splines

Choice of knots and/or degrees of freedom?

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis
- natural splines (add more constraints so that function is linear outside range of data)
- smoothing splines

Choice of knots and/or degrees of freedom? Smoothing splines place a knot at each data point, but adds a penalty to prevent over-fitting:

$$\sum (Y_i - g(x_i))^2 + \lambda \int g^{''}(t)^2 \, dt$$

Splines

- B-splines (reduces multicollinearity between terms from truncated basis) `splines` package in R to use `bs(x)` to construct basis
- natural splines (add more constraints so that function is linear outside range of data)
- smoothing splines

Choice of knots and/or degrees of freedom? Smoothing splines place a knot at each data point, but adds a penalty to prevent over-fitting:

$$\sum (Y_i - g(x_i))^2 + \lambda \int g^{''}(t)^2 \, dt$$

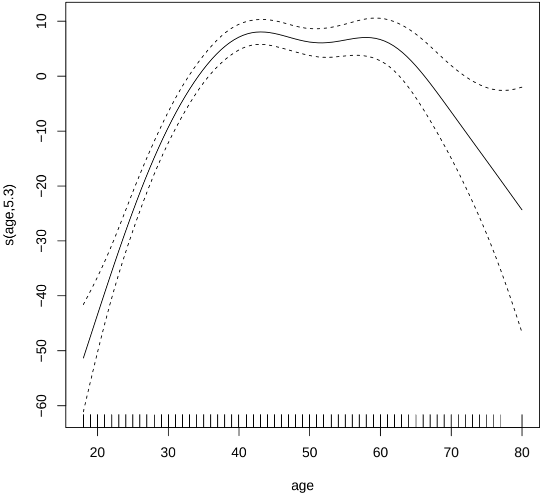
This can be reformulated as a Bayesian model with a Gaussian g-prior. packages use LOOCV or GCV to choose λ

Fitting GAMs in R

```
> wage.gam = gam(wage ~ s(age), data=Wage)
> summary(wage.gam)
Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 111.7036     0.7282   153.4   <2e-16 ***

Approximate significance of smooth terms:
         edf Ref.df     F p-value
s(age) 5.298  6.399 44.34  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

R-sq.(adj) =  0.0864   Deviance explained =  8.8%
GCV = 1594.2  Scale est. = 1590.9    n = 3000
```

Fitted Curve

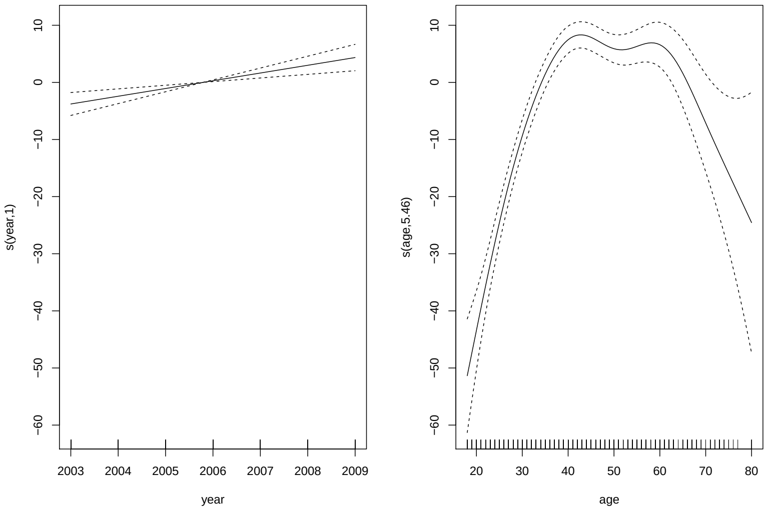
```
plot(wage.gam, rug=T)
```



More terms

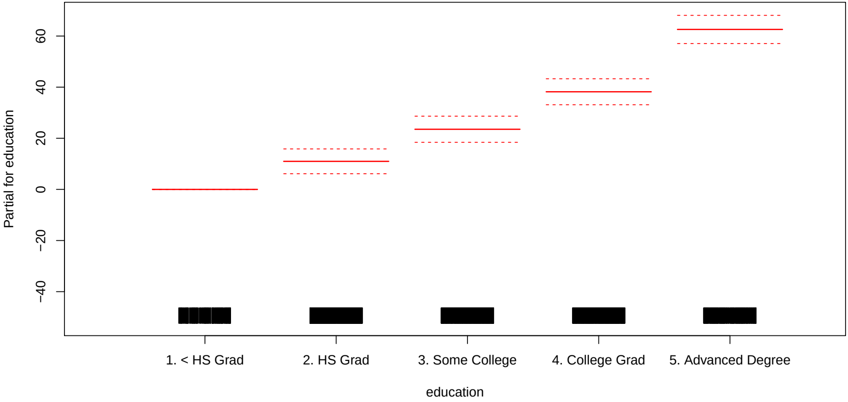
```
> wage.gam2 = gam(wage ~ s(year,k=7) + s(age), data=Wage)
> summary(wage.gam2)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 111.7036     0.7266   153.7   <2e-16 ***
---

Approximate significance of smooth terms:
          edf Ref.df     F  p-value
s(year) 1.000  1.000 14.18 0.000169 ***
s(age)  5.462  6.568 43.37  < 2e-16 ***

R-sq.(adj) =  0.0905   Deviance explained = 9.24%
GCV = 1587.7  Scale est. = 1583.7    n = 3000
```

Year and Age Smooth fits



Showing Factors

```
wage.gam3 = gam(wage ~ s(year,k=7) + s(age) + education,
                data=Wage)
termplot(wage.gam3, se=T, rug=T, ask=F, col.se=2)
```



Summary

GAMS:

- Allow flexible non-linear functions of predictors. Do not need to try various transformations or polymomials to capture relationships

Summary

GAMS:

- Allow flexible non-linear functions of predictors. Do not need to try various transformations or polymomials to capture relationships
- May be used to suggest parametric models (i.e linear or quadratic may be fine)

Summary

GAMS:

- Allow flexible non-linear functions of predictors. Do not need to try various transformations or polymomials to capture relationships
- May be used to suggest parametric models (i.e linear or quadratic may be fine)
- nonlinear functions can extend to multiple predictors for interactions, but soon run into curse of dimensionality

Summary

GAMS:

- Allow flexible non-linear functions of predictors. Do not need to try various transformations or polymomials to capture relationships
- May be used to suggest parametric models (i.e linear or quadratic may be fine)
- nonlinear functions can extend to multiple predictors for interactions, but soon run into curse of dimensionality
- Nonlinear fits can lead to improved prediction

Summary

GAMS:

- Allow flexible non-linear functions of predictors. Do not need to try various transformations or polymomials to capture relationships
- May be used to suggest parametric models (i.e linear or quadratic may be fine)
- nonlinear functions can extend to multiple predictors for interactions, but soon run into curse of dimensionality
- Nonlinear fits can lead to improved prediction
- Additive functions may be more interpretable