Unit 6: Generalized Additive Models

Henry Scharf

STAT 410

Polynomial regression

Basis functions

Putting the “Generalized” in GAMs

Multiple predictors

Unit goals

1. Be able to express generalized additive models (GAMs) both with mathematical notation and in R.
2. Be able to fit GAMs with multiple predictors to data in R and interpret summary output.
3. Be able to check for evidence that statistical assumptions of GAMs are violated.
4. Be able to generate predictions for new combinations of predictor/explanatory variables in R with uncertainty.

Polynomial regression

Wage data

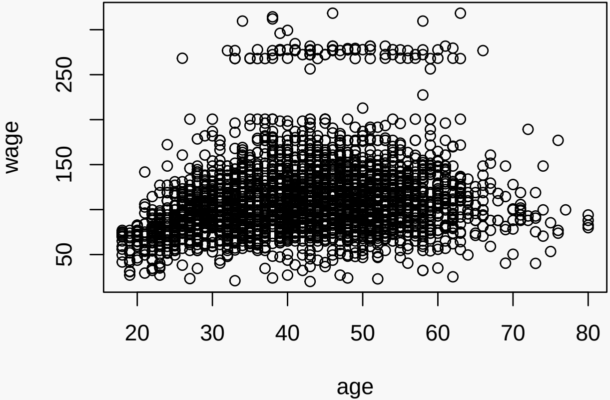
"...Wage data set, which contains income and demographic information for males who reside in the central Atlantic region of the United States." –p. 291 ISLR

```
library(ISLR2)
head(Wage) ## Wage demographic info from males living in the
##         year age            maritl     race       education             region
## 231655 2006  18 1. Never Married 1. White    1. < HS Grad 2. Middle Atlantic
## 86582  2004  24 1. Never Married 1. White 4. College Grad 2. Middle Atlantic
## 161300 2003  45       2. Married 1. White 3. Some College 2. Middle Atlantic
## 155159 2003  43       2. Married 3. Asian 4. College Grad 2. Middle Atlantic
## 11443  2005  50      4. Divorced 1. White      2. HS Grad 2. Middle Atlantic
## 376662 2008  54       2. Married 1. White 4. College Grad 2. Middle Atlantic
##              jobclass         health health_ins  logwage      wage
## 231655  1. Industrial      1. <=Good      2. No 4.318063  75.04315
## 86582  2. Information 2. >=Very Good      2. No 4.255273  70.47602
## 161300  1. Industrial      1. <=Good     1. Yes 4.875061 130.98218
## 155159 2. Information 2. >=Very Good     1. Yes 5.041393 154.68529
## 11443  2. Information      1. <=Good     1. Yes 4.318063  75.04315
## 376662 2. Information 2. >=Very Good     1. Yes 4.845098 127.11574
```

Wage data

(1) How would you describe the shape of the relationship between age and wage?

```
plot(wage ~ age, data = Wage)
```



Two ways to fit a quadratic function

The following two models are equivalent in that they result in the same fit.

Option 1

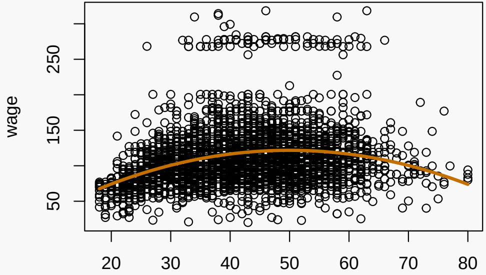
$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i$$

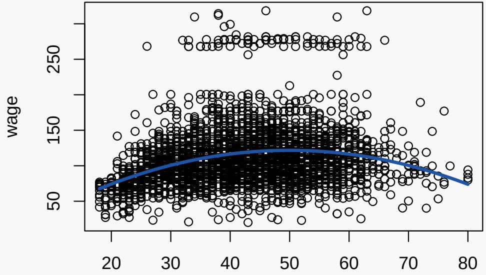
```
fit1 <- lm(wage ~ age + I(age^2),
           data = Wage)
```

Option 2

$$y_i = \alpha_0 + \alpha_1 (x_i - 50) + \alpha_2 (x_i - 50)^2 + \varepsilon_i$$

```
fit2 <- lm(wage ~ I(age - 50) + I((age - 50)^2),
           data = Wage)
```





Two ways to fit a quadratic function

```
summary(fit1)
##
## Call:
## lm(formula = wage ~ age + I(age^2), data
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -99.126 -24.309  -5.017  15.494 205.621
##
## Coefficients:
##                 Estimate Std. Error t value
## (Intercept) -10.425224   8.189780  -1.273
## age           5.294030   0.388689  13.620
## I(age^2)     -0.053005   0.004432 -11.960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '
##
## Residual standard error: 39.99 on 2997 de
## Multiple R-squared:  0.08209,    Adjusted
```

```
summary(fit2)
##
## Call:
## lm(formula = wage ~ I(age - 50) + I((age - 50)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -99.126 -24.309  -5.017  15.494 205.621
##
## Coefficients:
##                   Estimate Std. Error t value
## (Intercept)     121.763606   0.957896 127.116
## I(age - 50)      -0.006477   0.086974  -0.074
## I((age - 50)^2)  -0.053005   0.004432 -11.960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.
##
## Residual standard error: 39.99 on 2997 degrees
## Multiple R-squared:  0.08209,    Adjusted R-sq
```

Two ways to fit a quadratic function

We can easily derive the relationship between the β and α coefficients.

Option 1

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i$$

Option 2

$$\begin{aligned} y_i &= \alpha_0 + \alpha_1(x_i - 50) + \alpha_2(x_i - 50)^2 + \varepsilon_i \\ &= \underbrace{\alpha_0 - 50\alpha_1 + 50^2\alpha_2}_{\beta_0} + \underbrace{(\alpha_1 - 100\alpha_2)}_{\beta_1} x_i + \underbrace{\alpha_2}_{\beta_2} x_i^2 + \varepsilon_i \end{aligned}$$

```
coef(fit1)
##  (Intercept)          age     I(age^2)
## -10.42522426   5.29403003  -0.05300507
coef2 <- coef(fit2)
c(coef2[1] - 50 * coef2[2] + 50^2 * coef2[3],
  coef2[2] - 100 * coef2[3],
  coef2[3])
##     (Intercept)     I(age - 50) I((age - 50)^2)
##    -10.42522426      5.29403003     -0.05300507
```

Two ways to fit a quadratic function

One reason you might prefer a certain version is computational stability.

Option 1

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i$$

Option 2

$$\begin{aligned} y_i &= \alpha_0 + \alpha_1(x_i - 50) + \alpha_2(x_i - 50)^2 + \varepsilon_i \\ &= \underbrace{\alpha_0 - 50\alpha_1 + 50^2\alpha_2}_{\beta_0} + \underbrace{(\alpha_1 - 100\alpha_2)}_{\beta_1} x_i + \underbrace{\alpha_2}_{\beta_2} x_i^2 + \varepsilon_i \end{aligned}$$

```
cor(Wage$age, Wage$age^2)
## [1] 0.9866629
car::vif(fit1)
##      age I(age^2)
## 37.74097 37.74097
cor(Wage$age - 50, (Wage$age - 50)^2)
## [1] -0.6861565
car::vif(fit2)
##     I(age - 50) I((age - 50)^2)
##        1.889683        1.889683
```

Third way to fit a quadratic function (in R)

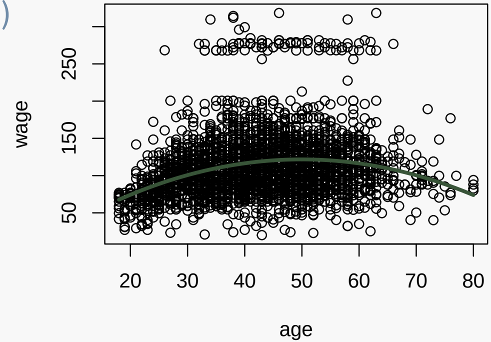
It is possible to find the particular specification that completely eliminates correlation between the predictors.

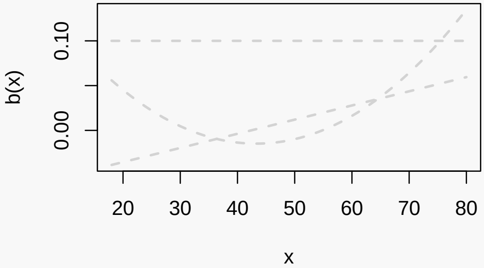
Option 3

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \varepsilon_i$$

```
fit3 <- lm(wage ~ poly(age, 2),
           data = Wage)
cor(fit3$model[, -1])
##                  1             2
## 1  1.000000e+00 -1.455694e-16
## 2 -1.455694e-16  1.000000e+00
```

(2) What could you change in the `lm(...)` function to allow for more “wiggliness” in the fit?





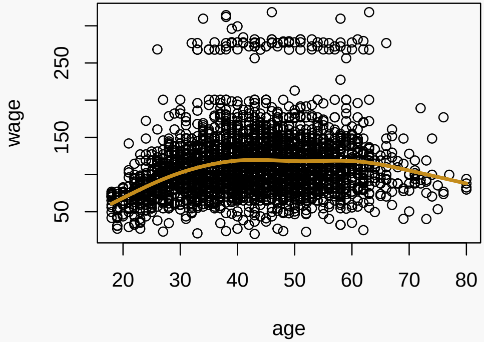
Basis functions

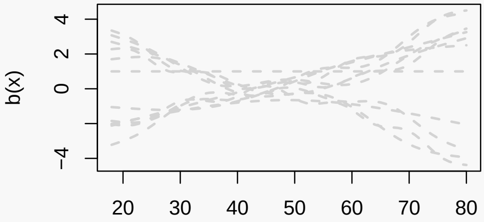
Regression with spline bases

- The next stage is to choose b_i to:
  - have nice computational properties
  - “span” a useful space of functions.
- There are MANY bases out there. Each have Pros and Cons, and often it doesn’t matter too much which one you pick.

$$y_i = \beta_0 + \sum_{i=1}^{k} \beta_i b_i(x_i) + \varepsilon_i$$

```
library(mgcv)
fit4 <- gam(wage ~ s(age, k = 12),
            data = Wage)
```



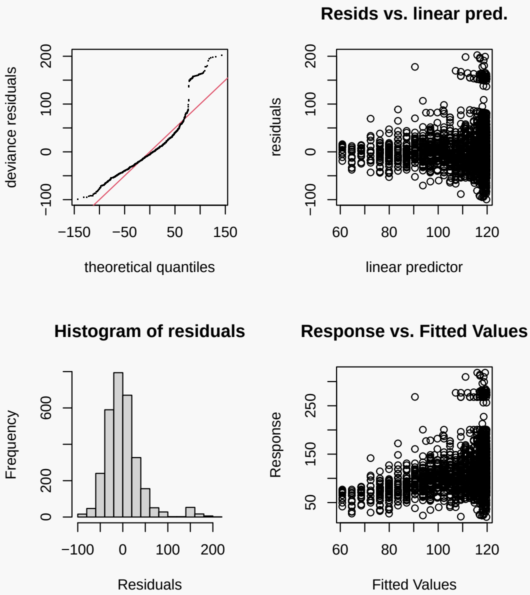


Checking assumptions

1. Normality of residuals
2. Independence of residuals
3. Homoskedasticity: constant variance for residuals
4. *Correct functional* relationship between response and predictors

```
gam.check(fit4)
```

(3) How do things look?

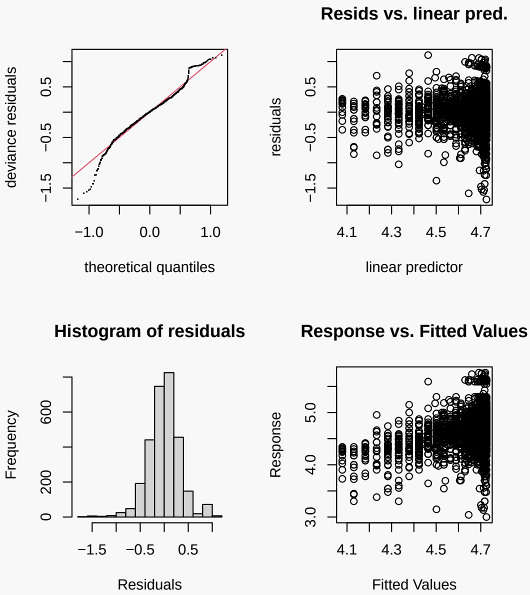


Checking assumptions

1. Normality of residuals
2. Independence of residuals
3. Homoskedasticity: constant variance for residuals
4. *Correct functional* relationship between response and predictors

```
fit5 <- gam(log(wage) ~ s(age, k = 12),
            data = Wage)
gam.check(fit5)
```

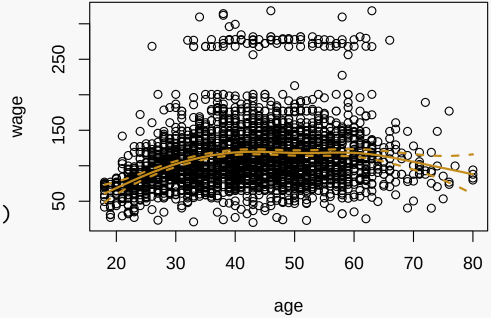
(4) Any better? What else could we try?



Prediction

- There are `predict()` methods for `gam` type objects, just like `lm`.

```
ages <- 18:80
pred <-
  predict(fit4,
         newdata = data.frame(age = ages),
         se.fit = T)
plot(wage ~ age, data = Wage)
lines(ages, pred$fit, lwd = 2, col = colors[4])
lines(ages, pred$fit + qnorm(0.005) *
        pred$se.fit, lwd = 2,
      col = colors[4], lty = 2)
lines(ages, pred$fit + qnorm(0.995) *
        pred$se.fit, lwd = 2,
      col = colors[4], lty = 2)
```



(5) What confidence level is depicted here?

Putting the “Generalized” in GAMs

Binary response variable: Return of the otters!

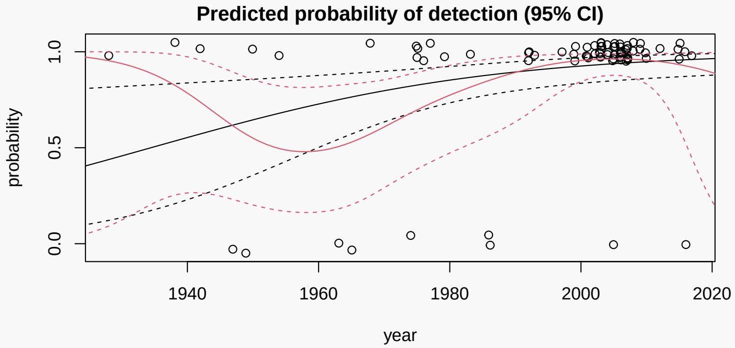
```
otters <- read.csv("../../data/River_Otters_-_High_Mountain_Lakes_[ds813].csv")
otters$Detected <- otters$Otters_Found > 0
otters$Region <- as.factor(otters$Region)
otters$Waterbody <- as.factor(otters$Waterbody)
head(otters, 3)
##           X       Y OBJECTID   Region Site_Name Waterbody Elevation_m Timeframe
## 1 -13501910 4947965        1 Cascades     Butte      Lake        1844    August
## 2 -13501910 4947965        2 Cascades     Butte      Lake        1844    August
## 3 -13518222 4939619        3 Cascades    Dersch     Marsh        2012    August
##   Year Otters_Found Rank       Source      Lat      Long   UTME    UTMN
## 1 2006            1    2 LVNP Records 40.56210 -121.2897 644789 4491553
## 2 2006            1    2 LVNP Records 40.56210 -121.2897 644789 4491553
## 3 2006            1    2 LVNP Records 40.50511 -121.4362 632496 4484997
##               DATUM Detected
## 1 UTM NAD83 Zone 10     TRUE
## 2 UTM NAD83 Zone 10     TRUE
## 3 UTM NAD83 Zone 10     TRUE
```

Binary response variable: Return of the otters!

Recall from Rlab 5 that we first looked at the isolated effect of year on detection.

```
fit_year_glm <- glm(Detected ~ Year, data = otters, family = binomial)
fit_year_gam <- gam(Detected ~ s(Year, k = 7), data = otters, family = binomial)
years <- 1900:2022
pred_year_glm <- predict(fit_year_glm, newdata = data.frame(Year = years), se.fit = T)
pred_year_gam <- predict(fit_year_gam, newdata = data.frame(Year = years), se.fit = T)
```

(6) What transformation needs to be done to the predictions to make this plot? Which predictions seem more "correct" to you?



Multiple predictors

Adding linear effects

- We will often be interested in including other predictor variables.
- These can have traditional linear effects. . .
- . . . or they can also have more flexible functional relationships.

```
fit_glm <- glm(Detected ~ Elevation_m + Waterbody + Year + Region,
               data = otters, family = binomial())
fit_gam <- gam(Detected ~ Elevation_m + Waterbody + s(Year, k = 7) + Region,
               data = otters, family = binomial())
```

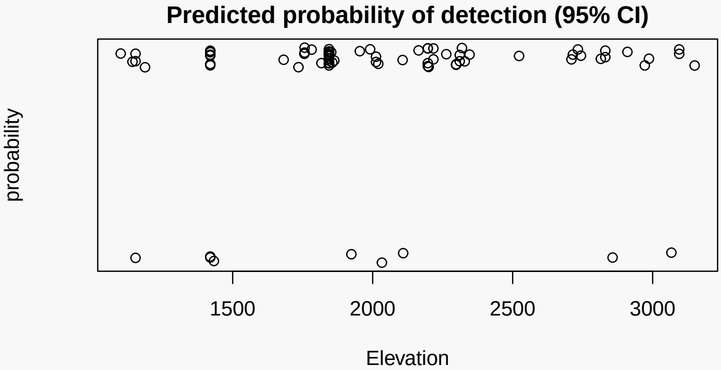
(7) Of the other predictors, which one might we consider for a non-linear relation using splines?

Adding non-linear effects

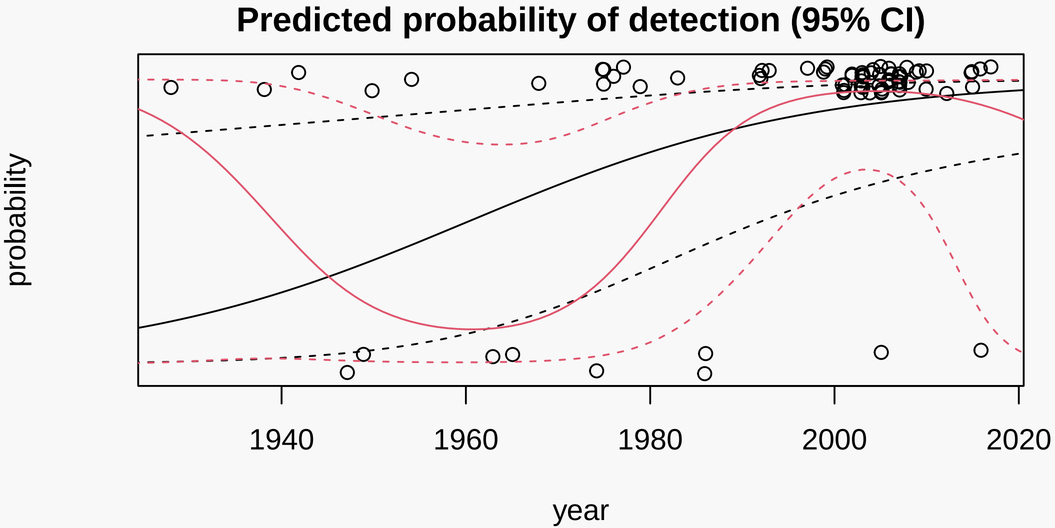
```
fit_alt <- gam(Detected ~ s(Elevation_m, k = 7) + Waterbody + s(Year, k = 7) + Region,
               data = otters, family = binomial())
summary(fit_alt)
##
## Family: binomial
## Link function: logit
##
## Formula:
## Detected ~ s(Elevation_m, k = 7) + Waterbody + s(Year, k = 7) +
##     Region
##
## Parametric coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)         5.236e+01  1.628e+07   0.000    1.000
## WaterbodyMarsh      2.018e-01  6.905e+07   0.000    1.000
## WaterbodyReservoir  7.919e-01  1.923e+00   0.412    0.681
## WaterbodyStream     5.277e+01  3.355e+07   0.000    1.000
## RegionKlamath      -5.022e+01  1.628e+07   0.000    1.000
## RegionSierra       -5.019e+01  1.628e+07   0.000    1.000
##
```

```
## Approximate significance of smooth terms:
##                    edf Ref.df Chi.sq p-value
## s(Elevation_m) 1.000  1.000  2.130  0.1444
## s(Year)        3.356  4.149  9.855  0.0469 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.337   Deviance explained = 44.6%
## UBRE = -0.35778  Scale est. = 1         n = 81
```

(8) We can interpret the p-value to mean there is weak evidence for a non-linear effect of elevation on the probability of detection. What do you see in the figure that confirms this result?



Multiple predictors



(9) Which color curves represent the GAM? How can you tell?