
Lecture 12: Polynomials and Splines

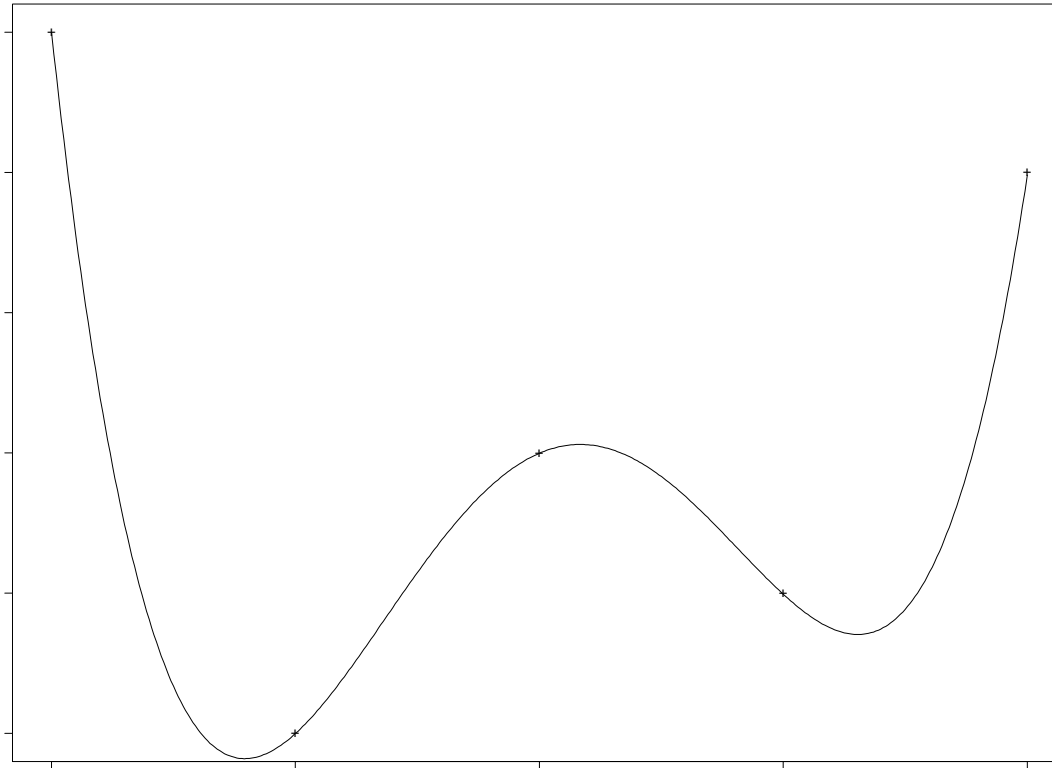
Reading Assignment:

- Muller and Fetterman, Chapter 9: “Polynomial Regression”

Motivation

Response-predictor relationships are often not linear. In most cases, we want to use tests and models with maximum power to detect association between a predictor and a response. The tests and models that we have discussed so far assume a linear relationship between $E(\mathbf{y})$ and \mathbf{x} , and power for detecting an association between \mathbf{y} and \mathbf{x} will be reduced when this is not true. (A linear model may have good power if the trend is nonlinear but still monotone, but power will likely be terrible for U-shaped or umbrella-shaped relationships.)

A simple way to capture non-linear relationships between an exposure and response of interest is to add polynomial terms to a linear model. With d distinct (x, y) pairs, a polynomial of order at most $d - 1$ will pass exactly through all the points. Consider the following example, in which $\mathbf{x} = (1, 2, 3, 4, 5)'$ and $\mathbf{y} = (5, 0, 2, 1, 4)'$.



Why wouldn't we always want to use this type of model, which perfectly fits the data?

Polynomial Models

	Order	Regression Model	Shape
0th	Zero	$y_i = \beta_0 + \varepsilon_i$	Horizontal Line
1st	Linear	$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$	Line
2nd	Quadratic	$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i$	Parabola
3rd	Cubic	$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i$	Ogive etc.

The number of points of inflection is one fewer than the number of monotone pieces of the curve, which is the order (degree) of the polynomial. The maximum order that can be fitted is one less than the number of distinct X values, but one should be able to defend the use of any model with higher than a 3rd order polynomial due to difficulties in interpretation.

We will use the GLM for estimation of parameters, but how?

Example: Romanesque Cathedrals in England

Consider the following data collected by Stephen J. Gould on the dimensions of medieval English cathedrals. (The Romanesque period in English architecture roughly dates to 1066-1180.)

Cathedral	Nave Height (ft)	Total Length (ft)
Durham	75	502
Canterbury	80	522
Gloucester	68	425
Hereford	64	344
Norwich	83	407
Peterborough	80	451
St. Albans	70	551
Winchester	76	530
Ely	74	547

The nave height is the height of the center of the church, as seen (for Canterbury Cathedral) on the following page.



We can fit a polynomial model, say a third order polynomial, to the cathedral data, treating length as the dependent variable and height as the independent variable. The \mathbf{X} matrix is given by

$$\mathbf{X} = \begin{bmatrix} 1 & 75 & 75^2 & 75^3 \\ 1 & 80 & 80^2 & 80^3 \\ 1 & 68 & 68^2 & 68^3 \\ 1 & 64 & 64^2 & 64^3 \\ 1 & 83 & 83^2 & 83^3 \\ 1 & 80 & 80^2 & 80^3 \\ 1 & 70 & 70^2 & 70^3 \\ 1 & 76 & 76^2 & 76^3 \\ 1 & 74 & 74^2 & 74^3 \end{bmatrix}.$$

Variable added-last tests for each term are not recommended for

model selection. (Why?)

To test association (i.e., is x related to y ?) when linearity is not assumed, we will conduct a *groupwise* test of all terms involving the predictor.

We fit a cubic model to the cathedral data below.

```
data romanesque; set romanesque;
heightsq=height*height; heightcu=heightsq*height;
run;
proc reg data=romanesque;
model length=height heightsq heightcu/tol vif ss1;
output out=out pred=pred;
run;
```

The REG Procedure

Model: MODEL1

Dependent Variable: length

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	34106	11369	7.41	0.0275
Error	5	7676.04705	1535.20941		
Corrected Total	8	41782			

Root MSE	39.18175	R-Square	0.8163
Dependent Mean	475.44444	Adj R-Sq	0.7061

Coeff Var 8.24108

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	-14512	30517	-0.48	0.6544
height	1	478.38312	1253.74118	0.38	0.7185
heightsq	1	-4.69654	17.10590	-0.27	0.7946
heightcu	1	0.01324	0.07752	0.17	0.8711

Parameter Estimates

Variable	DF	Type I SS	Tolerance	Variance Inflation
Intercept	1	2034427	.	0
height	1	3035.20888	0.00000317	315582
heightsq	1	31026	7.844759E-7	1274736
heightcu	1	44.79835	0.00000310	323080

We consider three basic inferential questions.

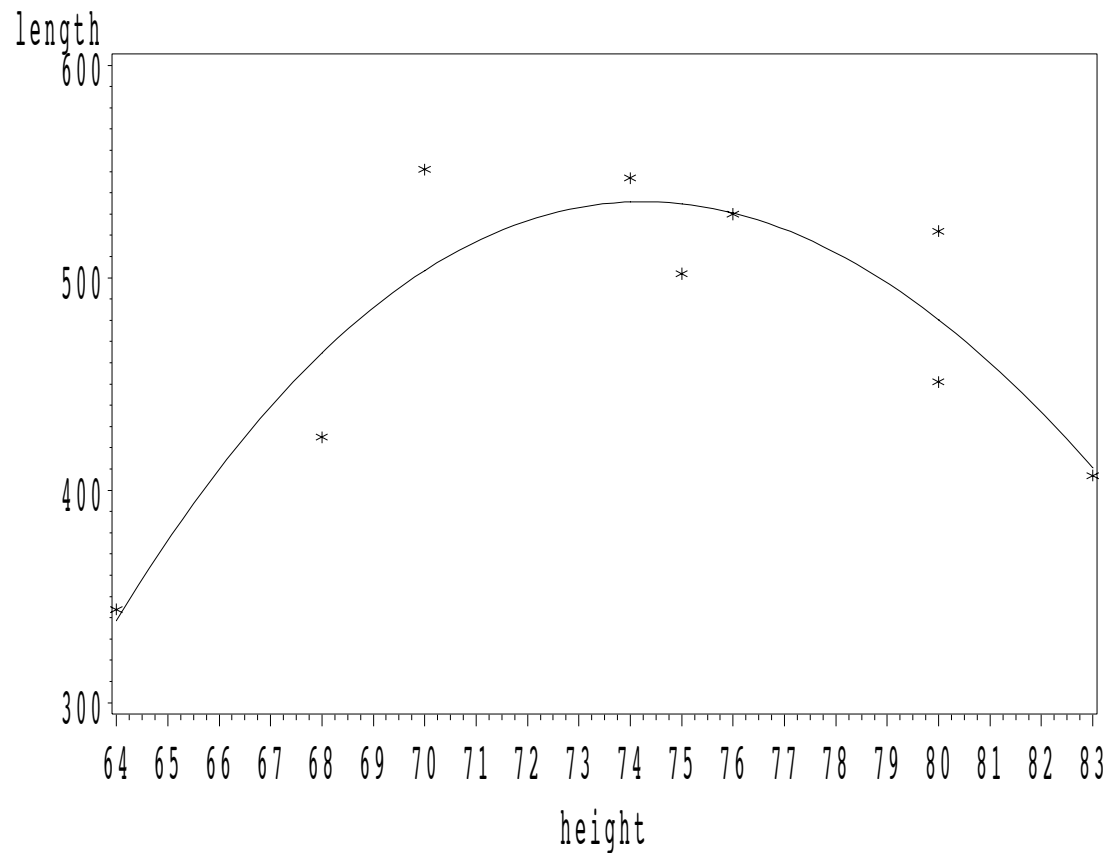
1. Are height and length related?
2. Is a cubic model significantly better than the quadratic model? If not, is a quadratic model significantly better than a linear model?
3. Should we have considered even higher-order polynomial terms?

How do you interpret the tolerance and VIF values?

Tolerance is a measure of collinearity, which equals to $1-R^2$ of predicting this covariate with all the other covariates. $VIF = 1/\text{tolerance}$.

The following code may be used to obtain a plot of observed and predicted length values for the cubic model.

```
symbol1 color=black i=none v=star;  
symbol2 color=black v=point line=1 i=RC;  
/* RL for linear, RQ for quadratic */  
proc gplot data=out;  
plot length*height=1 pred*height=2/overlay;  
run;
```



What do you think about the model fit based on the observed and predicted nave lengths? What do you think about inferences outside the range of the data?

Limitations of Natural Polynomials

Fitting higher-order natural polynomial models can be dangerous! Because independent variables in a polynomial model are functions of the same basic variable x , they are correlated, and computational difficulties due to collinearity may result. In fact, collinearity is almost certainly present if the order of the polynomial is high. Techniques like centering and the use of orthogonal polynomials will help, though there is no remedy for difficulty of interpretation for high-order polynomials in most settings.

With a polynomial model, extrapolation beyond the range of the data is even more dangerous than usual.

Orthogonal Polynomials

Orthogonal polynomials provide one good solution to numerical problems in a polynomial regression by using an alternate coding scheme. Natural polynomials yield the model

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i,$$

while orthogonal polynomials yield the model

$$y_i = \gamma_0 + \gamma_1 z_{i1} + \gamma_2 z_{i2} + \gamma_3 z_{i3} + \delta_i.$$

The orthogonal polynomial scores ($\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3$) are the solution to a system of equations under the constraints that the new predictors ($\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3$)

1. contain the same information as the original variables ($\mathbf{x}, \mathbf{x}^2, \mathbf{x}^3$),
2. are linear combinations of original data,
3. are mean zero (except for the intercept),

-
4. are mutually orthogonal (uncorrelated), and
 5. \mathbf{z}_1 captures all information in \mathbf{x} beyond location of response (intercept), \mathbf{z}_2 captures all information in \mathbf{x}^2 , beyond linear term in \mathbf{x} and the intercept, etc.

Finding the values of $\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3$ corresponds to solving a system of simultaneous equations for each subject:

$$\begin{aligned} z_{i1} &= a_{01} + a_{11}x_i \\ z_{i2} &= a_{02} + a_{12}x_i + a_{22}x_i^2 \\ &\vdots \\ z_{ip-1} &= a_{0p-1} + \sum_{j=1}^{p-1} a_{jp-1}(x_i)^j . \end{aligned}$$

Clearly, the new variables are linear combinations of the original ones. We may write

$$\begin{aligned}
 x_i &= b_{01} + b_{11}z_{i1} \\
 x_i^2 &= b_{02} + b_{12}z_{i1} + b_{22}z_{i2} \\
 &\vdots \\
 x_i^{p-1} &= b_{0p-1} + \sum_{j=1}^{p-1} b_{jp-1}z_{ij}.
 \end{aligned}$$

Thus we may (without losing any information) write either

$$y_i = \beta_0 + \beta_1 x_i + \dots + \beta_{p-1} x_i^{p-1} + \varepsilon_i$$

or

$$y_i = \alpha_0 + \alpha_1 z_{i1} + \dots + \alpha_{p-1} z_{ip-1} + \varepsilon_i.$$

The parameters β and α will not be numerically identical, but the overall F tests from these models are exactly the same. In addition, testing $H_0 : \alpha_k = 0$ for the orthogonal polynomial model is equivalent to an added-in-order test of $H_0 : \beta_k = 0$ in the natural polynomial

model.

If the predictor values are equally spaced and there are an equal number of observations at each of d values, finding the orthogonal predictors reduces to finding $d - 1$ vectors, each of length d . The $d - 1$ vectors are known as the orthogonal polynomial coefficients. The table in Appendix B.10 of Muller and Fetterman may be used to obtain coefficients if predictors are equally spaced and each predictor value has the same number of replicates.

Other Ways to Improve Fit of Natural Polynomial Models

- Scaling: For example, if a weight variable is measured in grams with a range of 1000-2000, use kg with a range of 1-2. (Presence of the intercept leads to preferring a range of 1-10 to avoid collinearity.)
- Centering: Often dramatically reduces collinearity (always eliminates any with intercept) and is strongly recommended as habitual technique.
- Pseudo-Centering: Some “nice” number often provides most of the numerical advantage of centering, while simplifying interpretation. So if a predictor is weight or blood pressure, we might pseudo-center it at a “healthy” value of the predictor (average in a healthy population say).

Model Selection in Polynomial Regression

Muller and Fetterman prefer *backwards selection*, starting at the largest desirable polynomial model (they recommend starting at a cubic) and evaluating smaller ones with added-in-order tests. They recommend stopping at the highest order polynomial that is significant and including all lower order terms. Model diagnostics are very important in polynomial models as in all other regression models.

More Flexible Models

Often, y does not behave linearly in all the predictors. The simplest way to describe a nonlinear effect of x is to include polynomial terms in the model. However, nonlinear effects may not follow polynomial forms. In such cases, transformations might be able to induce linearity, but often the transformation is not known or does not exist.

Linear Splines

Spline functions are piecewise polynomials used to fit curves. Within intervals of x , they are polynomials, and they are connected across the different intervals. The most simple type of spline function is a *linear spline function*, which is a piecewise linear function. Linear spline functions may also be fit in the framework of the GLM.

Divide the x axis into intervals with endpoints k_1, k_2, \dots, k_K , called *knots*. The linear spline function is given by

$$\begin{aligned} E(\mathbf{y}) = & \beta_0 + \beta_1 \mathbf{x} + \beta_2 (\mathbf{x} - k_1 \mathbf{J}_n)_+ + \beta_3 (\mathbf{x} - k_2 \mathbf{J}_n)_+ \\ & + \dots + \beta_{K+1} (\mathbf{x} - k_K \mathbf{J}_n)_+, \end{aligned}$$

where

$$(u)_+ = \begin{cases} u & u > 0 \\ 0 & u \leq 0. \end{cases}$$

The number of knots varies depending on the amount of data available.

We can rewrite the linear spline function as follows:

$$E(y_i) = \begin{cases} \beta_0 + \beta_1 x_i & x_i \leq k_1 \\ \beta_0 + \beta_1 x_i + \beta_2(x_i - k_1) & k_1 < x_i \leq k_2 \\ \beta_0 + \beta_1 x_i + \beta_2(x_i - k_1) + \beta_3(x_i - k_2) & k_2 < x_i \leq k_3 \\ \vdots & \\ \beta_0 + \beta_1 x_i + \beta_2(x_i - k_1) + \dots + \beta_{K+1}(x_i - k_K) & k_K < x_i. \end{cases}$$

We can fit this model in SAS by creating new variables $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K)$ where

$$\begin{aligned} \mathbf{x}_1 &= \mathbf{x} \\ \mathbf{x}_2 &= (\mathbf{x} - k_1)_+ \\ &\vdots \\ \mathbf{x}_K &= (\mathbf{x} - k_K)_+. \end{aligned}$$

Then, we can fit the model

$$E(\mathbf{y}) = \beta_0 + \beta_1 \mathbf{x}_1 + \beta_2 \mathbf{x}_2 + \dots + \beta_{K+1} \mathbf{x}_K$$

in any standard software package.

To test linearity in x , we simply test the hypothesis

$$H_0 : \beta_2 = \beta_3 = \dots = \beta_{K+1} = 0.$$

To construct a linear spline with knots at $k_1 = 70$ and $k_2 = 77$ for the cathedral data, we use the following \mathbf{X} :

$$\mathbf{X} = \begin{bmatrix} 1 & 75 & 5 & 0 \\ 1 & 80 & 10 & 3 \\ 1 & 68 & 0 & 0 \\ 1 & 64 & 0 & 0 \\ 1 & 83 & 13 & 6 \\ 1 & 80 & 10 & 3 \\ 1 & 70 & 0 & 0 \\ 1 & 76 & 6 & 0 \\ 1 & 74 & 4 & 0 \end{bmatrix} .$$

Then we can fit the model using the following SAS code.

```
data romanesque; set romanesque;
ht_a=0;
ht_b=0;
if height>70 then ht_a=height-70;
if height>77 then ht_b=height-77;
run;
```

```
proc reg;
model length=height ht_a ht_b;
run;
```

```
*****
```

The REG Procedure
 Model: MODEL1
 Dependent Variable: length

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	35677	11892	9.74	0.0157
Error	5	6105.05660	1221.01132		
Corrected Total	8	41782			

Root MSE	34.94297	R-Square	0.8539
Dependent Mean	475.44444	Adj R-Sq	0.7662
Coeff Var	7.34954		

Parameter Estimates

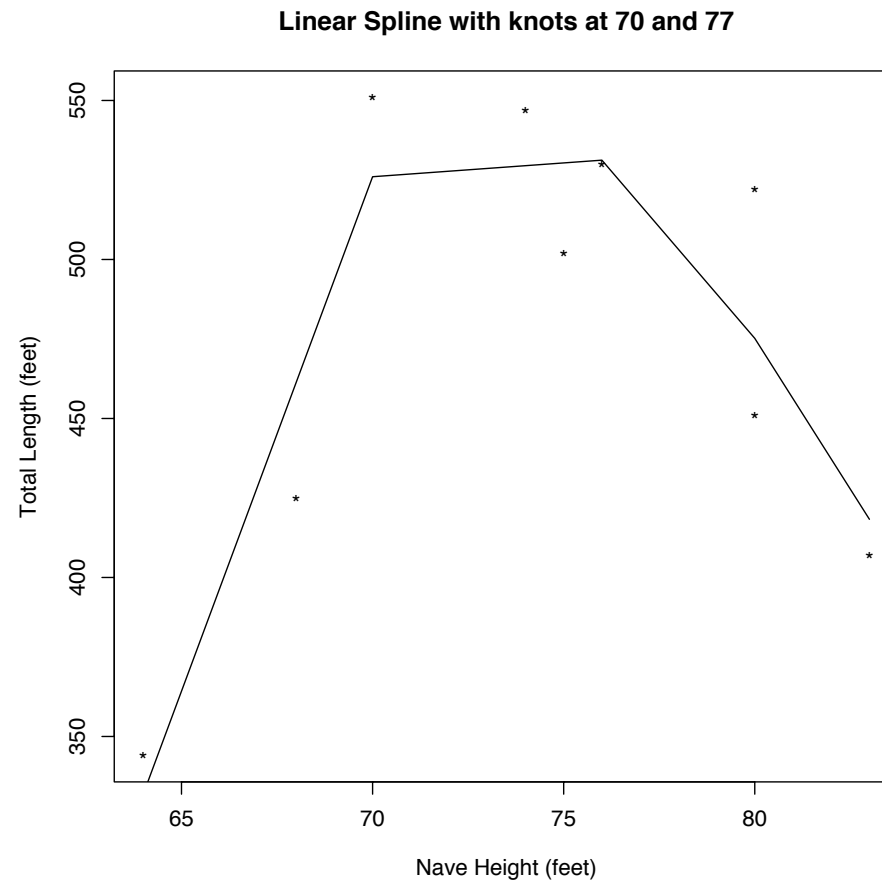
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	-1738.74528	534.64321	-3.25	0.0226
height	1	32.35377	7.92328	4.08	0.0095

ht_a	1	-31.48585	12.78943	-2.46	0.0571
ht_b	1	-19.83333	12.62576	-1.57	0.1770

Parameter Estimates

Variable	DF	Type I SS
Intercept	1	2034427
height	1	3035.20888
ht_a	1	29629
ht_b	1	3012.97872

The plot of fitted values vs. height is below.



Test linearity using this model. How do you interpret the parameter estimates?

Cubic Spline Functions

Although linear splines are simple and are good approximations for some relationships, they are not smooth, and they will not fit curved functions very well. To do so, we must use piecewise polynomials of higher orders. Cubic polynomials have nice properties and do a good job of fitting sharp curves. To make cubic splines smooth at the knots, we force the first and second derivatives of the function to agree at the knots.

Spline Functions in SAS

PROC GAM (generalized additive model) in SAS allows us to fit spline functions for covariates. These splines are slightly different from cubic spline functions but are based on the same ideas that we have discussed.

Categorization of Predictors

In epidemiology, categorizing continuous predictors is almost a default practice. The thought is that creating categories of exposure (or dichotomizing exposure into “exposed” and “unexposed” groups) avoids assumptions of linearity. Although categorization does make interpretation simple, it can make unnatural assumptions about the exposure effect and may also lead to power losses.

For example, suppose that we decide to divide churches into three groups: short (≤ 70 feet), medium (> 70 and ≤ 77 feet), and tall (> 77 feet). We can then define two variables to indicate whether churches are medium or tall:

$$\begin{aligned} \text{medium}_i &= \begin{cases} 1 & 70 < \text{height} \leq 77 \\ 0 & \text{otherwise} \end{cases} \\ \text{tall}_i &= \begin{cases} 1 & \text{height} > 77 \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

We can then fit the model

$$E(\text{length}) = \beta_0 + \beta_1 \text{medium}_i + \beta_2 \text{tall}_i + \varepsilon_i,$$

$i = 1, \dots, n$. In this model, we have

$$\begin{aligned} E(\text{length}|\text{short nave}) &= \beta_0 \\ E(\text{length}|\text{medium nave}) &= \beta_0 + \beta_1 \\ E(\text{length}|\text{tall nave}) &= \beta_0 + \beta_2. \end{aligned}$$

What does this model assume about the effect of increasing nave height among churches with short naves?

Below are SAS code and output from fitting this model.

```
data romanesque; set romanesque;
medium=1; tall=0;
if height<70.1 then medium=0;
if height>77 then medium=0;
if height>77 then tall=1;
run;
```

```
proc reg;
model length=medium tall/ss1;
run;
```

```
*****
```

The REG Procedure

Model: MODEL1

Dependent Variable: length

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	2	12254	6126.77778	1.24	0.3530
Error	6	29529	4921.44444		
Corrected Total	8	41782			

Root MSE	70.15301	R-Square	0.2933
Dependent Mean	475.44444	Adj R-Sq	0.0577
Coeff Var	14.75525		

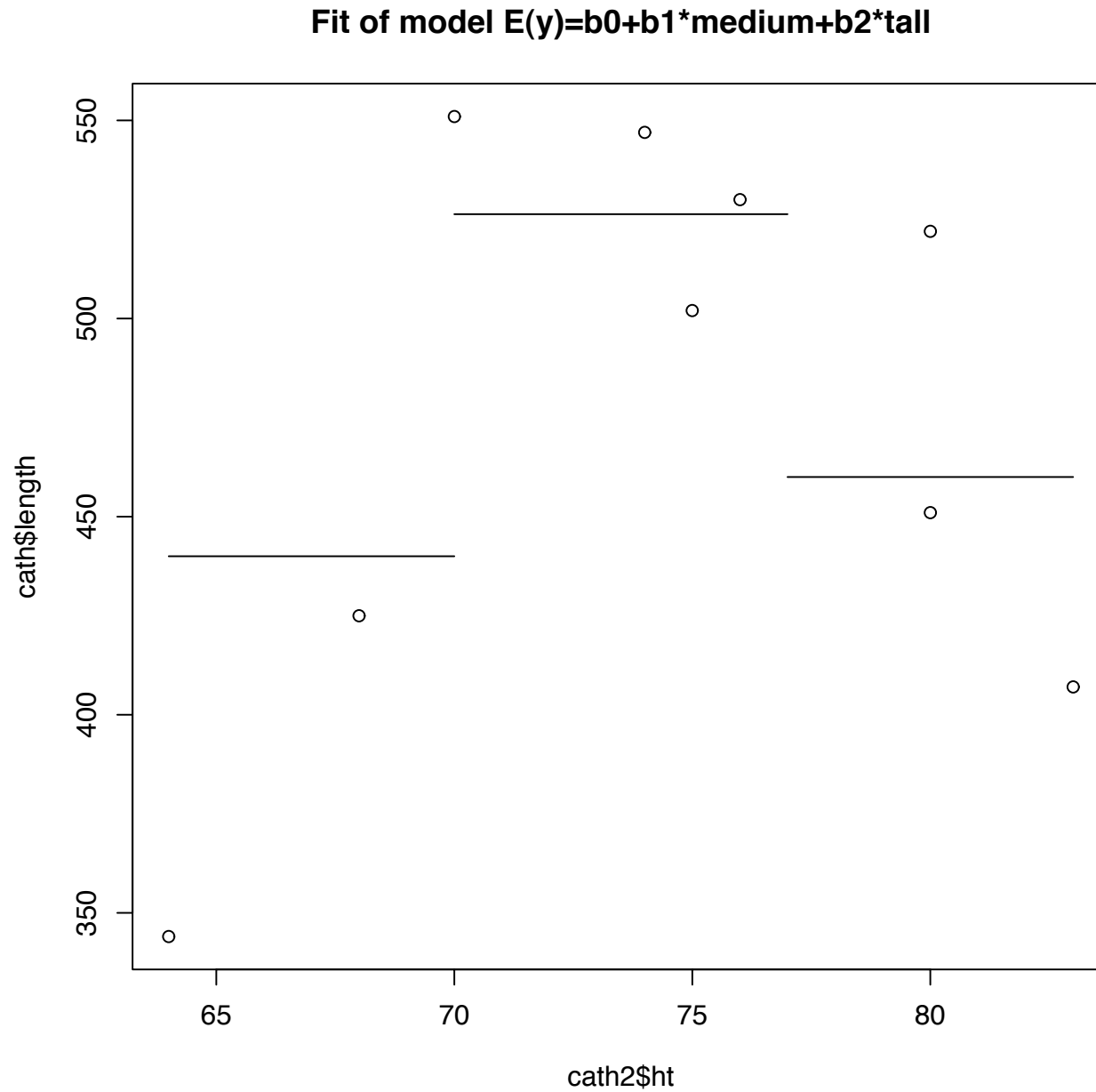
Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	440.00000	40.50286	10.86	<.0001
medium	1	86.33333	57.27969	1.51	0.1825
tall	1	20.00000	57.27969	0.35	0.7389

Parameter Estimates

Variable	DF	Type I SS
Intercept	1	2034427
medium	1	11654
tall	1	600.00000

A plot of the model fit is provided.



Test association in this model. In addition, give proper interpretations of all parameter estimates, and provide estimated conditional means for cathedrals with nave heights of 69.9 and 70.1 feet, respectively.

Nonparametric Regression

Nonparametric smoothers are tools that help determine the shape of the relationship between variables. These tools work best when interest is in one continuous predictor and one continuous response at a time.

Moving Average

The most simple nonparametric smoother is a *moving average*.

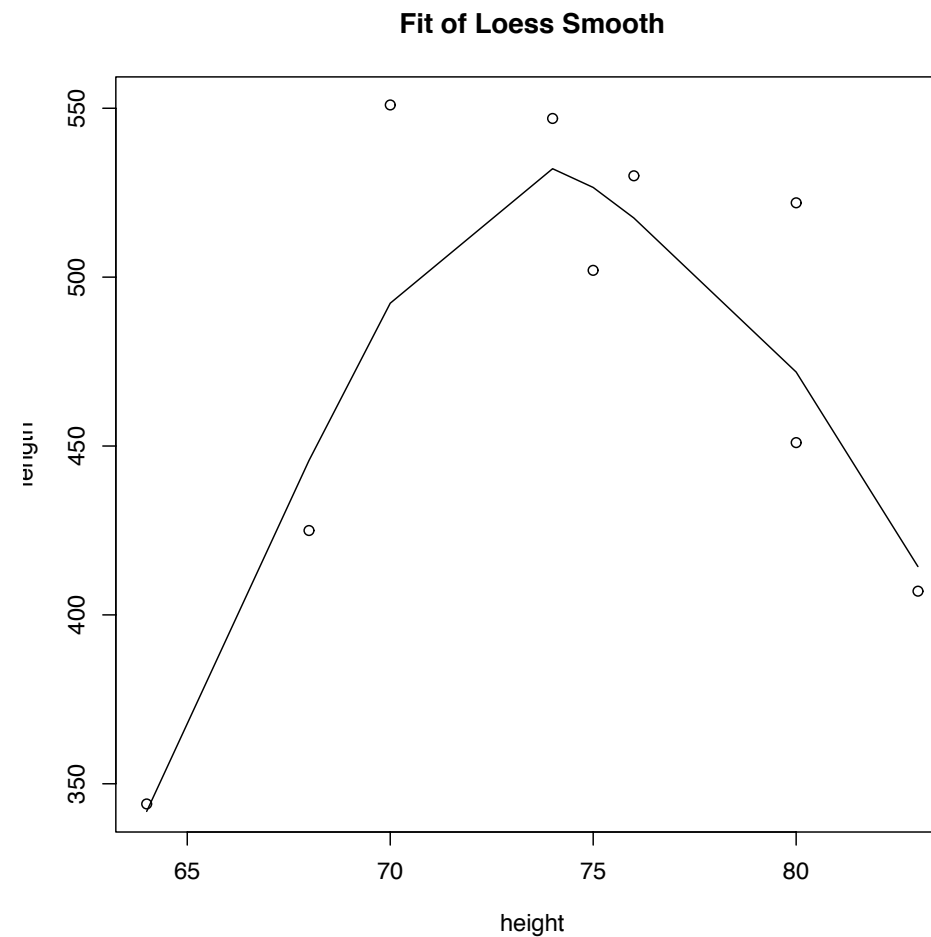
Suppose our data are $\mathbf{x} = (1, 2, 3, 5, 8)'$ and $\mathbf{y} = (2.1, 3.8, 5.7, 11.1, 17.2)'$. To smooth their relationship, we could estimate $E(Y|X = 2)$ by $\frac{2.1+3.8+5.7}{3}$. This is problematic at the upper and lower limits X , and estimates are sensitive to the interval width, which here is 3 data points. Moving average smoothers are also called moving flat line smoothers.

Loess

A moving least squares linear regression smoother is superior to the moving average approach. The smoother *loess* is the most popular smoother of this type. To get smoothed values of $E(Y)$ at $X = x$, we take all data with X values within a suitable interval (often $\frac{2}{3}$ of data points) about x , and then we fit a linear regression using only these points. The predicted value from this regression at $X = x$ will be used as our estimate of $E(Y|X = x)$. Then, we move to the next observed value of X and repeat the process.

Actually, *weighted least squares* estimates are typically used so that points closest to $X = x$ are given the most weight, and points further away are given less weight. For this reason, *loess* is called a locally weighted least squares method. Loess calculates a smoothed value for $E(Y)$ at each observed value of X , and estimates for other X 's are obtained by interpolation.

Next is a plot of the model fit using nonparametric regression with a loess smoother for nave height. Like splines, loess smoothers may be fit in PROC GAM.



Summary

Relationships between a response and predictor are often not linear. Nonlinear effects can be accommodated in the GLM framework by using

- polynomial terms (orthogonal polynomials are more stable than natural polynomials),
- transformations (often not known in advance),
- simple splines, or
- categorization of predictors (power loss; interpretation difficulties).

Smoothing splines and loess smoothers are useful in determining shapes of relationships and may be the best choice for presenting analysis results if relationships are highly nonlinear or when thresholds are of interest. However, such smoothers are often not the most powerful approach when relationships are monotone or can be well-represented by lower-order polynomials.

Next: Transformations

Reading Assignment:

- Muller and Fetterman, Chapter 10: “Transformations”
- Weisberg, Chapter 8: “Transformations”