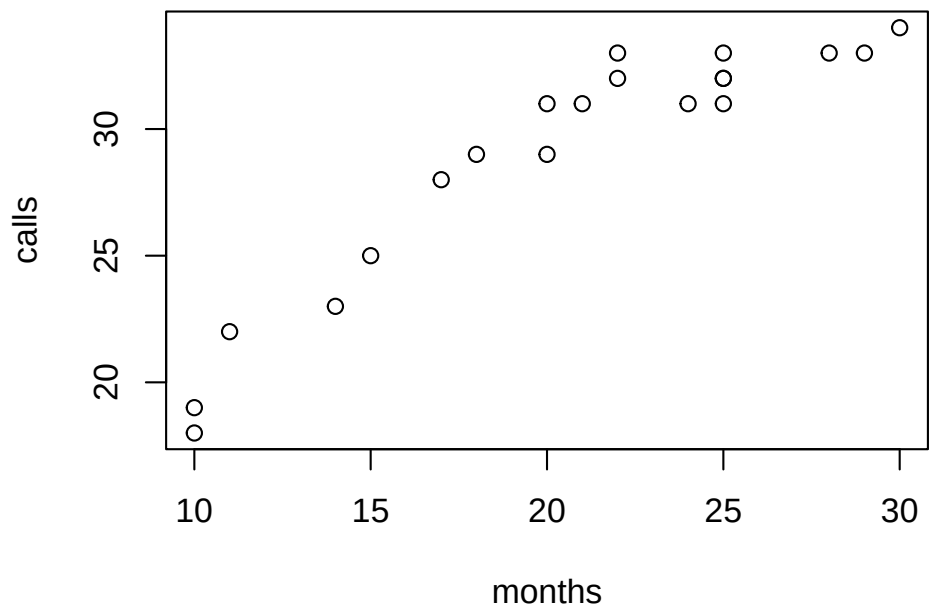
Nonlinearity in regression models

Jared S. Murray
The University of Texas at Austin
McCombs School of Business

Non Linearity

Example: *Telemarketing*

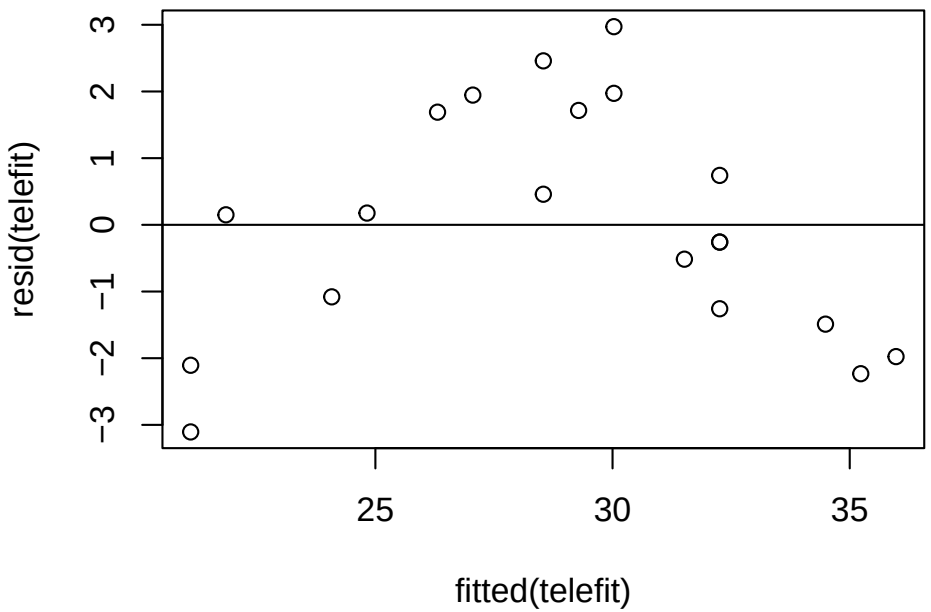
- How does length of employment affect productivity (number of calls per day)?



Non Linearity

Example: *Telemarketing*

- Residual plot highlights the non-linearity



Non Linearity

What can we do to fix this? We can use multiple regression and transform our X to create a nonlinear model...

Let's try

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$$

The data...

months	months2	calls
10	100	18
10	100	19
11	121	22
14	196	23
15	225	25
...	...	...

Telemarketing: Adding a squared term

In R, the one way to add a quadratic term (or other transformation) is using `I()` in the formula:

```
telefit2 = lm(calls~months + I(months^2), data=tele)
print(telefit2)

##
## Call:
## lm(formula = calls ~ months + I(months^2), data = tele)
##
## Coefficients:
## (Intercept)       months  I(months^2)
##    -0.14047      2.31020     -0.04012
```

Telemarketing: Adding a squared term

Another convenient way is to use `poly`:

```
telefit3 = lm(calls~poly(months, 2), data=tele)
print(telefit3)

##
## Call:
## lm(formula = calls ~ poly(months, 2), data = tele)
##
## Coefficients:
##      (Intercept)  poly(months, 2)1  poly(months, 2)2
##           28.950            19.936            -6.356
```

Telemarketing: Adding a squared term

Notice the difference in the coefficients? However, the fits are identical:

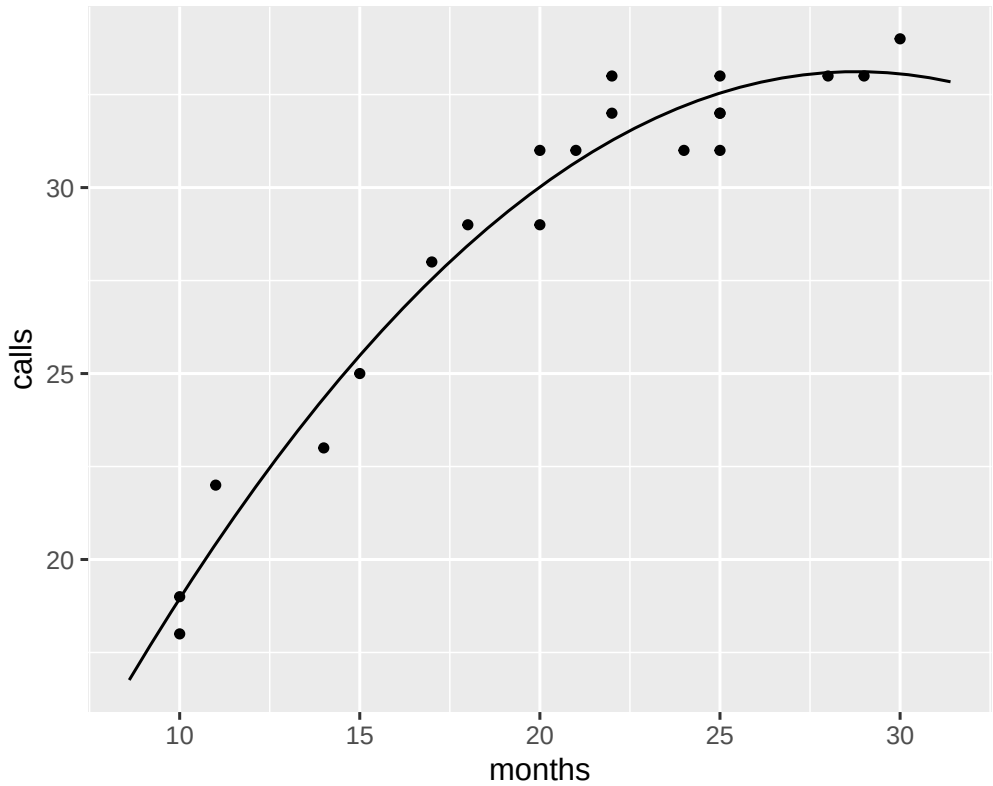
```
cor(predict(telefit2), predict(telefit3))

## [1] 1
```

`poly` *orthogonalizes* the linear and quadratic terms to eliminate the sample correlation between them, stabilizing their coefficient estimates

Using `I` will give you more interpretable coefficients.

Telemarketing

$$\hat{y}_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2$$



Telemarketing

What is the marginal effect of X on Y?

$$\frac{\partial \mathbb{E}[Y|X]}{\partial X} = \beta_1 + 2\beta_2 X$$

- To better understand the impact of changes in X on Y you should evaluate different scenarios.
- Moving from 10 to 11 months of employment raises average productivity by 1.47 calls
- Going from 25 to 26 months only raises the average number of calls by 0.27.
- This is similar to the effect of variable interactions we saw earlier – the effect of a change in X depends on what the original value of X was, instead of some other variable.

Polynomial Regression

We could keep going beyond X^2...

In general, we can add powers of X to get polynomial regression:

$Y = \beta_0 + \beta_1 X + \beta_2 X^2 \ldots + \beta_m X^m$

You can fit basically any mean function if m is big enough.

But usually, $m = 2$ does the trick.

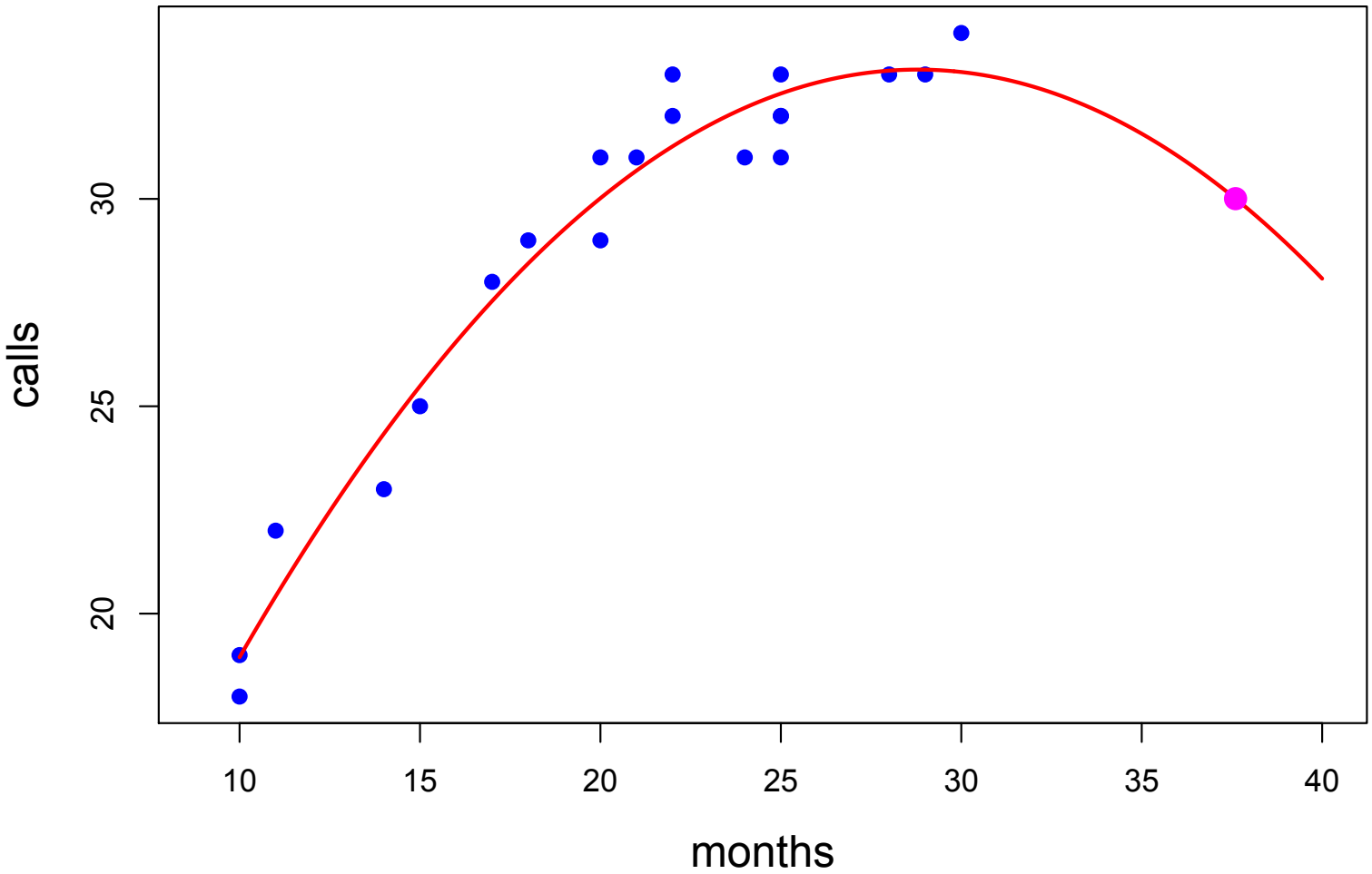
Closing Comments on Polynomials

We can always add higher powers (cubic, etc) if necessary.

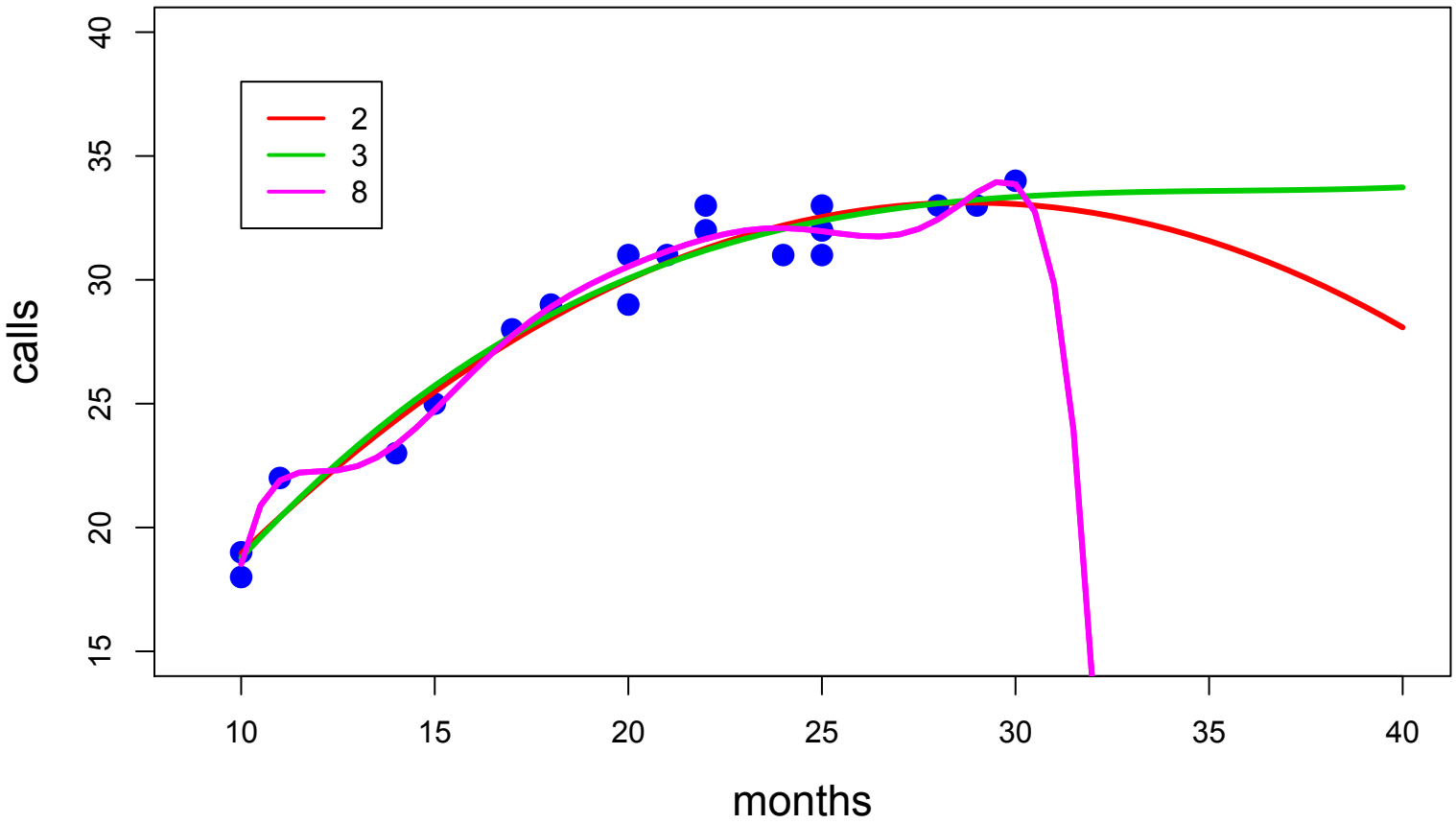
Be very careful about predicting outside the data range. The curve may do unintended things beyond the observed data.

Watch out for over-fitting... remember, simple models are "better".

Be careful when extrapolating...



...and, be careful when adding more polynomial terms!



Other nonlinear terms

Polynomials are a useful and interpretable way to get nonlinear effects, but there is nothing privileged about them over other nonlinear transformations of X

E.g., we've already seen log transforms of X are useful when expected changes in Y depend on *percentage* changes in X

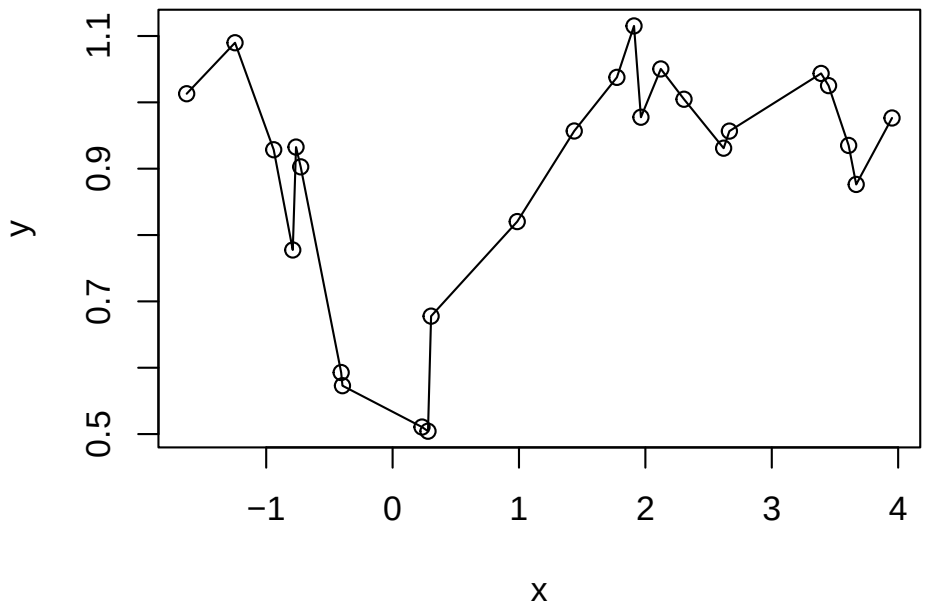
Smoothing splines are a generic way to incorporate nonlinear effects in regression models like

$$y_i = f(x_i) + \epsilon_i$$

Smoothing splines

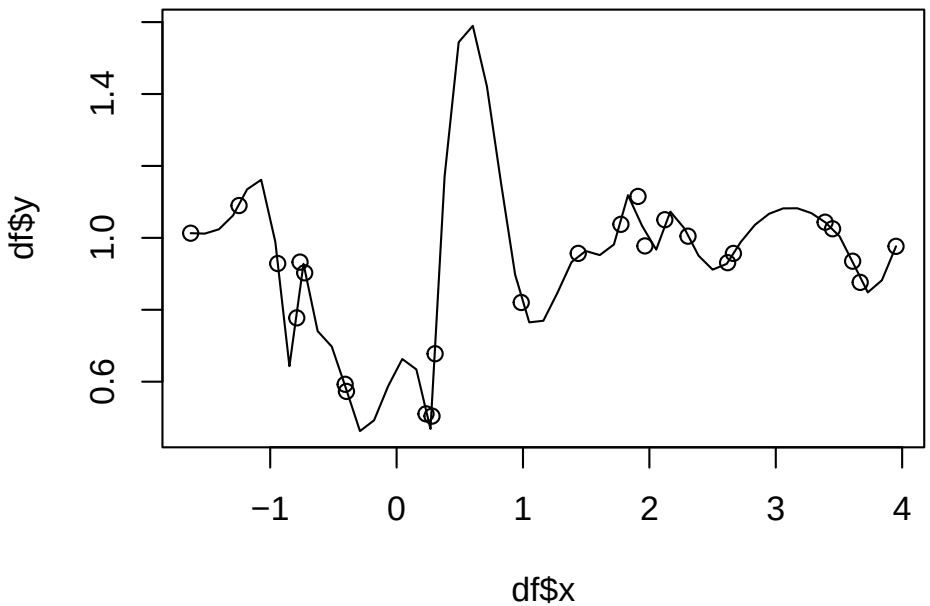
A *spline* is a *piecewise* polynomial constructed to interpolate between data points (with no "ties" in)

Linear spline:



Smoothing splines

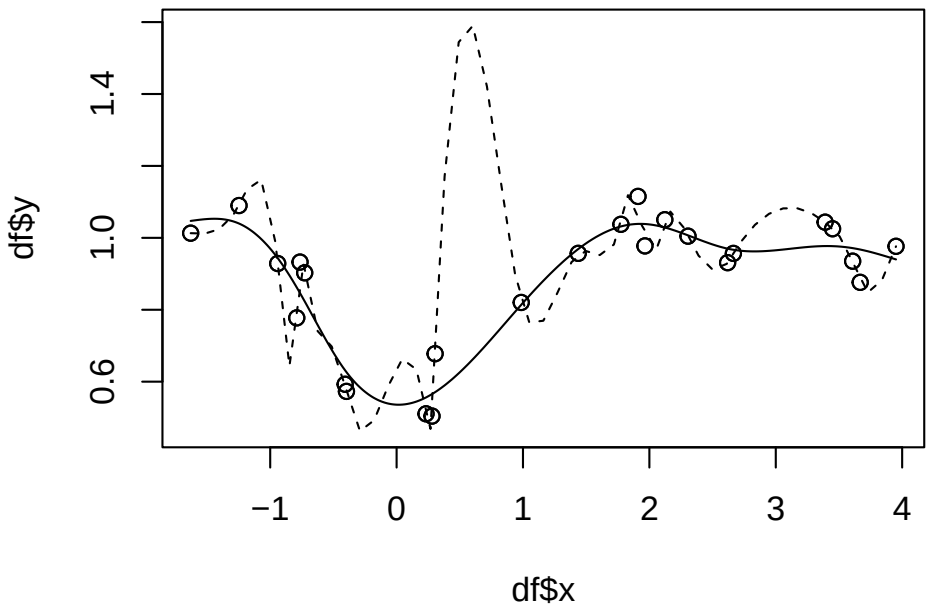
Cubic spline:



Smoothing splines

What if we don't want to interpolate exactly between points?
(And we don't!)

Trade off interpolation i.e., fit to the observed data in exchange for simpler functions:



Smoothing splines

How do we express the tradeoff between interpolation and complexity? One way: Find the function $\hat{f}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i - f\left(x_i\right)\right)^2 + \lambda \int \left(f''(x)\right)^2 dx$$

The first term is the MSE, the second penalizes "wiggly" functions.

The solution is always a cubic spline!

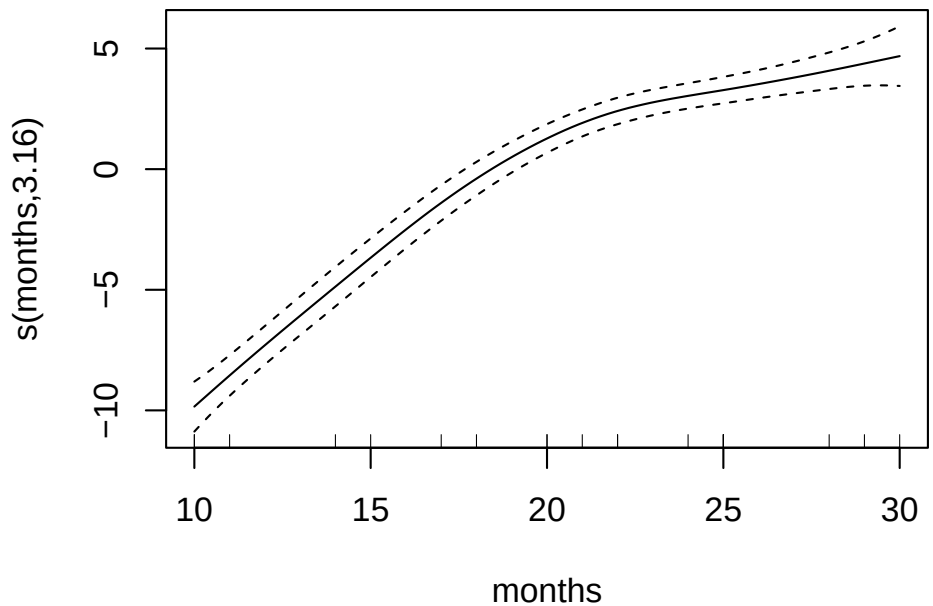
Choose a λ to optimize out of sample prediction errors, as usual.

Smoothing splines: Telemarketing

```
library(mgcv)
telefit3 = gam(calls~s(months), data=tele); summary(telefit3)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## calls ~ s(months)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  28.9500     0.2108   137.3   <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(months) 3.162  3.867 127.2  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.963   Deviance explained = 96.9%
## GCV = 1.1225  Scale est. = 0.88889   n = 20
```

Smoothing splines

```
mean(tele$calls)

## [1] 28.95

plot(telefit3)
```



Smoothing splines

We can generalize the univariate model to multiple $X's$ and get an *additive* model:

$$y_i = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \epsilon_i$$

We can use cubic splines for all the $f_j's$, or specify other forms (linear, etc)

NOTE: The *level* of each function is arbitrary, so they are constrained to average to zero: $\sum_{i=1}^{m} \hat{f}_j(x_j) = 0$. The *shape* is the interesting feature.

Interpreting additive models

In the model

$$y_i = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \epsilon_i$$

$f_j(d) - f_j(c)$ is the change in the predicted value for y when $X_j = d$ versus $X_j = c$, **holding the other variables constant**

The partial derivative of the predicted value in x_j is just $f'(x_j)$

Let's see an example...

Smoothing splines

What does this have to do with multiple regression? Representing f as a cubic spline means that

$$f(x) = \sum_{j=1}^{m+4} \beta_j g_j(x)$$

for some functions $g_1, g_2, \ldots, g_{m+4}$. This is like adding quadratic or cubic terms constructed from x, and using regression to find the coefficients.

Smoothing splines

What are the functions? They can be represented in different ways; one is

$$g_1(x) = 1, g_2(x) = x, \ldots g_{k+1}(x) = x^k$$
$$g_{k+1+j}(x) = (x - t_j)^k_{+,} \quad j = 1, \ldots m$$

for $k = 3$, where $(x - t_j)_{+,}$ is $\max(0, (x - t_j))$

Nonlinear models: Summary

The best nonlinear models are motivated by an understanding of the problem – this relationship should be quadratic, multiplicative, etc.

If a simple polynomial or log transform isn't motivated by the problem at hand, or won't do the trick to make your residual plots look reasonable, *additive* models using smoothing splines are usually a better alternative than high-degree polynomials