Generalized Additive Models

Largely adapted from Edwards chapter 17 & 18
and Zurr et al., 2019

Erin Winslow & Natalie Love

8/4/20

What is a GAM: generalized additive model

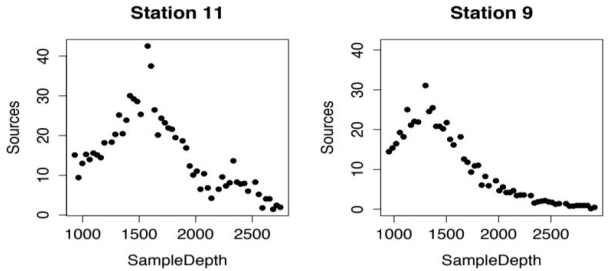
useful for modeling **nonlinear relationships**

Known for being data driven (as opposed to model driven): different from nonlinear least squares

- nls uses a particular curve the data is fit to
- GAMs model nonlinear relationships without having a specific formula relating x and y
    - Assume Y is some smooth function of X $\longrightarrow$ based on the data, what does that function look like?
        - Methods are **non-parametric** because no nonlinear function is specified in advance

Underlying assumptions: functions are additive and the components are smooth

Pros and cons of GAMs



Sources = estimated number of sources of bioluminescence per cubic meter.

Pros (can ask questions like)

- How does y depend on x without assuming a particular curve?
- Does relationship between x and y differ between groups of data (sites, years, etc)?
- What is the best relationship to control for effect of x (even if we don't care about the effect of x)?

Cons

- No interpretable equation for relationship between x and y (only plotting)
- Predicting is harder but not impossible
- Combining these models with other complexity (mixed models, etc) is tricky but not impossible

Why might I want to use a GAM?

(There are obviously lots of other reasons out there)

You have determined a linear regression model won't really work

- Residuals have a clear pattern when plotted
- You've already added explanatory variables and explored interaction terms, and still see patterns
- You do not want to transform your data

→ one alternative is to move onto smoothing models aka GAMs

Challenge of GAMs

Finding a smooth function that approximates the patterns in the data, but does not over or under fit the data

There are two choices to make when doing this

1. Which kind of smoother to use
2. How "wiggly" the smoother should be

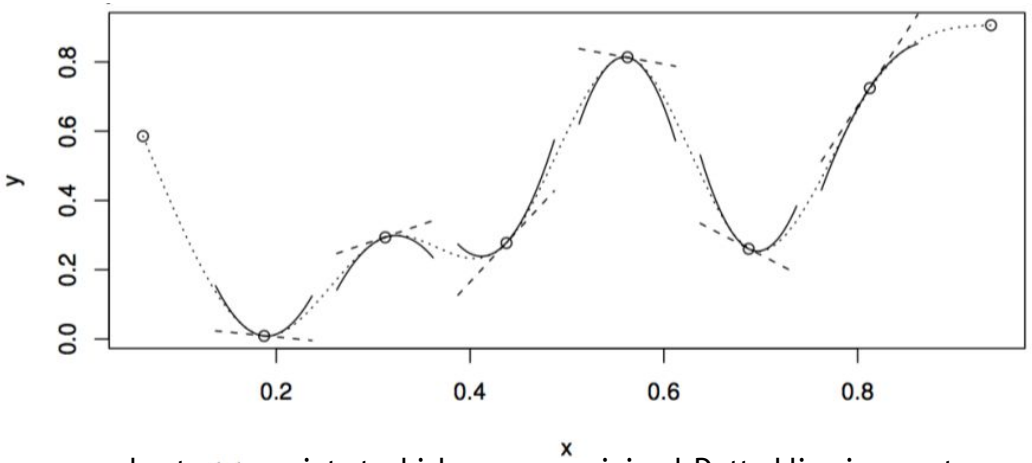
There are many smoothers to choose from, but typically results are very similar

Two packages, `gam` and `mgcv`

Cubic regression splines (smoother #1)

Basic idea: pasting together a bunch of cubic polynomials

- Recall: cubic polynomial $\longrightarrow y = a + bX + cX^2 + dX^3$



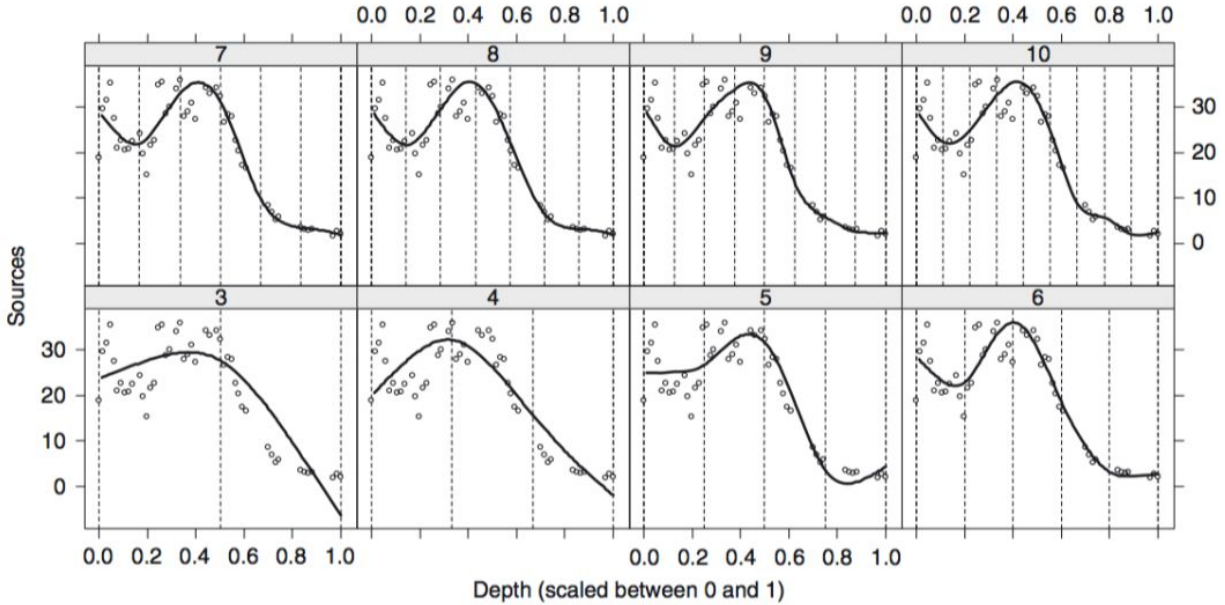
knots (○) = point at which curves are joined. Dotted line is an extra illustration to show the curve is smooth

Objective: to find a set of cubic polynomials that are smooth at the breaks (called 'knots') (○)

- Knots will match neighboring two sections in value and first two derivatives

Cubic regression splines (smoother #1)

We can choose the number of knots to adjust the smoothness (or wiggliness)



Number at the top of each panel indicates the # of knots used, the more knots the more wiggly, but converges around 5-6 knots

Tradeoff between under and overfitting

- Too smooth = miss patterns in the data
- Too wiggly = will fit subtle/random patterns that reduce predictive performance

Key challenge of GAMs: find optimal curve that balances under and overfitting

Ok cool but how wiggly: GCV score

generalized cross-validation score

This helps us determine how wiggly our curve should be

- find the maximum likelihood estimates for the spline parameters, using similar algorithms as for GLMs ⟶ constant λ determines the penalty for wiggliness

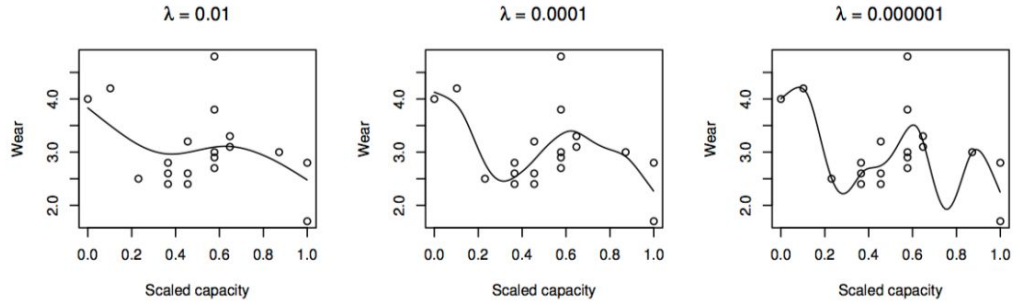


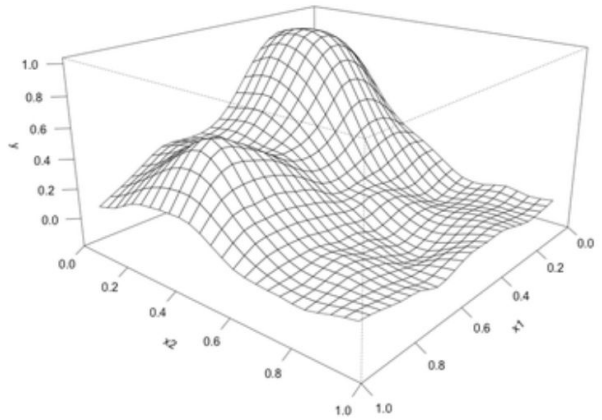
Figure 3.8 *Penalized regression spline fits to the engine wear versus capacity data using three different values for the smoothing parameter.*

If λ =0, then there is no penalty for wiggliness,

Thin plate regression splines (smoother #2)

Thin plate splines are the two dimensional analog of cubic regression splines

Refers to a physical analogy involving the bending of a thin sheet of metal ⟶ rather than a single curve, thin plate splines are represented as a bendable surface



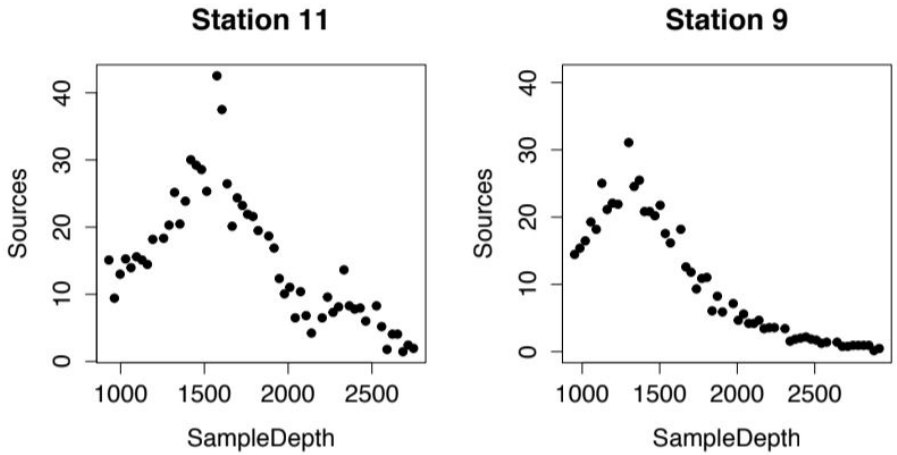
Smoother and factor interactions

Becomes more interesting to compare two smooths → in this case compare two stations of bioluminescence

Bioluminescence has nonlinear, smooth looking relationship with depth, but relationship looks a little different between stations

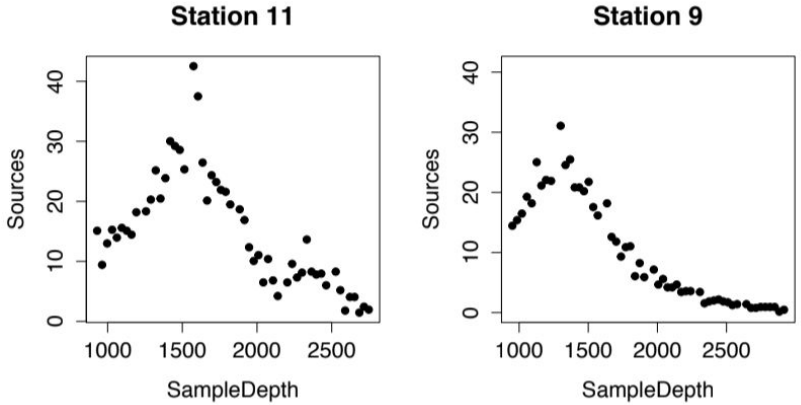
- Peak bioluminescence at station 9 looks shallower

Does the relationship between Y and X differ between these groups?



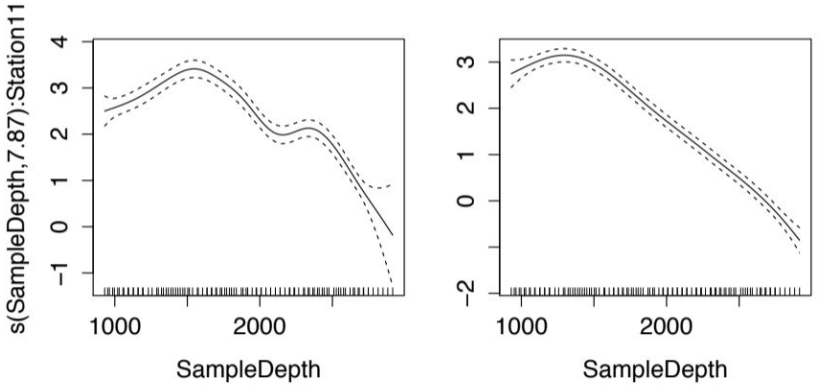
Sources = estimated number of sources of bioluminescence per cubic meter.

Smoother and factor interactions cont.



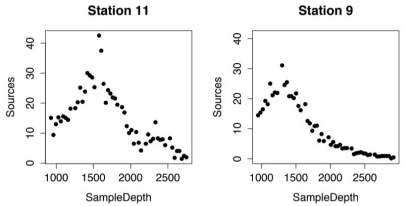
Method 1: fit a separate smoother for each station

```
mod1 = gam(log(Sources) ~ Station + s(SampleDepth, by = Station), data = lumsub)
```



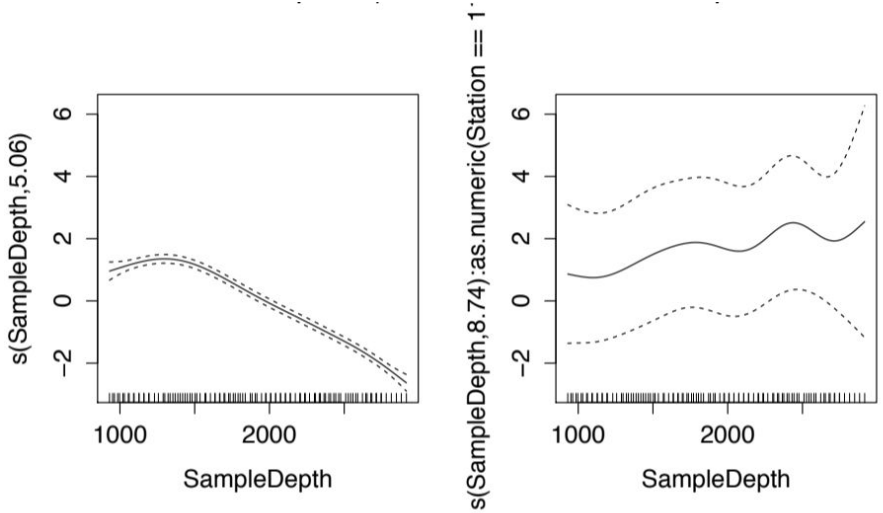
Can visually see here that the relationship between depth and source are different between the stations, each smoother is significant

Smoother and factor interactions cont.



Method 2: fit 1) a smoother for the depth profile at station 9, and 2) a smoother that quantifies the difference between station 11 and station 9

```
mod2 = gam(log(Sources) ~ Station + s(SampleDepth) + s(SampleDepth, by =
as.numeric(Station == 11)), data = lumsub)
```



smoother quantifies how station 11 differs from the overall relationship fit by the first smoother

First smoother is similar to what we found for station 9 before. Second smoother quantifies how station 11 differs from station 9, the closer to zero the curve is, the less of a difference between the two curves there is

- shallower depths (around 1000 meters) the two curves are similar, but as depth increases there are more bioluminescent sources at station 11
- confidence intervals on this effect are large, but the overall difference in shape between the sites is significant