Generalized Additive Models

Overview

- What is a GAM?
- What is smoothing?
- How do GAMs work?
- Fitting GAMs using dsm

What is a GAM?

”gam”

1. *Collective noun used to refer to a group of whales, or rarely also of porpoises; a pod.*
2. *(by extension) A social gathering of whalers (whaling ships).*

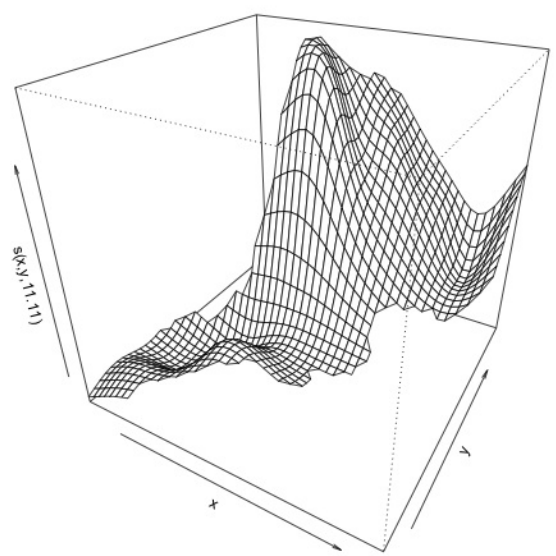
(via Natalie Kelly, AAD. Seen in Moby Dick.)

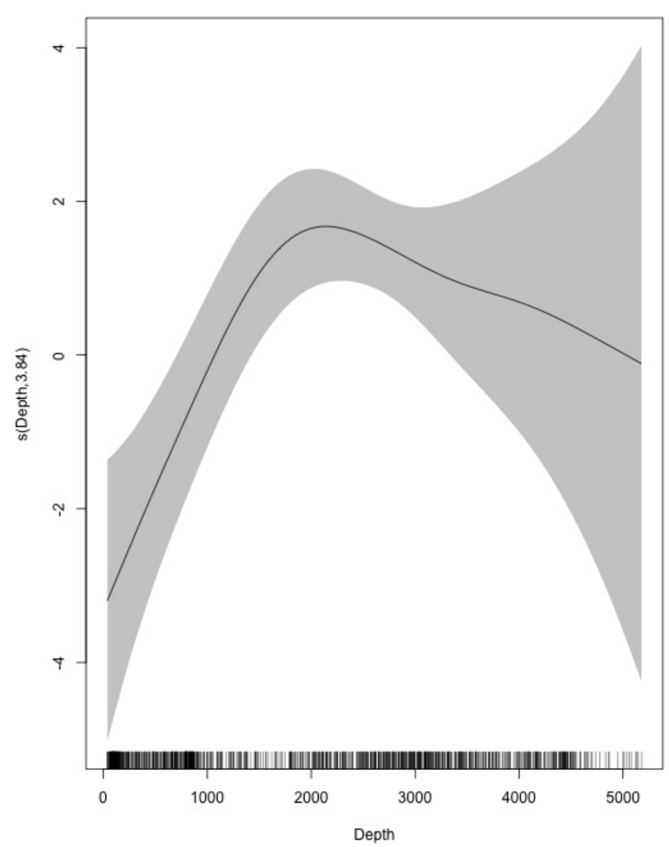
Generalized Additive Models

- Generalized: many response distributions
- Additive: terms **add** together
- Models: well, it's a model…

What does a model look like?

- Count n_j distributed according to some count distribution
- Model as sum of terms





Mathematically...

Taking the previous example…

$$n_j = A_j \hat{p}_j \exp\left[\beta_0 + s(\mathrm{y}_j) + s(\mathrm{Depth}_j)\right] + \epsilon_j$$

where $\epsilon_j \sim N(0, \sigma^2), \quad n_j \sim$ count distribution

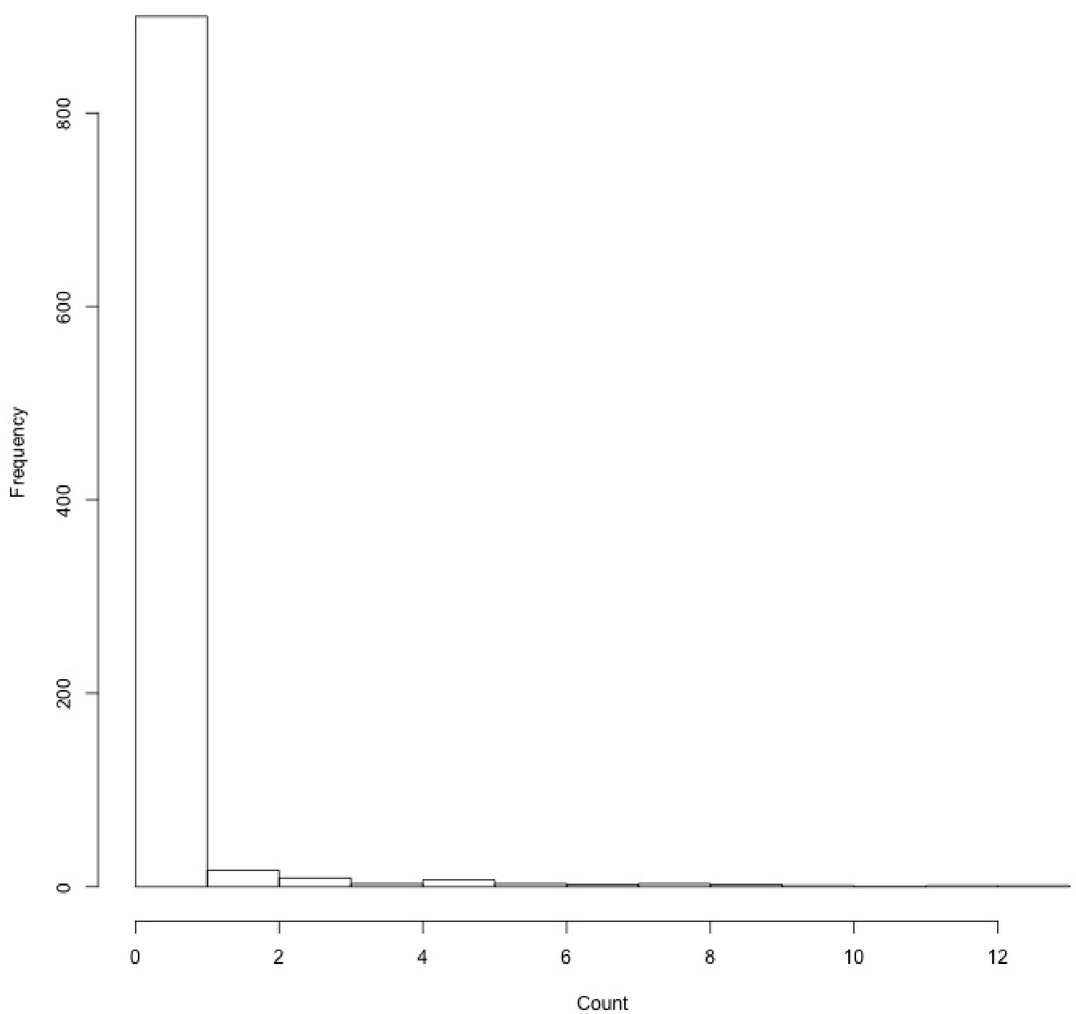
- area of segment - offset
- probability of detection in segment
- link function
- model terms

Response

$$n_j = A_j \hat{p}_j \exp\left[\beta_0 + \boldsymbol{s}(\mathrm{y}_j) + \boldsymbol{s}(\mathrm{Depth}_j)\right] + \epsilon_j$$

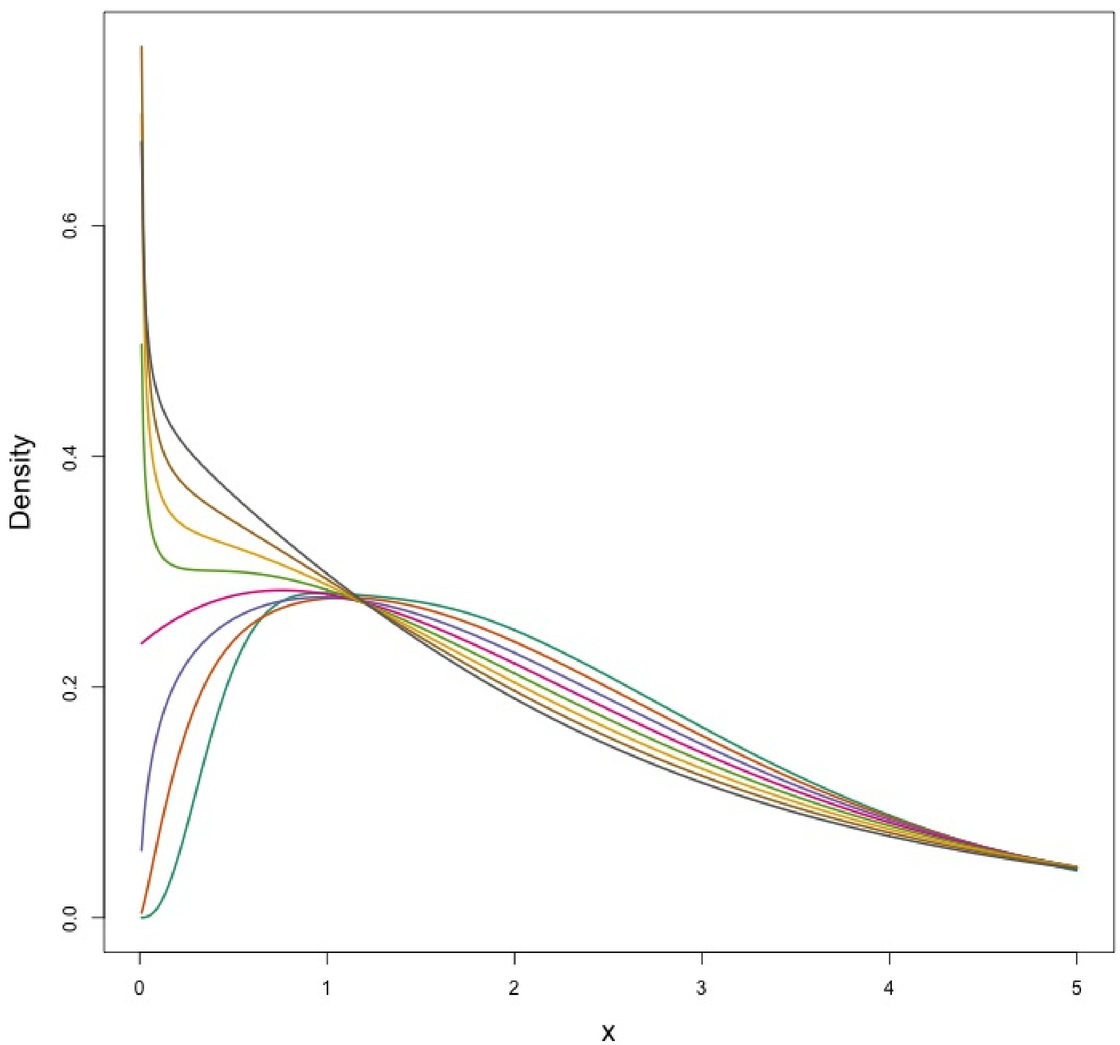
where $\epsilon_j \sim N(0, \sigma^2)$, $\quad n_j \sim \text{count distribution}$

Count distributions



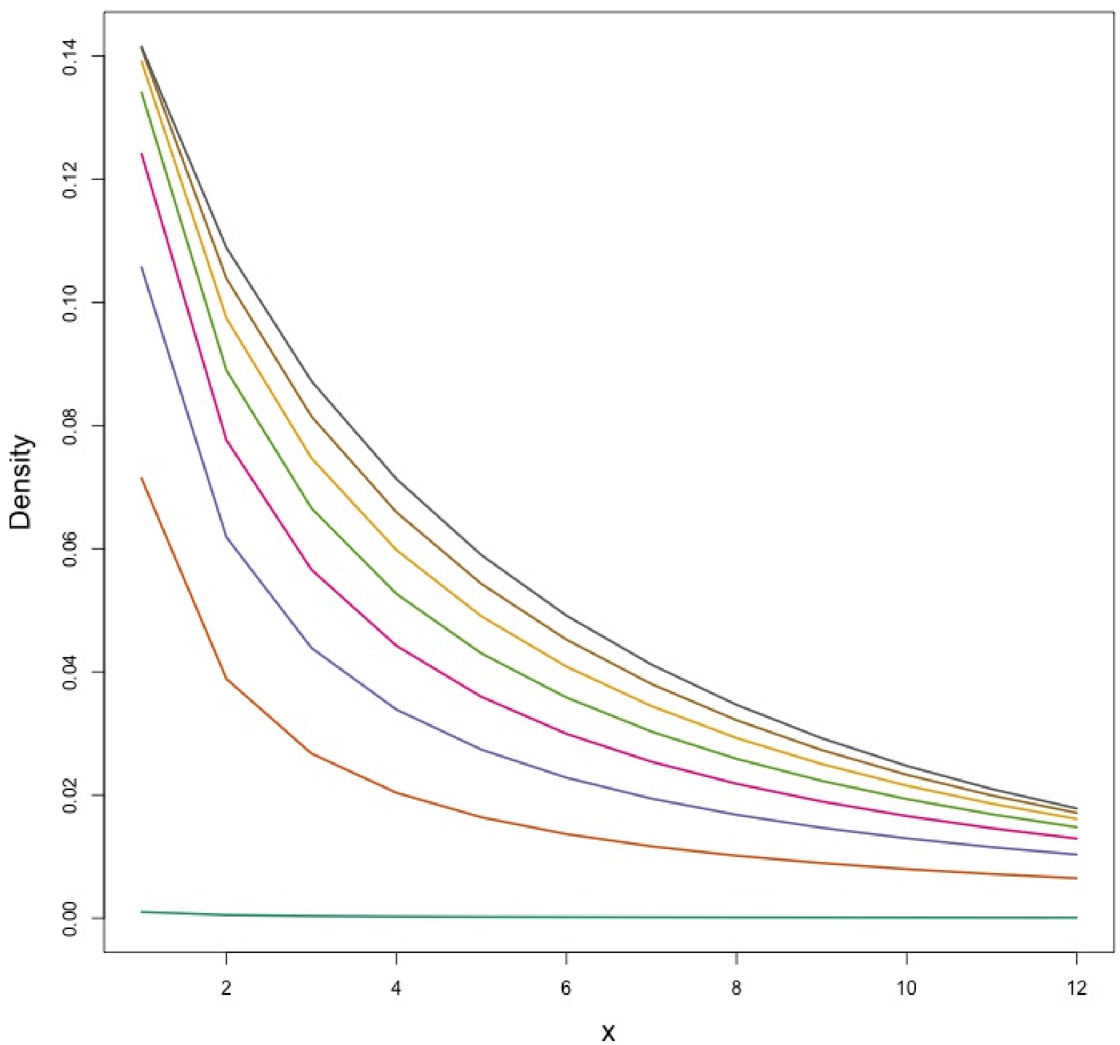
- Response is a count (not not always integer)
- Often, it's mostly zero (that's complicated)
- Want response distribution that deals with that
- Flexible mean-variance relationship

Tweedie distribution



- $\mathrm{Var}\,(\mathrm{count}) = \phi\mathbb{E}(\mathrm{count}$
- Common distributions are sub cases:
  - $q = 1 \Rightarrow$ Poisson
  - $q = 2 \Rightarrow$ Gamma
  - $q = 3 \Rightarrow$ Normal
- We are interested in $1 < q < 2$
- (here $q = 1.2, 1.3, \ldots, 1.9$)

Negative binomial distribution



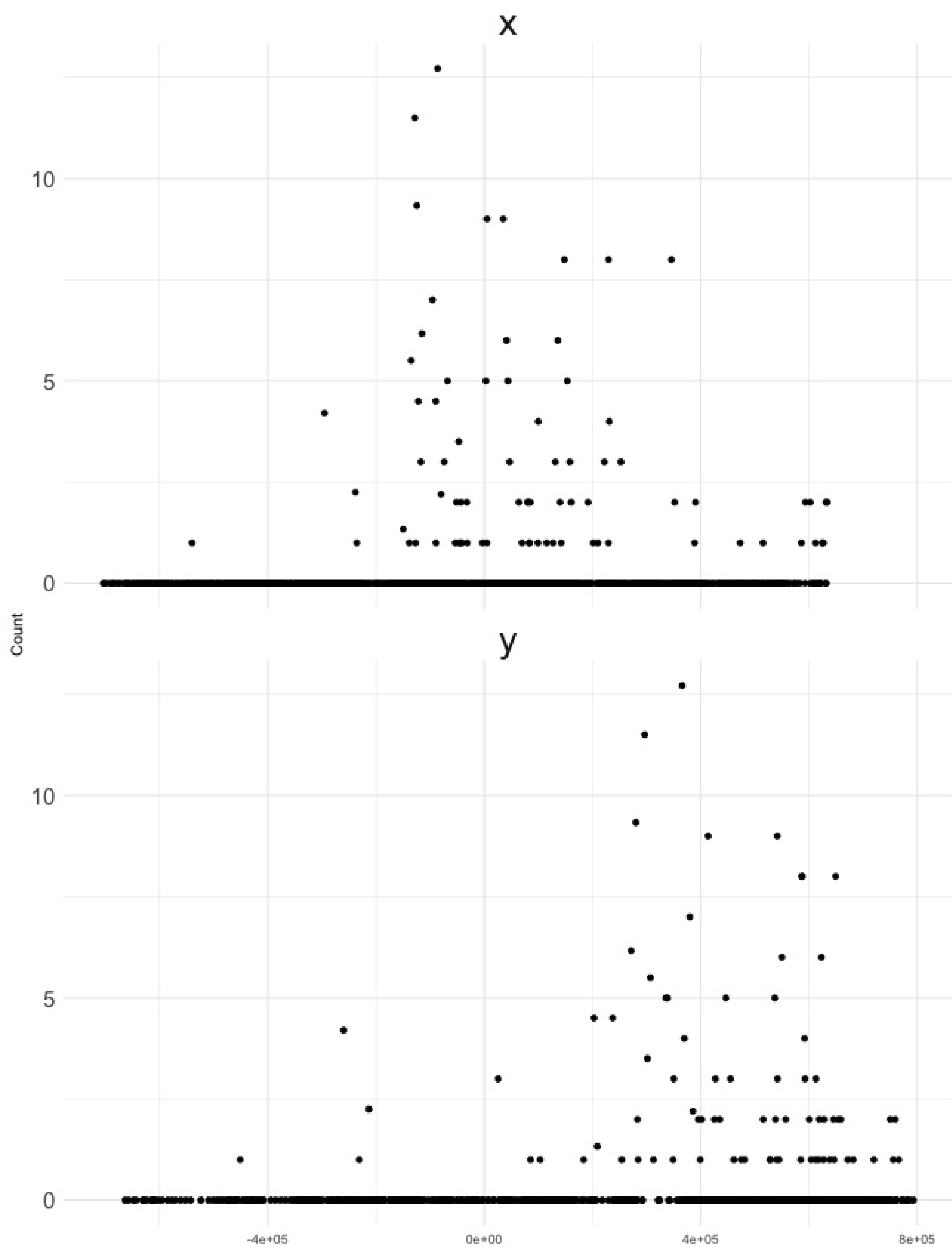
- $\mathrm{Var}\left(\mathrm{count}\right) = \mathbb{E}(\mathrm{count}) + \kappa \mathbb{E}(\mathrm{count})^2$
- Estimate κ
- Is quadratic relationship a "strong" assumption?
- Similar to Poisson: $\mathrm{Var}\left(\mathrm{count}\right) = \mathbb{E}(\mathrm{count})$

Smooth terms

$$n_j = A_j \hat{p}_j \exp\left[\beta_0 + \boldsymbol{s}(\mathrm{y}_j) + \boldsymbol{s}(\mathrm{Depth}_j)\right] + \epsilon_j$$

where $\epsilon_j \sim N(0, \sigma^2)$, $\quad n_j \sim$ count distribution

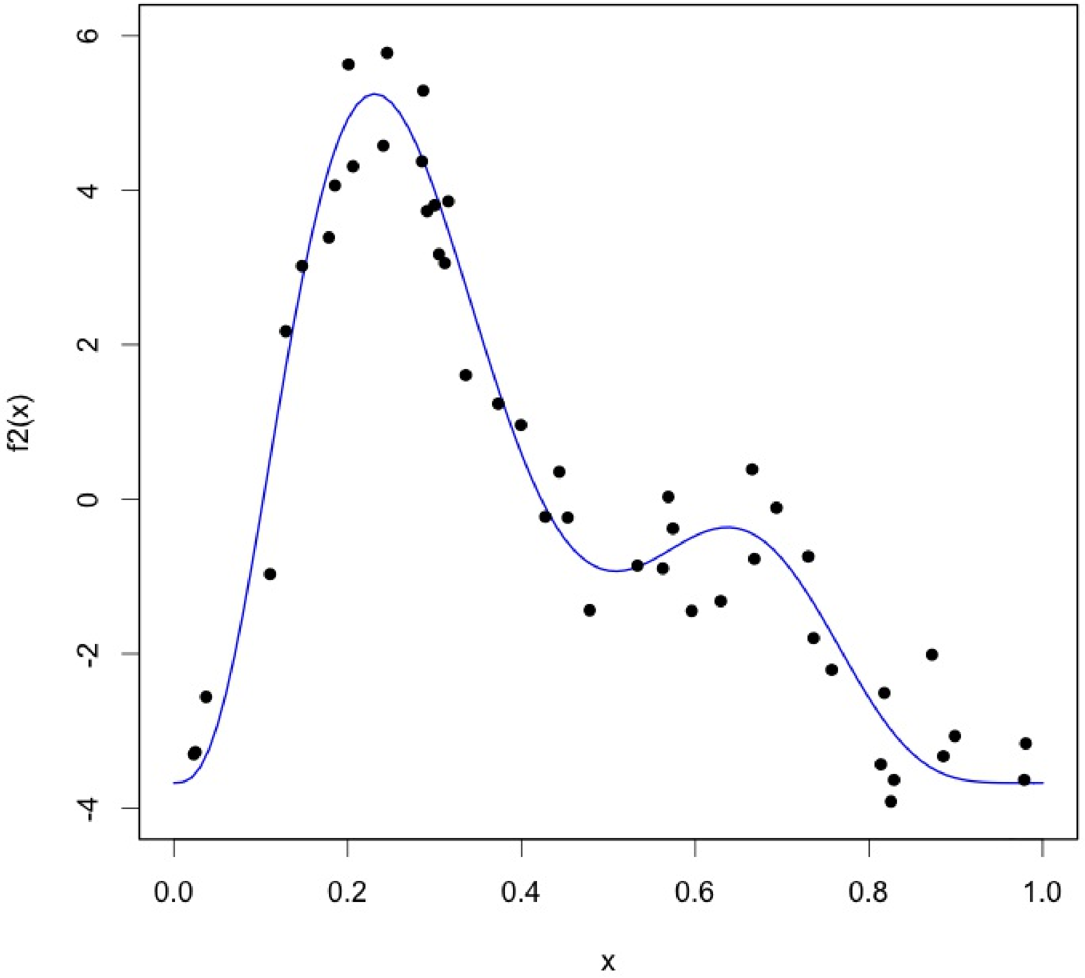
Okay, but what about these ”s” things?



- Think s=**smooth**
- Want to model the covariates flexibly
- Covariates and response not necessarily linearly related!
- Want some wiggles

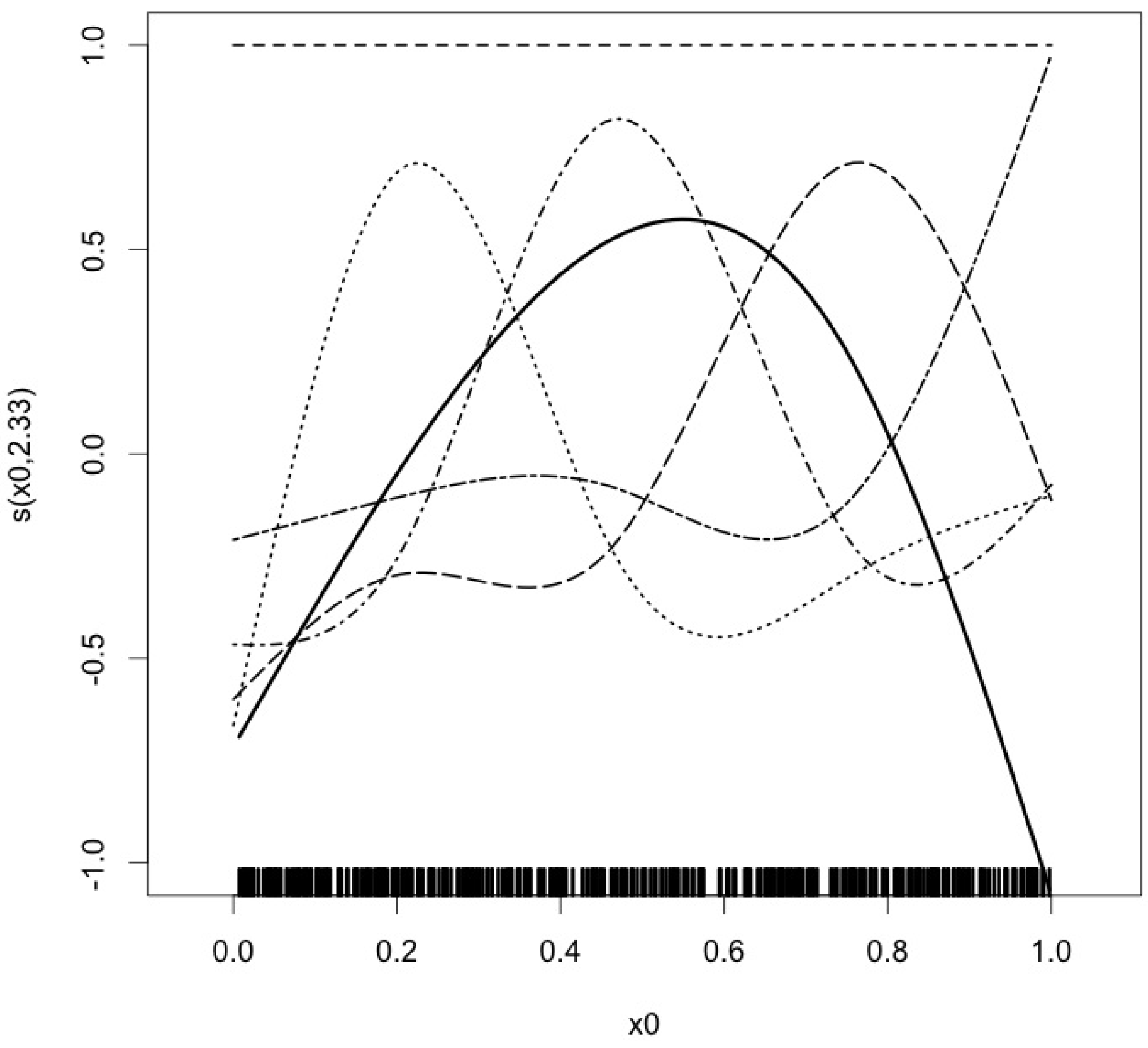
What is smoothing?

Straight lines vs. interpolation



- Want a line that is “close” to all the data
- Don't want interpolation – we know there is “error”
- Balance between interpolation and “fit”

Splines

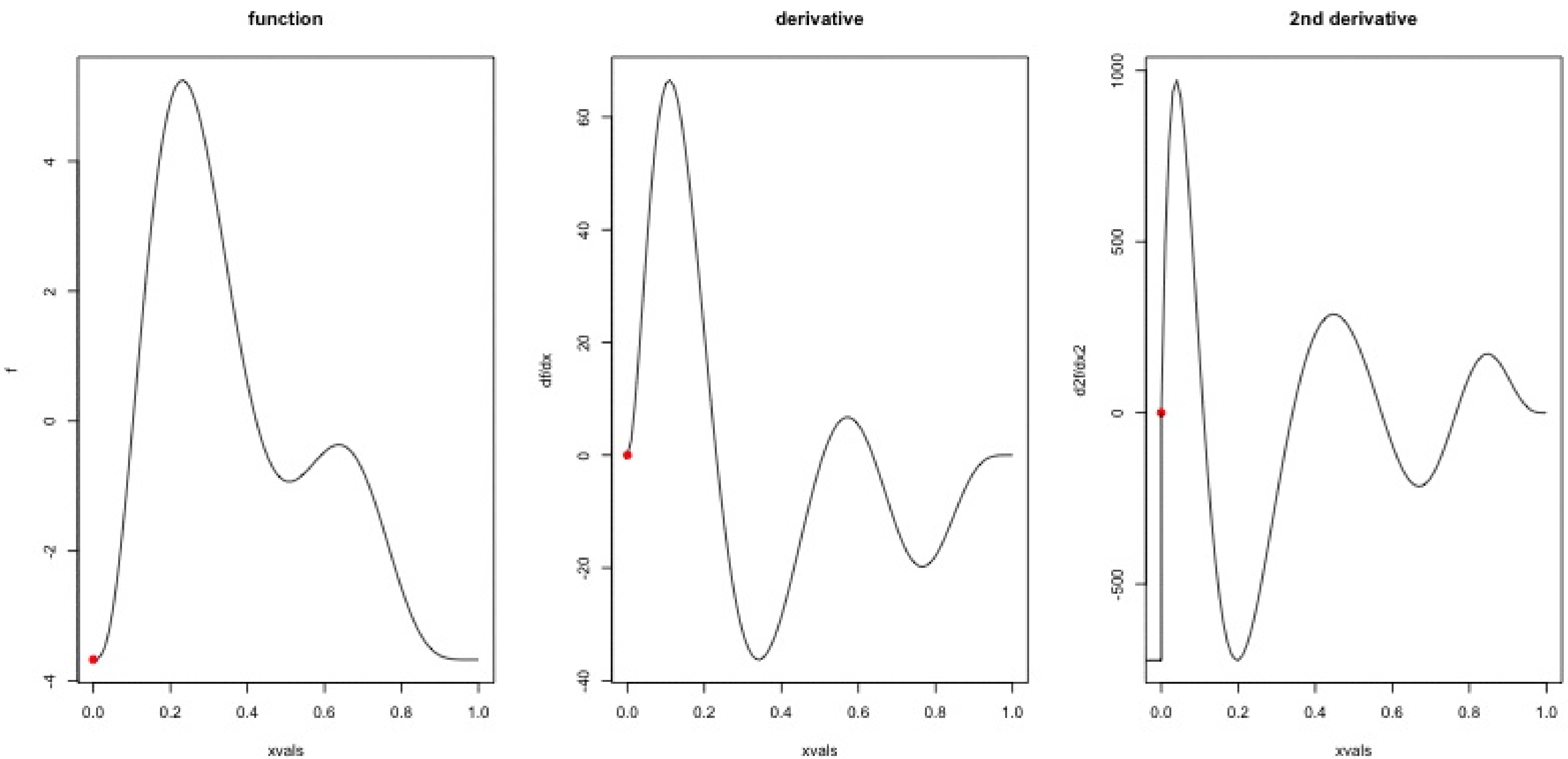


- Functions made of other, simpler functions
- **Basis functions** b_k, estimate β_k
- $s(x) = \sum_{k=1}^{K} \beta_k b_k(x)$
- Makes the maths much easier

Measuring wigglyness

- Visually:
  - Lots of wiggles == NOT SMOOTH
  - Straight line == VERY SMOOTH
- How do we do this mathematically?
  - Derivatives!
  - (Calculus *was* a useful class afterall)

Wigglyness by derivatives



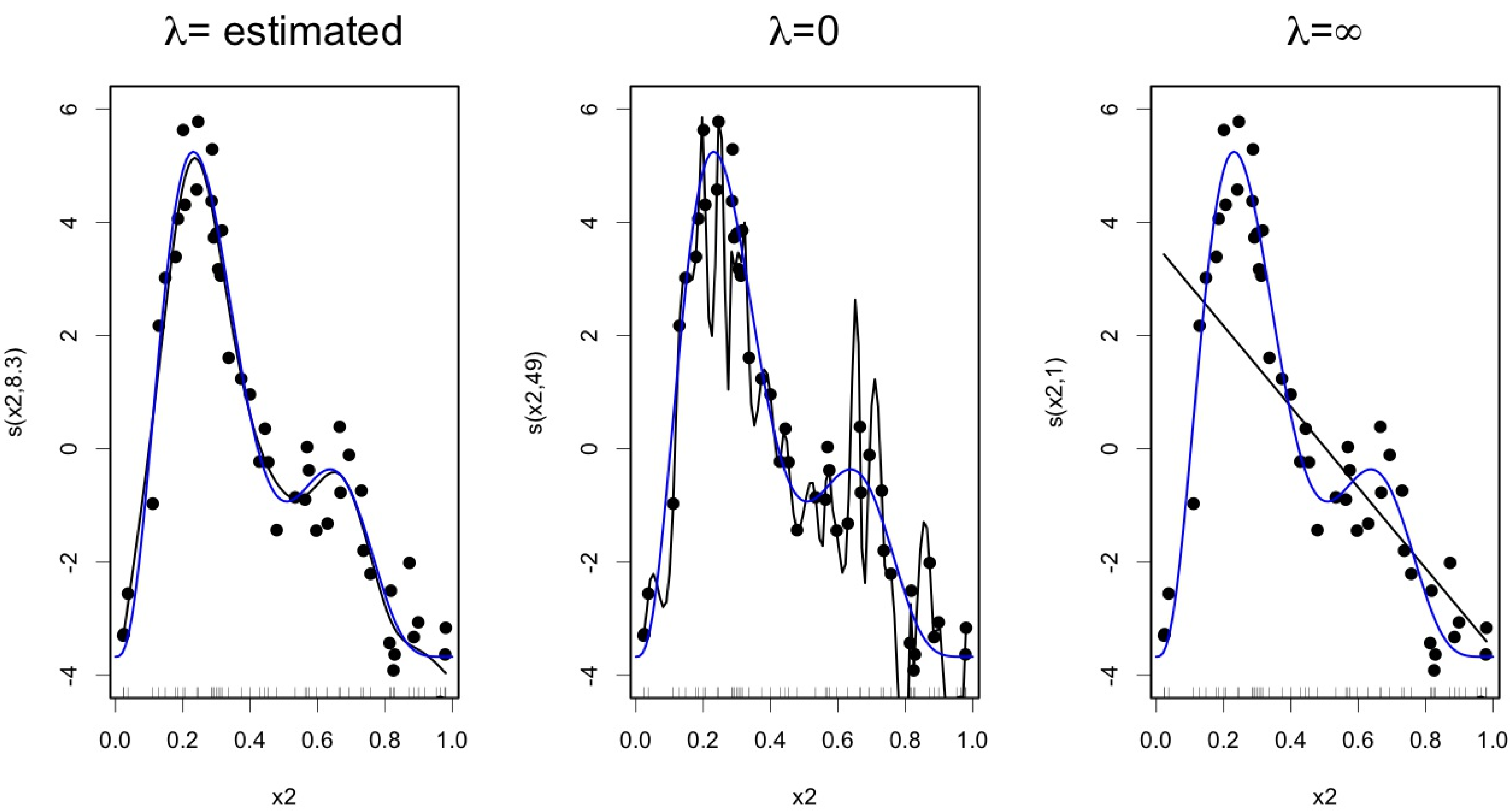
Making wigglyness matter

- Integration of derivative (squared) gives wigglyness
- Fit needs to be **penalised**
- **Penalty matrix** gives the wigglyness
- Estimate the β_k terms but penalise objective
    - "closeness to data" + penalty

Penalty matrix

- For each b_k calculate the penalty
- Penalty is a function of β
  - $\lambda\beta^{\mathrm{T}}S\beta$
- S calculated once
- smoothing parameter (λ) dictates influence

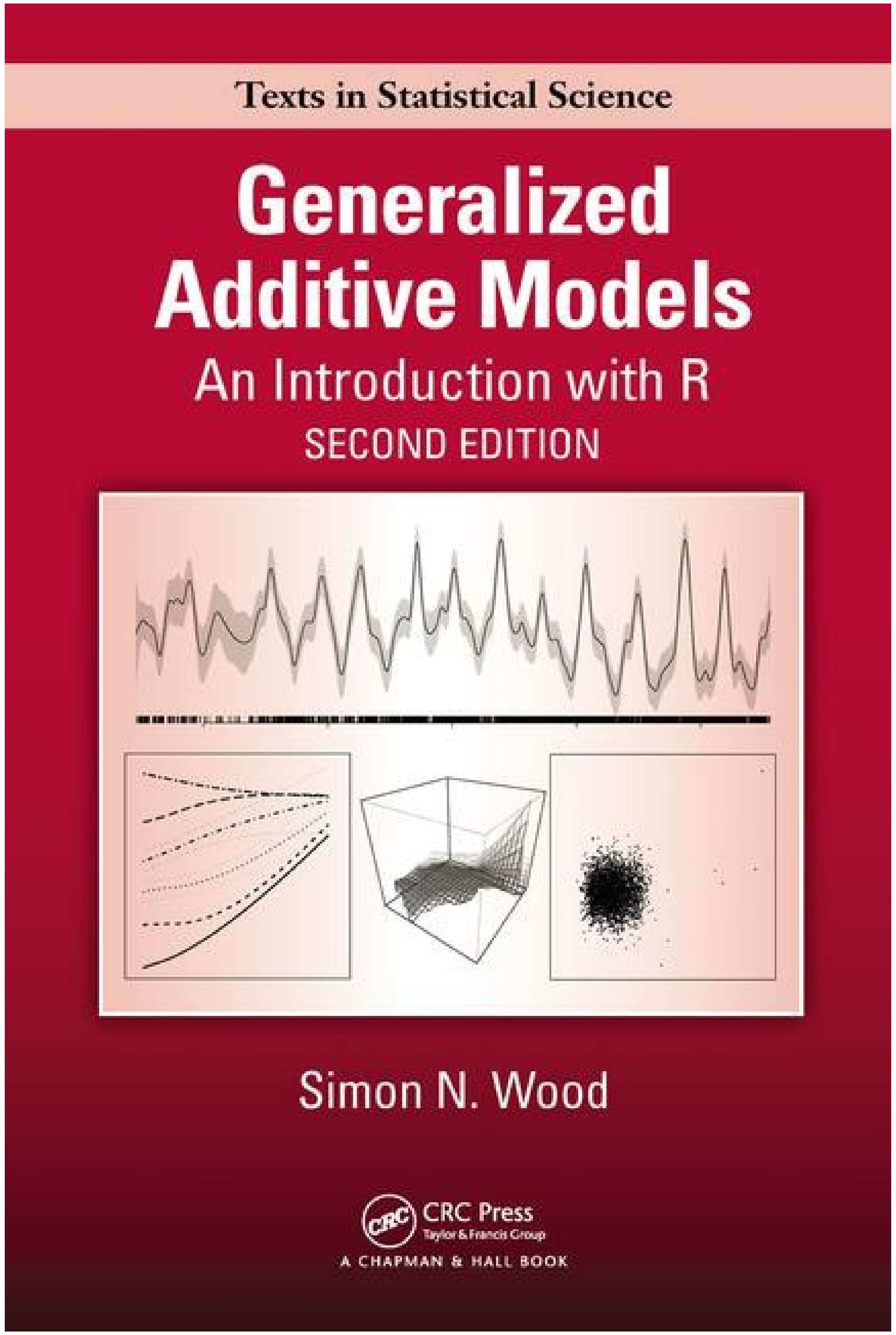
Smoothing parameter



How wiggly are things?

- We can set **basis complexity** or “size” (k)
  - Maximum wigglyness
- Smooths have **effective degrees of freedom** (EDF)
- EDF < k
- Set k “large enough”

Why GAMs are cool...

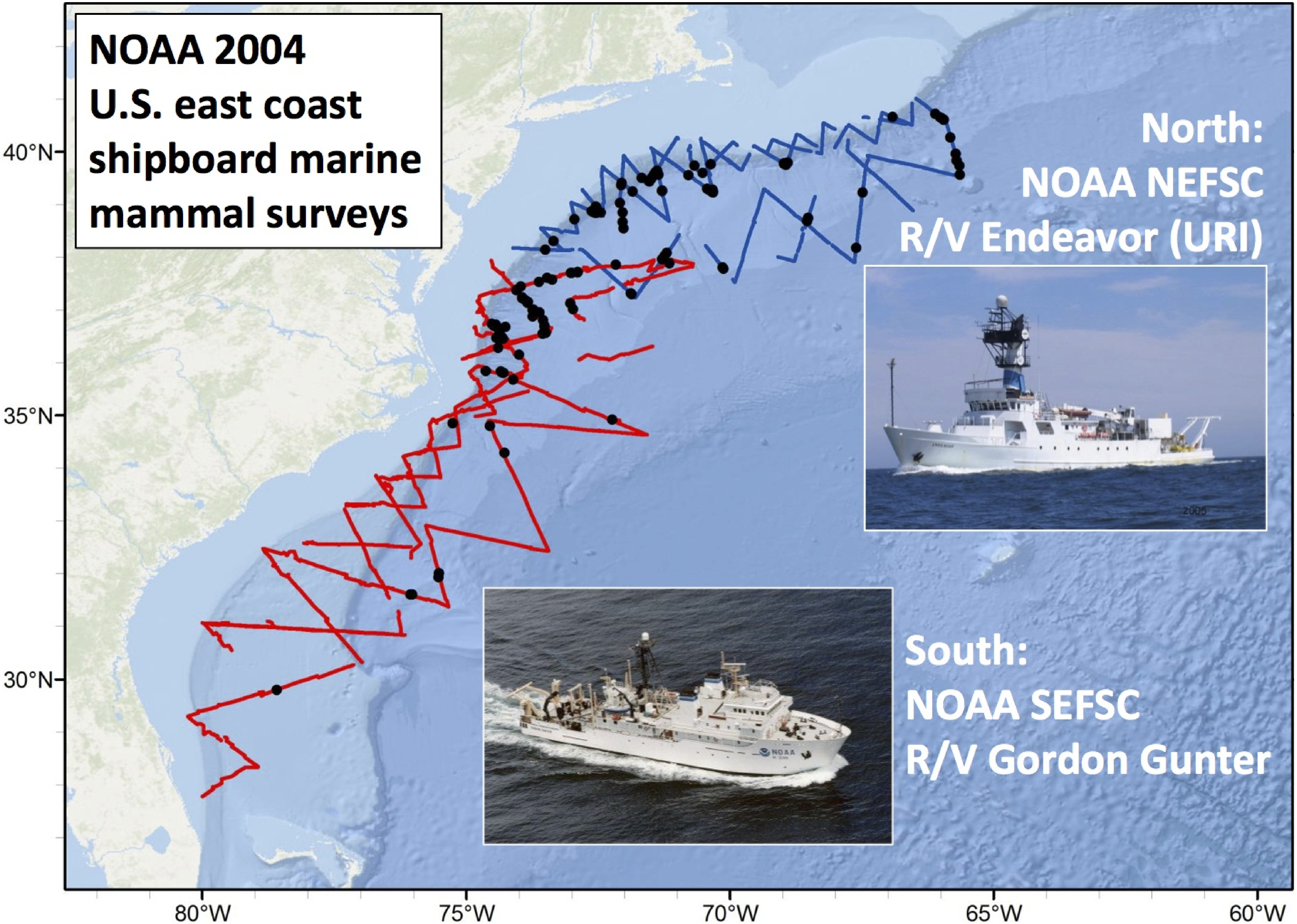


- Fancy smooths (cyclic, boundaries, ...)
- Fancy responses (exp family and beyond!)
- Random effects (by equivalence)
- Markov random fields
- Correlation structures
- See Wood (2006/2017) for a handy intro

Okay, that was a lot of theory…

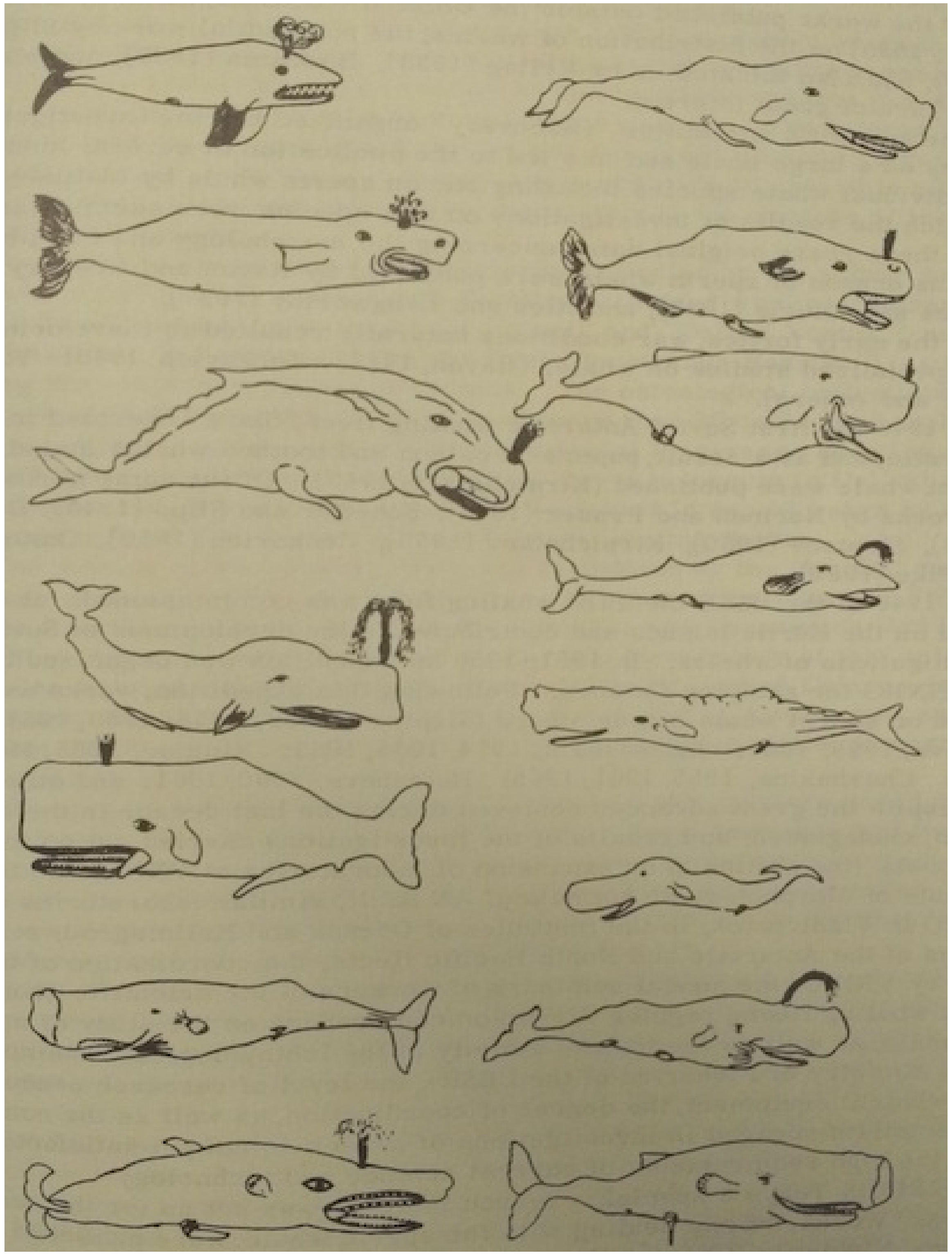
Example data

Example data



Example data

Sperm whales off the US east coast



- Hang out near canyons, eat squid
- Surveys in 2004, US east coast
- Combination of data from 2 NOAA cruises
- Thanks to Debi Palka (NOAA NEFSC), Lance Garrison (NOAA SEFSC) for data. Jason Roberts (Duke University) for data prep.

Model formulation

- Pure spatial, pure environmental, mixed?
- May have some prior knowledge
  - Biology/ecology
- What are drivers of distribution?
- Inferential aim
  - Abundance
  - Ecology

Fitting GAMs using dsm

Translating maths into R

$$n_j = A_j \hat{p}_j \exp\left[\beta_0 + s(\mathrm{y}_j)\right] + \epsilon_j$$

where $\epsilon_j \sim N(0, \sigma^2), \quad n_j \sim$ count distribution

- inside the link: formula=count ~ s(y)
- response distribution: family=nb() or family=tw()
- detectability: ddf.obj=df_hr
- offset, data: segment.data=segs, observation.data=obs

Your first DSM

```
library(dsm)
dsm_x_tw <- dsm(count~s(x), ddf.obj=df,
                segment.data=segs, observation.data=obs,
                family=tw())
```

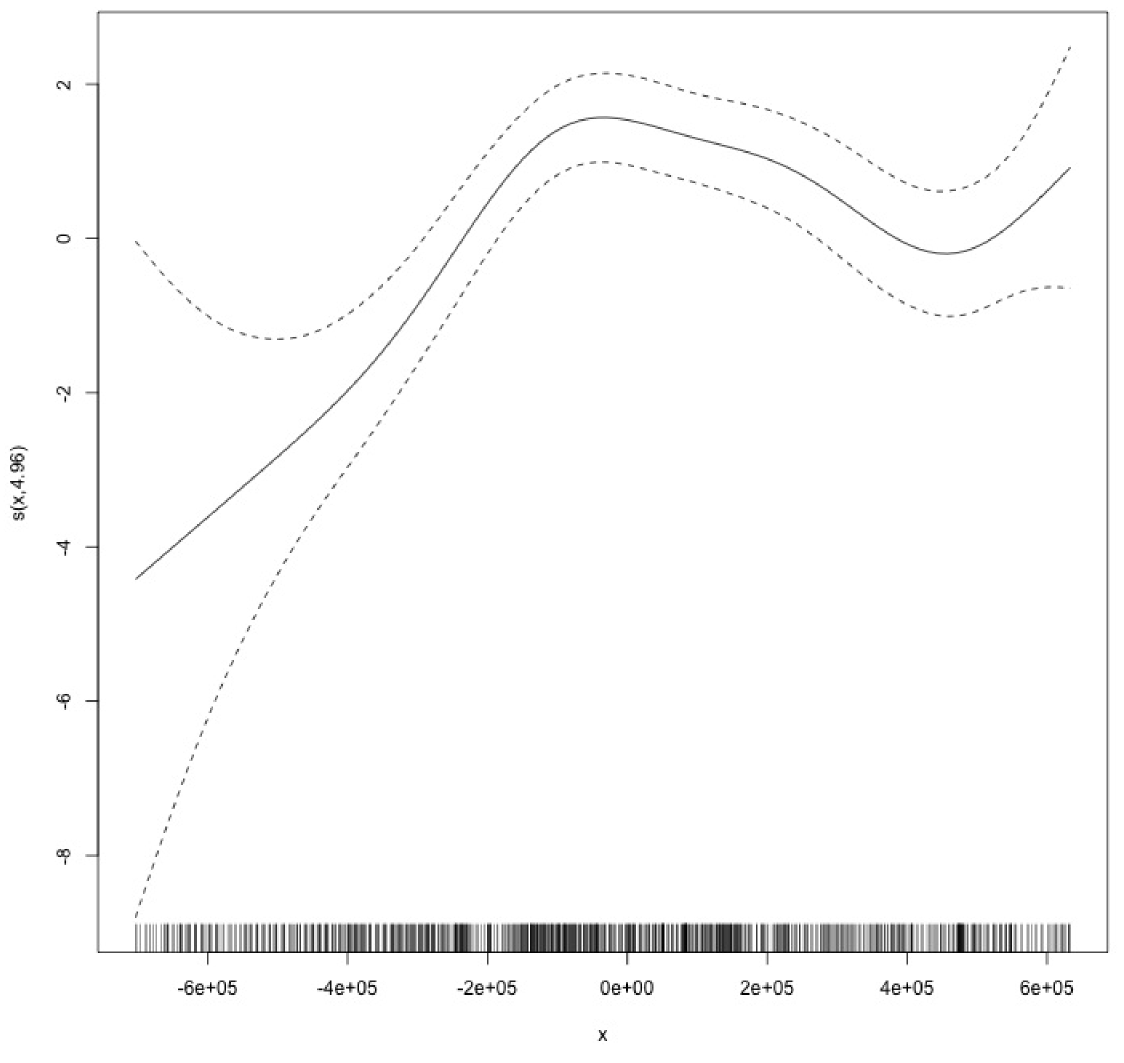
dsm is based on mgcv by Simon Wood

What did that do?

```
summary(dsm_x_tw)
```

```
Family: Tweedie(p=1.326)
Link function: log

Formula:
count ~ s(x) + offset(off.set)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -19.8115     0.2277  -87.01   <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(x) 4.962  6.047 6.403 1.07e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.0283   Deviance explained = 17.7%
-REML = 409.94  Scale est. = 6.0413    n = 949
```

Plotting



- plot(dsm_x_tw)
- Dashed lines indicate +/- 2 standard errors
- Rug plot
- On the link scale
- EDF on y axis

Adding a term

- Just use +

```
dsm_xy_tw <- dsm(count ~ s(x) + s(y),
                 ddf.obj=df,
                 segment.data=segs,
                 observation.data=obs,
                 family=tw())
```

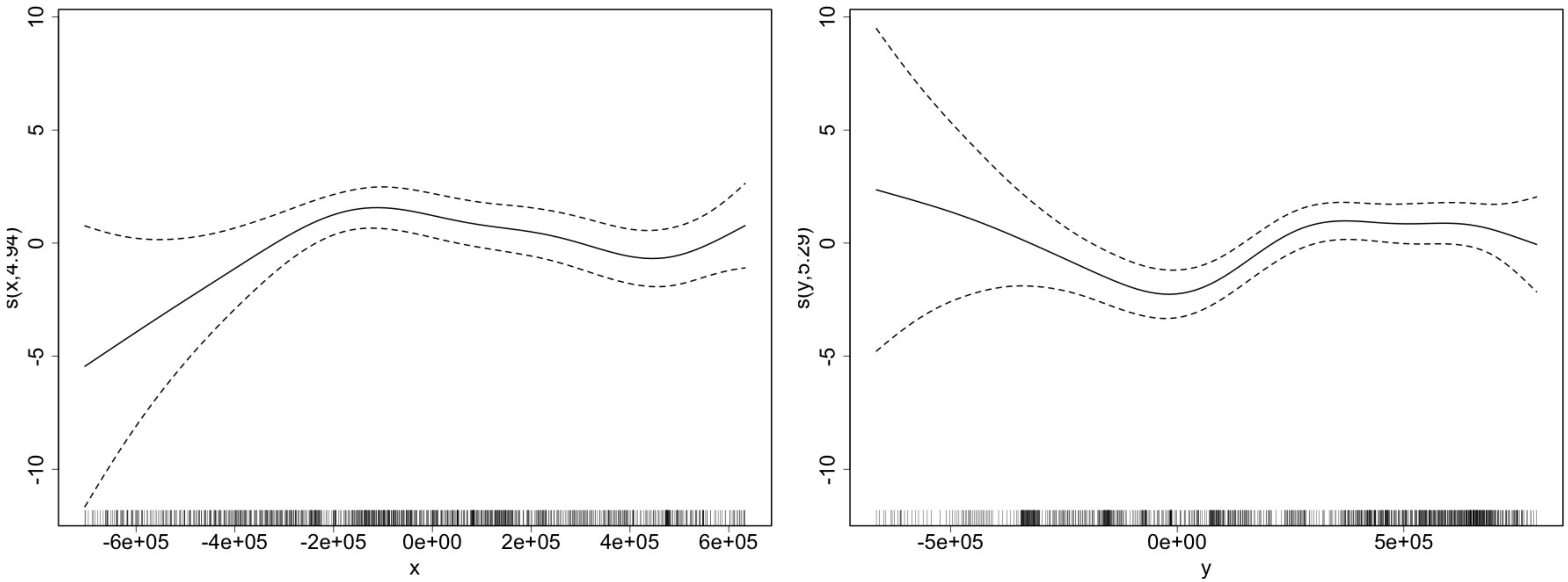
Summary

```
summary(dsm_xy_tw)
```

```
Family: Tweedie(p=1.306)
Link function: log

Formula:
count ~ s(x) + s(y) + offset(off.set)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.0908     0.2381  -84.39   <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(x) 4.943  6.057 3.224 0.004239 **
s(y) 5.293  6.420 4.034 0.000322 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.0678   Deviance explained = 27.3%
-REML = 399.84  Scale est. = 5.3157    n = 949
```

Plotting

```
plot(dsm_xy_tw, pages=1)
```



- scale=0: each plot on different scale
- pages=1: plot together

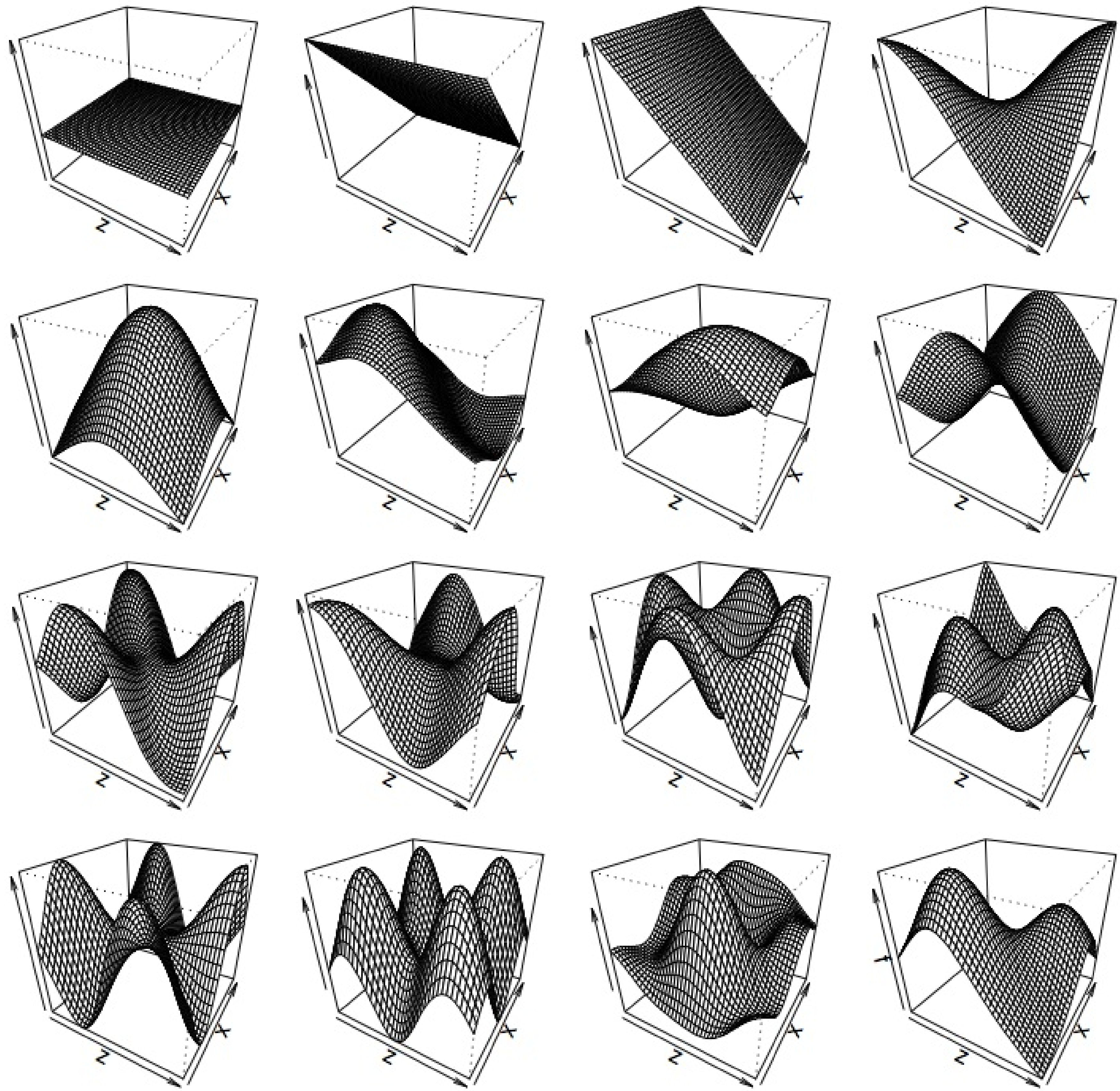
Bivariate terms

- Assumed an additive structure
- No interaction
- We can specify s(x,y) (and s(x,y,z,…))

Thin plate regression splines

- Default basis
- One basis function per data point
- Reduce # basis functions (eigendecomposition)
- Fitting on reduced problem
- Multidimensional

Thin plate splines (2-D)



Bivariate spatial term

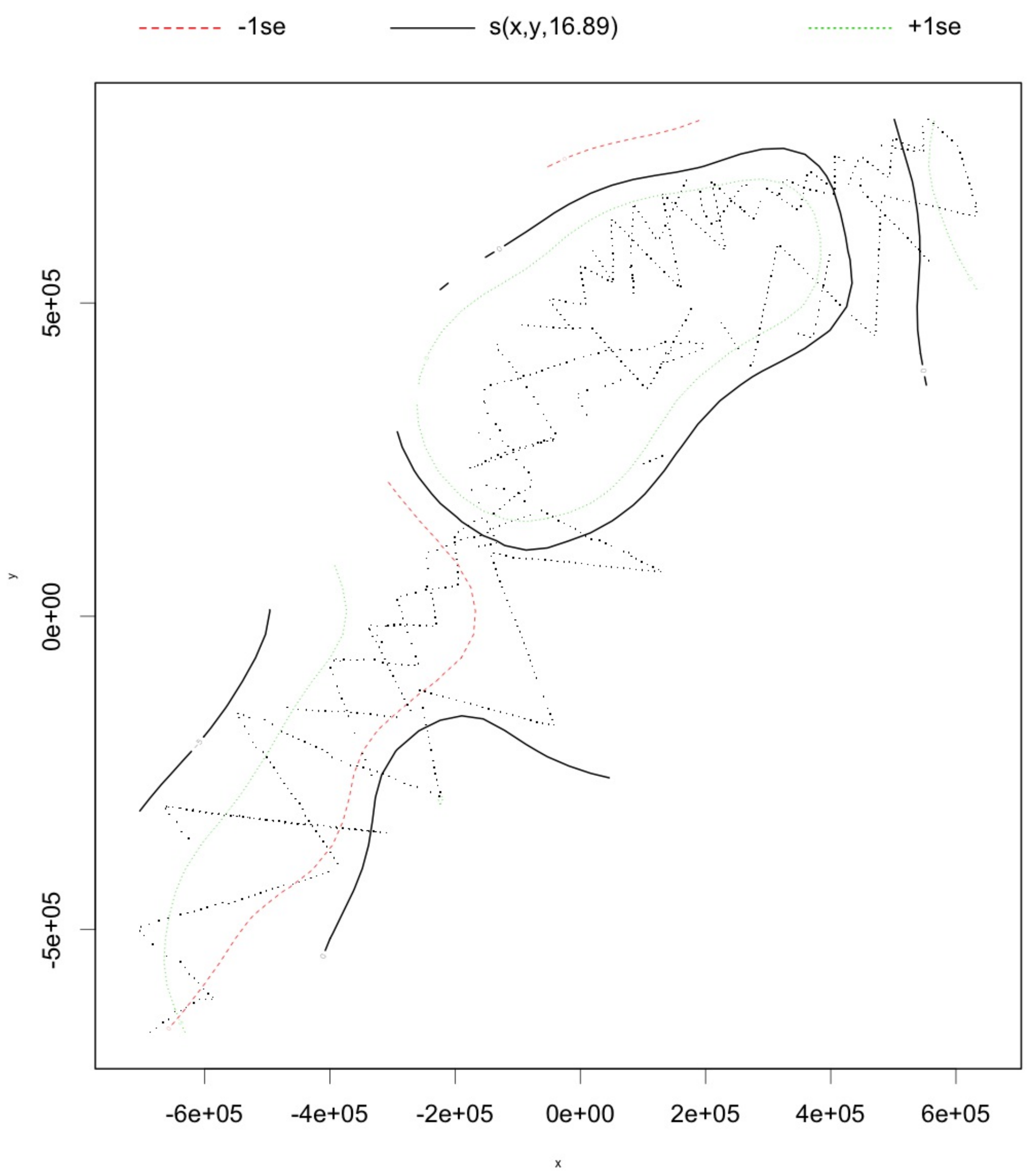
```
dsm_xyb_tw <- dsm(count ~ s(x, y),
                  ddf.obj=df,
                  segment.data=segs,
                  observation.data=obs,
                  family=tw())
```

Summary

```
summary(dsm_xyb_tw)
```

```
Family: Tweedie(p=1.29) 
Link function: log 

Formula:
count ~ s(x, y) + offset(off.set)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -20.2745     0.2477  -81.85   <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(x,y) 16.89  21.12 4.333 3.73e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.102   Deviance explained = 34.6%
-REML = 394.86  Scale est. = 4.8248    n = 949
```

Plotting... erm...

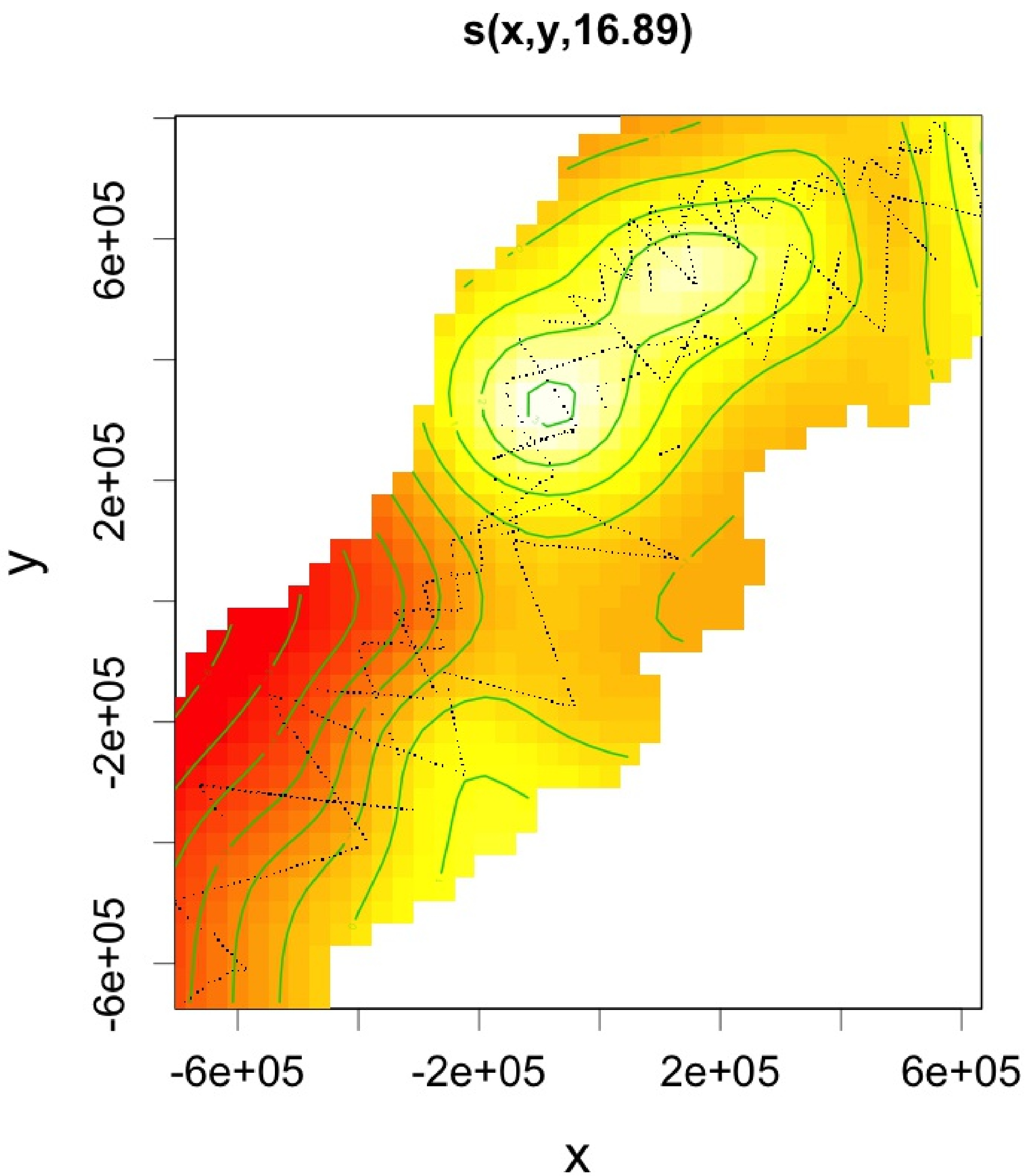



```
plot(dsm_xyb_tw)
```

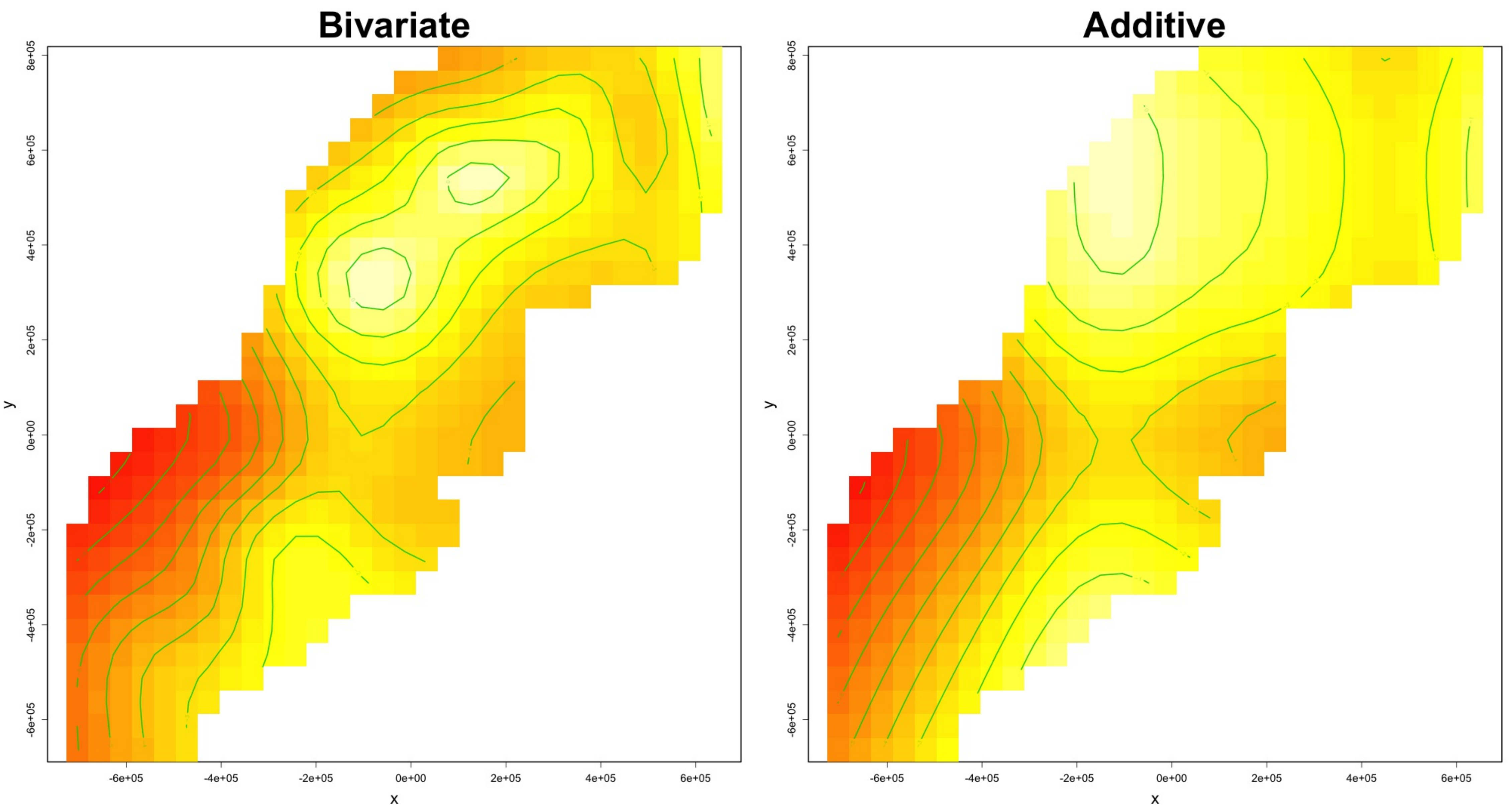
Let's try something different

```
plot(dsm_xyb_tw, select=1,
     scheme=2, asp=1)
```

- Still on link scale
- too.far excludes points far from data



Comparing bivariate and additive models



Let’s have a go…