Hierarchical Generalized Additive Models in R with mgcv

Natalya Gallo

Hierarchical generalized additive models in ecology: an introduction with mgcv

Eric J. Pedersen[1,2], David L. Miller[3,4], Gavin L. Simpson[5,6] and Noam Ross[7]

[1] Northwest Atlantic Fisheries Center, Fisheries and Oceans Canada, St. John's, NL, Canada
[2] Department of Biology, Memorial University of Newfoundland, St. John's, NL, Canada
[3] Centre for Research into Ecological and Environmental Modelling, University of St Andrews, St Andrews, UK
[4] School of Mathematics and Statistics, University of St Andrews, St Andrews, Scotland, UK
[5] Institute of Environmental Change and Society, University of Regina, Regina, SK, Canada
[6] Department of Biology, University of Regina, Regina, SK, Canada
[7] EcoHealth Alliance, New York, NY, USA

ABSTRACT

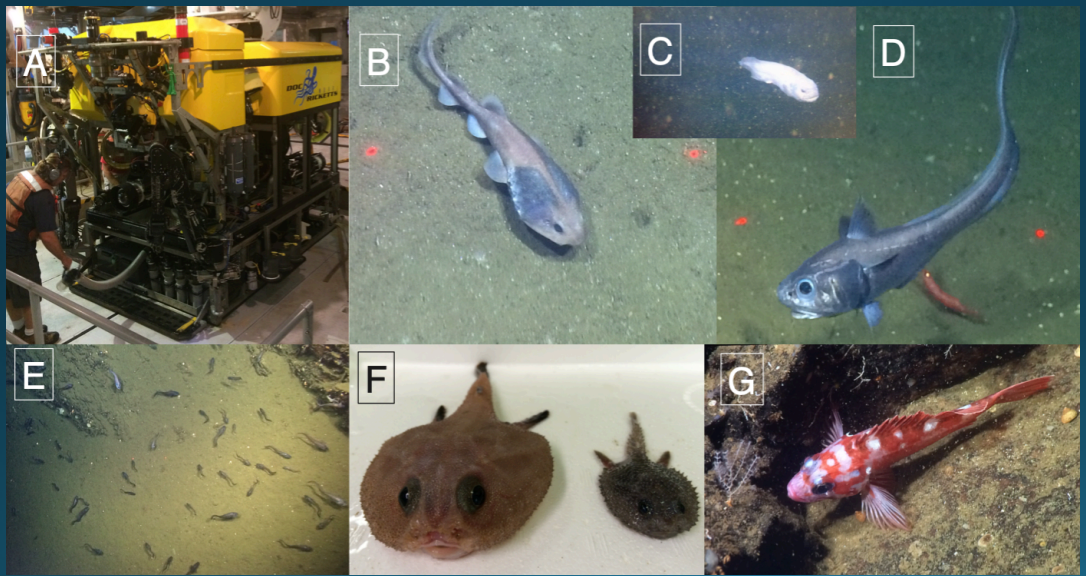
In this paper, we discuss an extension to two popular approaches to modeling complex structures in ecological data: the generalized additive model (GAM) and the hierarchical model (HGLM). The hierarchical GAM (HGAM), allows modeling of nonlinear functional relationships between covariates and outcomes where the shape of the function itself varies between different grouping levels. We describe the theoretical connection between HGAMs, HGLMs, and GAMs, explain how to model different assumptions about the degree of intergroup variability in functional response, and show how HGAMs can be readily fitted using existing GAM software, the **mgcv** package in R. We also discuss computational and statistical issues with fitting these models, and demonstrate how to fit HGAMs on example data. All code and data used to generate this paper are available at: github.com/eric-pedersen/mixed-effect-gams.

R-Users Meeting

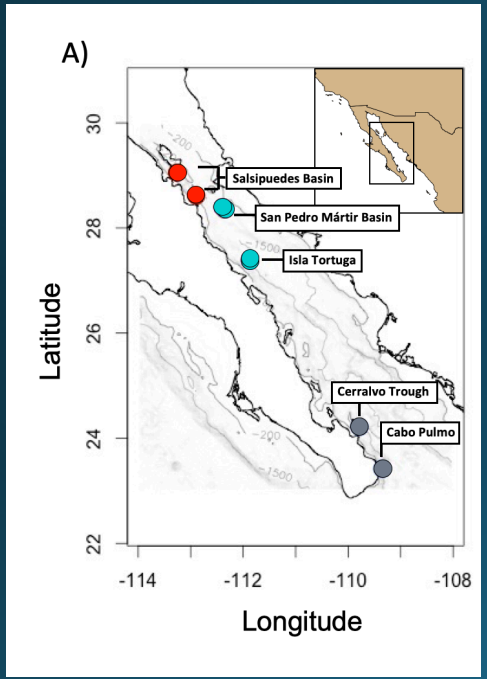
January 29, 2020

How My GAM Story Begins

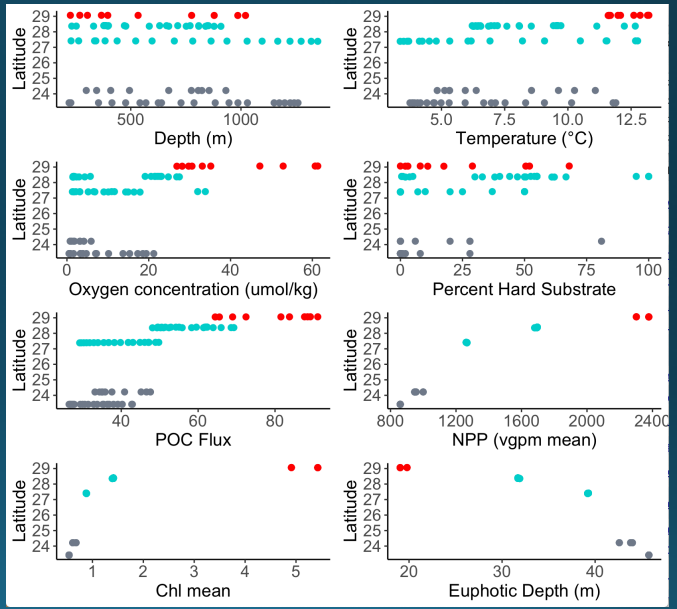
What Explains Differences in Demersal Fish Density and Diversity in the Gulf of California?



MBARI 2015 Cruise to the Gulf of California (Jim Barry – PI)



Demersal fish density & diversity ~ f(environmental variables)



Gallo et al. (Accepted) MEPS

Possible Approaches

Generalized Linear Models

Generalized Additive Models

Random Forest

FISHERIES OCEANOGRAPHY

FISHERIES OCEANOGRAPHY *Fish. Oceanogr. 24:2, 162–176, 2015*

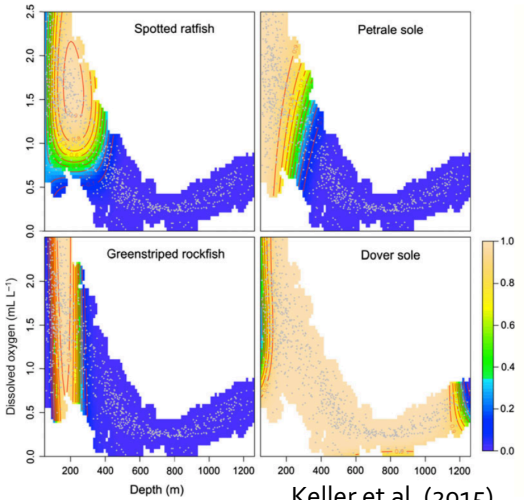
Occurrence of demersal fishes in relation to near-bottom oxygen levels within the California Current large marine ecosystem

AIMEE A. KELLER,[1,*] LORENZO CIANNELLI,[2] W. WALDO WAKEFIELD,[3] VICTOR SIMON,[1] JOHN A. BARTH[2] AND STEPHEN D. PIERCE[2]

[1]*Fishery Resource Analysis and Monitoring Division, Northwest Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, 2725 Montlake Boulevard East, Seattle, Washington 98112, U.S.A.*
[2]*College of Earth, Ocean, and Atmospheric Sciences (CEOAS), Oregon State University, Corvallis, OR 97331-5503, U.S.A.*
[3]*Fishery Resource Analysis and Monitoring Division, Northwest Fisheries Science Center, National Marine Fisheries Service, National Oceanographic and Atmospheric Administration, 2032 S. OSU Drive, Newport, OR 97365, U.S.A*

using a binomial Generalized Additive Model. The models for each species included terms for position, day of the year, salinity, near-bottom temperature and the interaction term between depth and near-bottom DO. Spotted ratfish and petrale sole were sensitive to changes in near-bottom oxygen, while greenstriped rockfish and Dover sole show no changes in probability of occurrence in relation to changes in oxygen concentration.

Key words: bottom dissolved oxygen, demersal fish catch, Dover sole, greenstriped rockfish, Northeast Pacific, petrale sole, probability of occurrence, species richness, spotted ratfish



Keller et al. (2015)

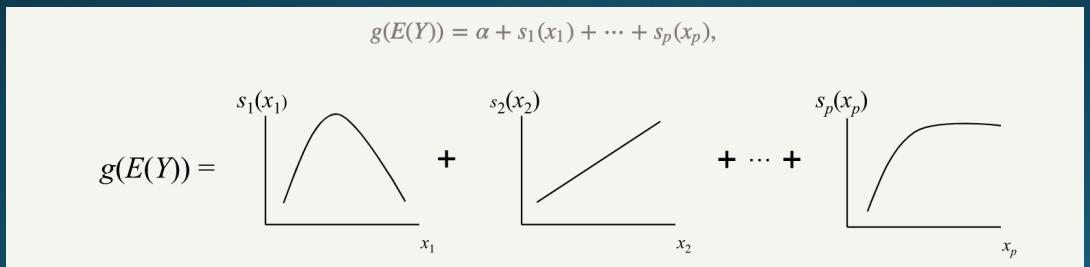
What is a generalized additive model (GAM)?

A GAM is an additive modeling technique where the impact of the predictive variables is captured through smooth functions which can be nonlinear and are fit to the underlying patterns in the data

Useful for predicting complex, nonlinear, possibly interacting relationships, and understanding and making inferences about those relationships

Originally invented by Trevor Hastie and Robert Tibshirani (1986, 1990): the approach has stayed the same though the calculations for model fitting have changed

R package: mgcv (Wood 2017)



Larsen 2015: https://multithreaded.stitchfix.com/blog/2015/07/30/gam/

Generalized Additive Models (GAMs)

Generalized: Can handle many distributions of normal, binomial, count, or other data (not limited to normally distributed data)
Additive: Terms simply add together, but terms themselves are not linear
Model: Model

$$\mathbb{E}(Y) = g^{-1}\left(\beta_0 + \sum_{j=1}^{J} f_j(x_j)\right),$$

Pedersen et al. (2019)

E(Y) = the expected value of the response Y (with an appropriate distribution and link function *g*)
Distributions: binomial (trial), Poisson (count), Gamma (strictly positive real responses)
f_j is a smooth function of the covariate x_j
B_o is an intercept term
g^{-1} is the inverse link function

Splines: the essential building blocks of a GAM

Splines are functions composed of simpler functions called "basis functions"

Each function gets a coefficient to fit it to the curve, and then the basis functions with coefficients add together to get a spline.

$$f_j(x_j) = \sum_{k=1}^{K} \beta_{j,k} b_{j,k}(x_j).$$

Pedersen et al. (2019)

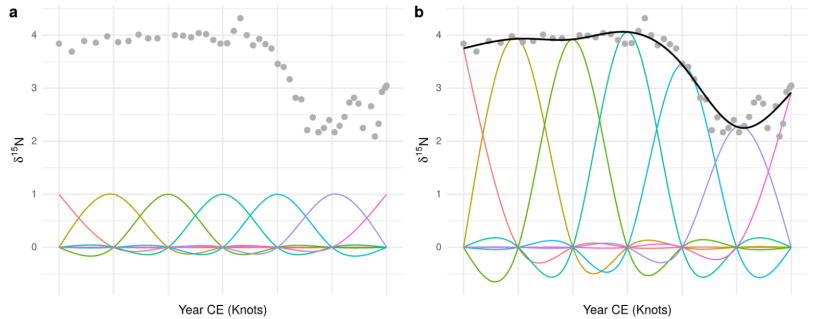
Each smoother f_i is represented by a sum of K simpler, fixed *basis functions* ($b_{j,k}$) multiplied by corresponding coefficients ($B_{j,k}$), which need to be estimated

K: determines the maximum complexity of each smoother

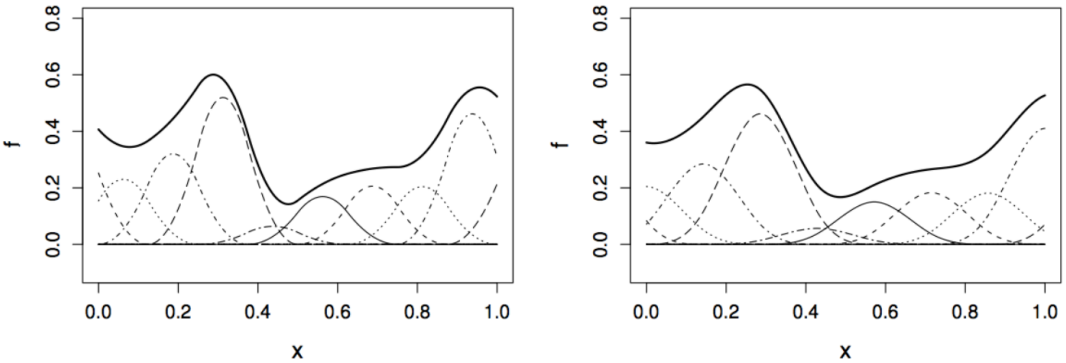
There are many different basis functions you can use, and basis functions can have 1, 2, or more dimensions.

Basis functions

Gavin Simpson York University 2017 Presentation



Noam Ross and Gavin Simpson ESA 2018 GAM Course



How do decide amount of wiggliness?

Need to optimize degree of wiggliness when fitting a GAM to best fit your data while avoiding fitting noise

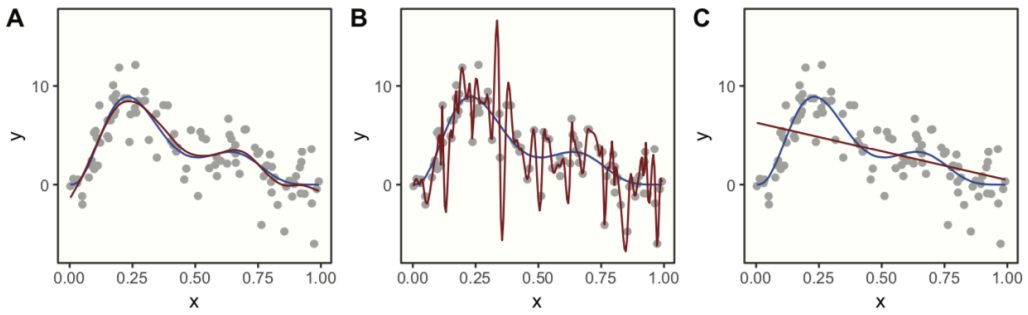


Figure 2 Effect of different choices of smoothing parameter (λ) on the shape of the resulting smoother (red lines). (A) λ estimated using REML; (B) λ set to zero (no smoothing); (C) λ is set to a very large value. The blue line in each panel is the known model used to simulate the data.

Full-size DOI: 10.7717/peerj.6876/fig-2

Pedersen et al. (2019)

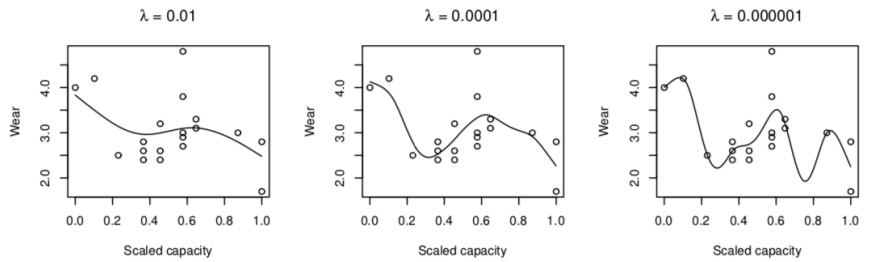


Figure 3.8 *Penalized regression spline fits to the engine wear versus capacity data using three different values for the smoothing parameter.*

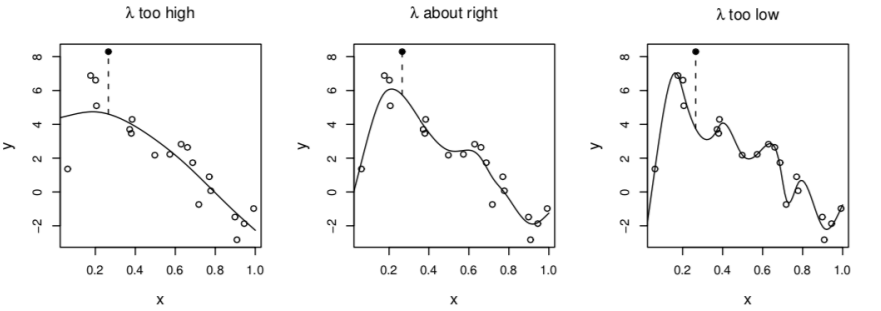


Figure 3.9 Illustration of the principle behind cross validation. In this case the fifth datum (•) has been omitted from fitting and the continuous line shows a penalized regression spline fitted to the remaining data (◦). When the smoothing parameter is too high the spline fits many of the data poorly and does no better with the missing point. When λ is too low the spline fits the noise as well as the signal and the extra variability that this induces causes it to predict the missing datum rather poorly. For the intermediate λ the spline is fitting the underlying signal quite well, but smoothing through the noise: as a result the missing datum is reasonably well predicted. Cross validation leaves out each datum from the data in turn and considers the average ability of models fitted to the remaining data to predict the left out datum.

Wood (2017)

How do decide amount of wiggliness?

$$L_p = \log(\text{Likelihood}) - \lambda W$$

W measures **wigglyness**

Gavin Simpson York University 2017 Presentation

(log) likelihood measures closeness to the data

We use a **smoothing parameter** λ to define the trade-off, to find the spline coefficients B_k that maximize the **penalized** log-likelihood

ƛ (lambda): smoothing parameter.

If ƛ is set to zero, then no smoothing is used (usually leading to overfit data). If ƛ is set to a very large value, the penalty removes all terms that have any wiggliness, resulting in a straight line.

mgcv will select ƛ for you. Best option is to use REML (restricted maximum likelihood) to select ƛ.

Setting K

$$f_j(x_j) = \sum_{k=1}^{K} \beta_{j,k} b_{j,k}(x_j).$$

Pedersen et al. (2019)

Each smoother f_i is represented by a sum of K simpler, fixed *basis functions* ($b_{j,k}$) multiplied by corresponding coefficients ($B_{j,k}$), which need to be estimated

K: determines the maximum complexity of each smoother (can be thought of as the number of basis functions that can make up the curve)

K must be set large enough that the ƛ penalty does the rest

Bigger K increases computational costs

In mgcv, default K values are arbitrary

Comparing a GAM to a linear model (LM) or generalized linear model (LM)

Linear Model:

```
lm(y ~ x1 + x2, data = data)
```

$$y_i = \beta_0 + \sum_j \beta_j x_{ji} + \epsilon_i$$

Generalized Linear Model:

```
glm(y ~ x1 + x2, data = data, family = binomial)
```

Generalized Additive Model (mgcv):

```
gam(y ~ x1 + s(x2),          #model formula
    data = data,             #your data
    family = gaussian,       #define data distribution
    method = "REML")         #specify how mgcv should pick lambda (λ)
```

$$y_i = \beta_0 + \sum_j s_j(x_{ji}) + \epsilon_i$$

Specifying Spline Parameters for the GAM

```
gam(y~s(x1, bs = "tp", k = 10, ...))
```

bs = specifies the basis function ("tp" – thin-plate is the default), k specifies the maximum number of basis functions (REML will decrease number of basis functions to optimize wiggliness)

`family` arguments usable with `gam`

Simon Wood
Presentation, mgcv
GAMs in R

- `gaussian` (default) is useful for real valued response data.
- `Gamma` is useful for strictly positive real valued data. The default link is only useful in some waiting time applications, and the log link is more often used.
- `poisson` is useful when the response is count data of some sort.
- `binomial` is used most often for binary (logistic) regression, but is applicable to any response that is the number of successes from a known number of trials.
- `inverse.gaussian` is for strictly positive real response variables: useful for various 'time to event' data.
- `quasi` does not define a full distribution, but allows inference when only the mean variance relationship can be well approximated. `quasipoisson` and `quasibinomial` are special cases. Not useable with likelihood based smoothness selection.
- `Tweedie` is an alternative to quasi when $\text{var}(y) = \phi\mu^p$, $1 < p < 2$, and a full distribution is required (for a non-negative real response).
- `negbin` is useful for overdispersed count data, but computation is slow.

Considering Interactions Between Covariates with GAMs

Two Non-Interacting Covariates
y ~ s(x1) + s(x2)

Two Interacting Covariates with Similar Wiggliness (same Lambda) (may be appropriate for lat,long)
y ~ s(x1, x2)

Two Interacting Covariates with Different Wiggliness (Different Lambda)
y ~ te(x1, x2) #uses a tensor spline

Two Interacting Covariates that you want to look at main effects and interaction separately
y ~ te(x1) + te(x2) + ti(x1,x2) #ti is a tensor interaction

2D GAMs are good for representing things in space

gam.check() plots

`gam.check()` creates 4 plots:

1. Quantile-quantile plots of residuals. If the model is right, should follow 1-1 line
2. Histogram of residuals
3. Residuals vs. linear predictor
4. Observed vs. fitted values

`gam.check()` uses deviance residuals by default

plot.gam() produces approximate 95% confidence intervals (at +/- 2 SEs)
→ use seWithMean = TRUE call in plot.gam()

Let's move to Pedersen et al. (2019)

PeerJ

Hierarchical generalized additive models in ecology: an introduction with mgcv

Eric J. Pedersen[1,2], David L. Miller[3,4], Gavin L. Simpson[5,6] and Noam Ross[7]

[1] Northwest Atlantic Fisheries Center, Fisheries and Oceans Canada, St. John's, NL, Canada
[2] Department of Biology, Memorial University of Newfoundland, St. John's, NL, Canada
[3] Centre for Research into Ecological and Environmental Modelling, University of St Andrews, St Andrews, UK
[4] School of Mathematics and Statistics, University of St Andrews, St Andrews, Scotland, UK
[5] Institute of Environmental Change and Society, University of Regina, Regina, SK, Canada
[6] Department of Biology, University of Regina, Regina, SK, Canada
[7] EcoHealth Alliance, New York, NY, USA

ABSTRACT

In this paper, we discuss an extension to two popular approaches to modeling complex structures in ecological data: the generalized additive model (GAM) and the hierarchical model (HGLM). The hierarchical GAM (HGAM), allows modeling of nonlinear functional relationships between covariates and outcomes where the shape of the function itself varies between different grouping levels. We describe the theoretical connection between HGAMs, HGLMs, and GAMs, explain how to model different assumptions about the degree of intergroup variability in functional response, and show how HGAMs can be readily fitted using existing GAM software, the **mgcv** package in R. We also discuss computational and statistical issues with fitting these models, and demonstrate how to fit HGAMs on example data. All code and data used to generate this paper are available at: github.com/eric-pedersen/mixed-effect-gams.

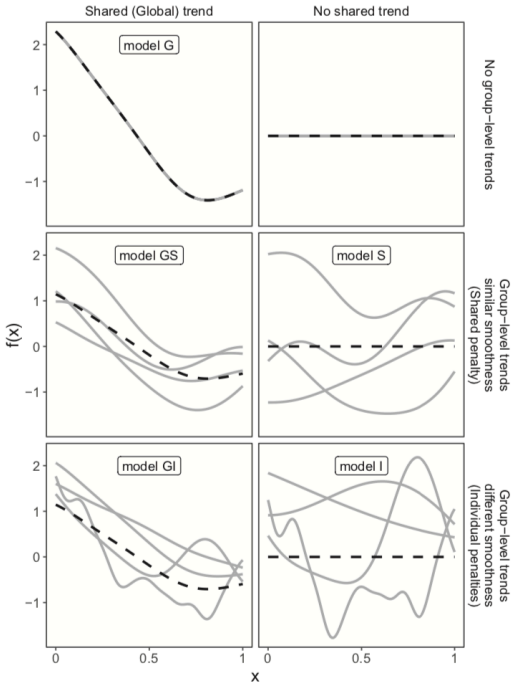


Figure 4 Alternate types of functional variation $f(x)$ that can be fitted with HGAMs. The dashed line indicates the average function value for all groups, and each solid line indicates the functional value at a given predictor value for an individual group level. The null model (of no functional relationship between the covariate and outcome, top right), is not explicitly assigned a model name.

Full-size DOI: 10.7717/peerj.6876/fig-4

Hierarchical GAMs == GAMs that allow you to model intergroup variability

3 Questions to Consider:

1. Should each group have its own smoother, or will a common smoother suffice?
2. Do all of the group-specific smoothers have the same wiggliness, or should each group have its own smoothing parameter?
3. Will the smoothers for each group have a similar shape to one another—a shared global smoother?

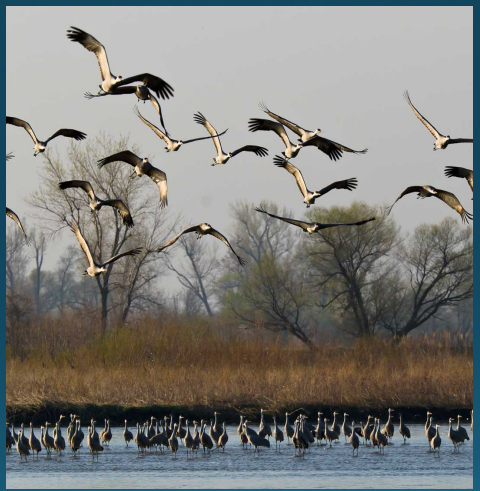
Different possibilities for groups

3 Example Datasets:

CO_2 Uptake in Grasses

Credit: Anne Ebeling, FSU
(Not from actual experiment)

Bird Movement Along a Migration Corridor



Credit: US Fish and Wildlife
(Not from actual experiment)

Lake Zooplankton Community

(Not from actual experiment)

Let’s look at some code

Model G: A single common smoother for all groups or observations

```
CO2_modG <-
gam(log(uptake) ~ s(log(conc), k=5, bs="tp") +  s(Plant_uo, k=12, bs="re"),
data=CO2, method="REML", family="gaussian")

bird_modG <-
gam(count ~ te(week, latitude, bs=c("cc", "tp"), k=c(10, 10)),
data=bird_move, method="REML", family="poisson",
knots=list(week=c(0, 52)))
```

Model GS: A global smoother plus group-level smoothers that have the same wiggliness

```
CO2_modGS <-
gam(log(uptake) ~ s(log(conc), k=5, m=2) +
s(log(conc), Plant_uo, k=5,  bs="fs", m=2),
data=CO2, method="REML")

bird_modGS <-
gam(count ~ te(week, latitude, bs=c("cc", "tp"), k=c(10, 10), m=2) +
t2(week, latitude, species, bs=c("cc", "tp", "re"), k=c(10, 10, 6), m=2, full=TRUE),
data=bird_move, method="REML", family="poisson",
knots=list(week=c(0, 52)))
```

Model GI: A global smoother plus group-level smoothers with differing wiggliness

```
CO2_modGI <- gam(log(uptake) ~
s(log(conc), k=5, m=2, bs="tp") +
s(log(conc), by=Plant_uo, k=5, m=1, bs="tp") +
 s(Plant_uo, bs="re", k=12),
data=CO2, method="REML")

bird_modGI <- gam(count ~ species +
 te(week, latitude, bs=c("cc", "tp"), k=c(10, 10), m=2) +
te(week, latitude, by=species, bs= c("cc", "tp"), k=c(10, 10), m=1),
data=bird_move, method="REML", family="poisson",
knots=list(week=c(0, 52)))
```

Model S: Group-specific smoothers without a global smoother but all smoothers have similar wiggliness

```
CO2_modS <- gam(log(uptake) ~
s(log(conc), Plant_uo, k=5, bs="fs", m=2),
data=CO2, method="REML")

bird_modS <- gam(count ~
t2(week, latitude, species, bs=c("cc", "tp", "re"),  k=c(10, 10, 6), m=2, full=TRUE),
data=bird_move, method="REML", family="poisson",
knots=list(week=c(0, 52)))
```

Model I: Group-specific smoothers without a global smoother and with differing wiggliness

```
CO2_modI <- gam(log(uptake) ~
s(log(conc), by=Plant_uo, k=5, bs="tp", m=2) +
s(Plant_uo, bs="re", k=12),
data=CO2, method="REML")

bird_modI <- gam(count ~
species +
te(week, latitude, by=species, bs=c("cc", "tp"), k=c(10, 10), m=2),
data=bird_move, method="REML", family="poisson",
knots=list(week=c(0, 52)))
```

Summary

GAMs give us a powerful framework to model flexible nonlinear relationship
GAMs are GLMs plus extra wiggles (HGAMs are HGLMs plus extra wiggles)
GAMs use basis functions to make smooths and use a penalty (ƛ) to tradeoff wiggliness/generality

You make these choices when fitting a GAM:

- k=n (number of knots, i.e. maximum wiggliness)
- family=n (what distribution does your data belong to)
- bs=n (what type of smoother to use)

You need to make sure your smooths are wiggly enough by increasing and testing k=n
Always check your model (gam.check())

Additional Resources:

Gallo et al. (accepted) Dissolved oxygen and temperature best predict deep-sea fish community structure in the Gulf of California with implications for climate change. *MEPS*
Keller et al. (2017) Species-specific responses of demersal fishes to near-bottom oxygen levels within the California Current large marine ecosystem. *MEPS*
Book: Wood (2017) Generalized Additive Models: An Introduction with R
Kim Larsen (2015): GAM: The Predictive Modeling Silver Bullet: https://multithreaded.stitchfix.com/blog/2015/07/30/gam/
Gavin Simpson York University 2017 Presentation Slides: https://github.com/gavinsimpson/gams-yorku-canada-150
Noam Ross and Gavin Simpson ESA 2018 Course on GAMs: https://noamross.github.io/mgcv-esa-2018/
Noam Ross Presentation: Nonlinear models in R: The Wonderful World of mgcv: https://www.youtube.com/watch?v=q4_t8jXcQgc
Simon Wood Slides on mgcv in R: https://people.maths.bris.ac.uk/~sw15190/mgcv/tampere/mgcv.pdf
Gavin Simpson's GAM-heavy blog: www.fromthebottomoftheheap.net