Generalized Additive Models!

Gavin Fay
slides from Megan Winton (mwinton@umassd.edu)
MAR 536 Biological Statistics II
February 23 2023

Additional acknowledgements: Steve Cadrin & Jim Thorson

Generalized linear models: Quick review

Steps for fitting a GLM:

1. **Specify distribution for response variable**
   - **What we want to predict**
2. **Specify link function**
   - **Remember the link function calculates the expected value of the response variable given the linear predictor**
3. **Specify linear predictor**
   - **What we think influences what we want to predict**

Example:

What is the relationship between counts of species *i* and covariate x?

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \boldsymbol{\beta}\mathbf{x}_i$$

How do you choose which distribution?

Step 1: Is your response variable DISCRETE or CONTINUOUS?

Common distributions for response variables

Step 1: If DISCRETE

Name	Notation	Domain	Range
Bernoulli	$B \sim Bernoulli(p)$	$0 \leq p \leq 1$	$B = \{0, 1\}$
Binomial	$N \sim Binomial(p, n)$	$0 \leq p \leq 1$	$N = \{0, 1, \ldots, n\}$
Poisson	$N \sim Poisson(\lambda)$	$\lambda > 0$	$N = \{0, 1, \ldots, \infty\}$
Negative binomial	$N \sim NegativeBinomial(\lambda, \theta)$	$\lambda > 0$ $\theta > 0$	$N = \{0, 1, \ldots, \infty\}$

Common distributions for response variables

Step 1: If DISCRETE

Name	Notation	Domain	Range
Bernoulli	$B \sim Bernoulli(p)$	$0 \le p \le 1$	B = {0, 1}
Binomial	$N \sim Binomial(p, n)$	$0 \le p \le 1$	N = {0, 1, …, n}
Poisson	$N \sim Poisson(\lambda)$	$\lambda > 0$	N = {0, 1, …,∞}
Negative binomial	$N \sim NegativeBinomial(\lambda, \theta)$	$\lambda > 0$ $\Theta > 0$	N = {0, 1, …,∞}

Step 1a: What is the range of possible values?

- **0 or 1 -> Bernoulli**
- **Between 0 and N (N is # of trials) -> Binomial**
- **>= 0 Poisson**
- **>= 0 and variance changes with mean -> Negative binomial**

Common distributions for response variables

Step 1: If CONTINUOUS

Name	Notation	Domain	Range
Normal	$Y \sim Normal(\mu, \sigma^2)$	$\sigma^2 > 0$	Unrestricted
Lognormal	$\log(Y) \sim Normal(\mu, \sigma^2)$	$\sigma^2 > 0$	$Y > 0$
Gamma	$Y \sim Gamma(\mu, CV)$	$\mu > 0$ $CV > 0$	$Y > 0$
Beta	$p \sim Beta(\alpha, \beta)$	$\alpha > 0$, $\beta > 0$	$0 < p < 1$

Step 1b: What is the range of possible values?

- **$-\infty$ to $+\infty$ -> Normal**
- **> 0 -> Lognormal or Gamma**
- **> 0 and < 1 -> Beta**

Step 1b: Is there precedent?

Choice of link functions

Step 2: Specify link function based on selected distribution

- Remember that the link function acts like transformation of the

  response variable.
- The link function establishes the connection between the linear predictor and the mean of the distribution.
- There is a 'natural link' associated with each distribution – the canonical link function
  - For our Poisson example:

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \boldsymbol{\beta}\mathbf{x}_i$$

- Canonical link is typically used, but don't neglect alternatives
  - Enter ?family to see options in R

What makes GLMs linear?

Step 3: Specify linear predictor

- **Linear predictor expresses our hypothesis about what influences our response variable**
- **In a GLM, all terms in the linear predictor are linear.**
    - **Expanding on our Poisson example:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + \beta_2 \mathrm{Temp}_i$$

What makes GLMs linear?

Step 3: Specify linear predictor

- **Linear predictor expresses our hypothesis about what influences our response variable**
- **In a GLM, all terms in the linear predictor are linear.**
  - **Expanding on our Poisson example:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + \beta_2 \mathrm{Temp}_i$$

- **If we suspect things are nonlinear, we can include a polynomial:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + \beta_2 \mathrm{Temp}_i + \beta_3 \mathrm{Temp}_i^2$$

What makes GLMs linear?

Step 3: Specify linear predictor

- **Linear predictor expresses our hypothesis about what influences our response variable**
- **In a GLM, all terms in the linear predictor are linear.**
    - **Expanding on our Poisson example:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + \beta_2 \mathrm{Temp}_i$$

- **If we suspect things are nonlinear, we can include a polynomial:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + \beta_2 \mathrm{Temp}_i + \beta_3 \mathrm{Temp}_i{}^2$$

This is still a GLM!

Getting to GAMs

Step 3: Specify linear predictor

- **A GAM includes at least one nonlinear smoothing function or spline.**
- **To make our Poisson example a GAM:**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_1 \mathrm{Time}_i + f(\mathrm{Temp}_i)$$

- **Always use 'mgcv' rather than default gam( )!**
- **There are tons of splines to choose from.**

**** A 'regular' additive model is just a GAM assuming a normal distribution with an identity link function.**

Spline types in package 'mgcv'

Thin plate spline (tp):

- does not use knots
- can be used for multiple covariates (i.e., for interactions)
- computationally expensive

Cubic regression splines (cr):

- uses knots
- can only be used for single covariates
- computationally less expensive

Cyclic cubic regression splines (cc):

- A cr, but has the same start and end point (e.g. for modelling seasonality)

Spline types in package 'mgcv'

Splines with shrinkage: allow for the complete removal of covariates during fitting if they are not needed

- Thin plate spline with shrinkage (ts):
- Cubic regression splines with shrinkage (cs):

Tensor products (te):

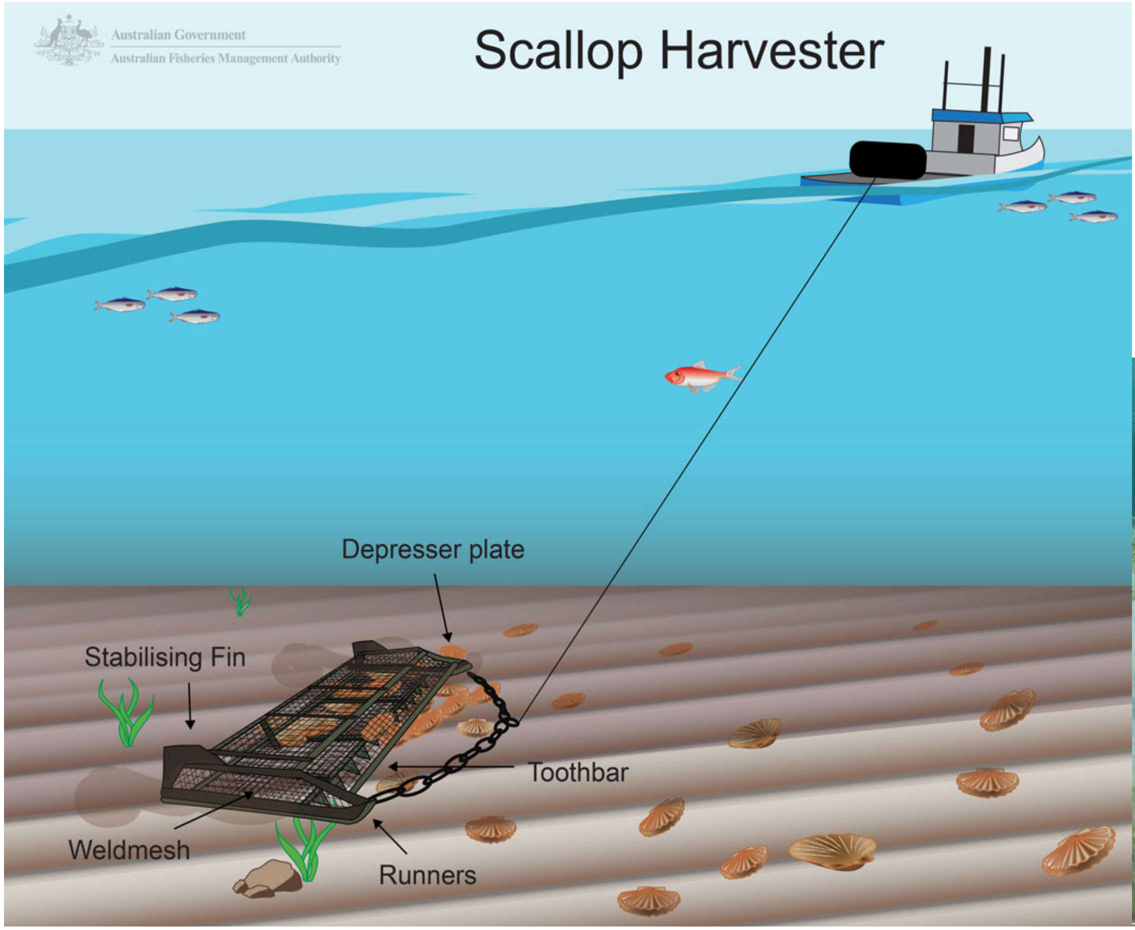
- another alternative if you have multiple covariates with interactions
- Advantage = invariant to relative scaling of covariates

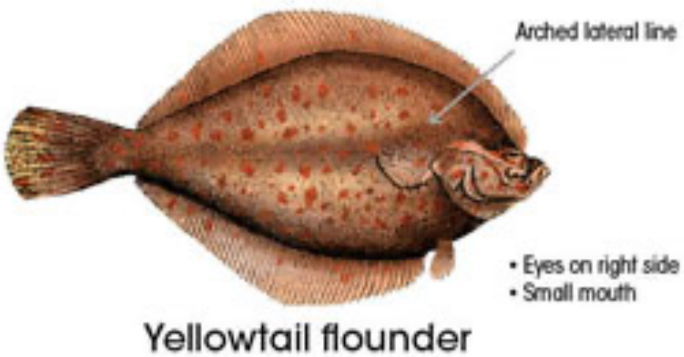
And more!

- **Enter ?smooth.terms after loading the 'mgcv' library in R**

Motivating example

Can we identify seasonal trends in yellowtail flounder bycatch in the sea scallop fishery to inform bycatch mitigation measures?

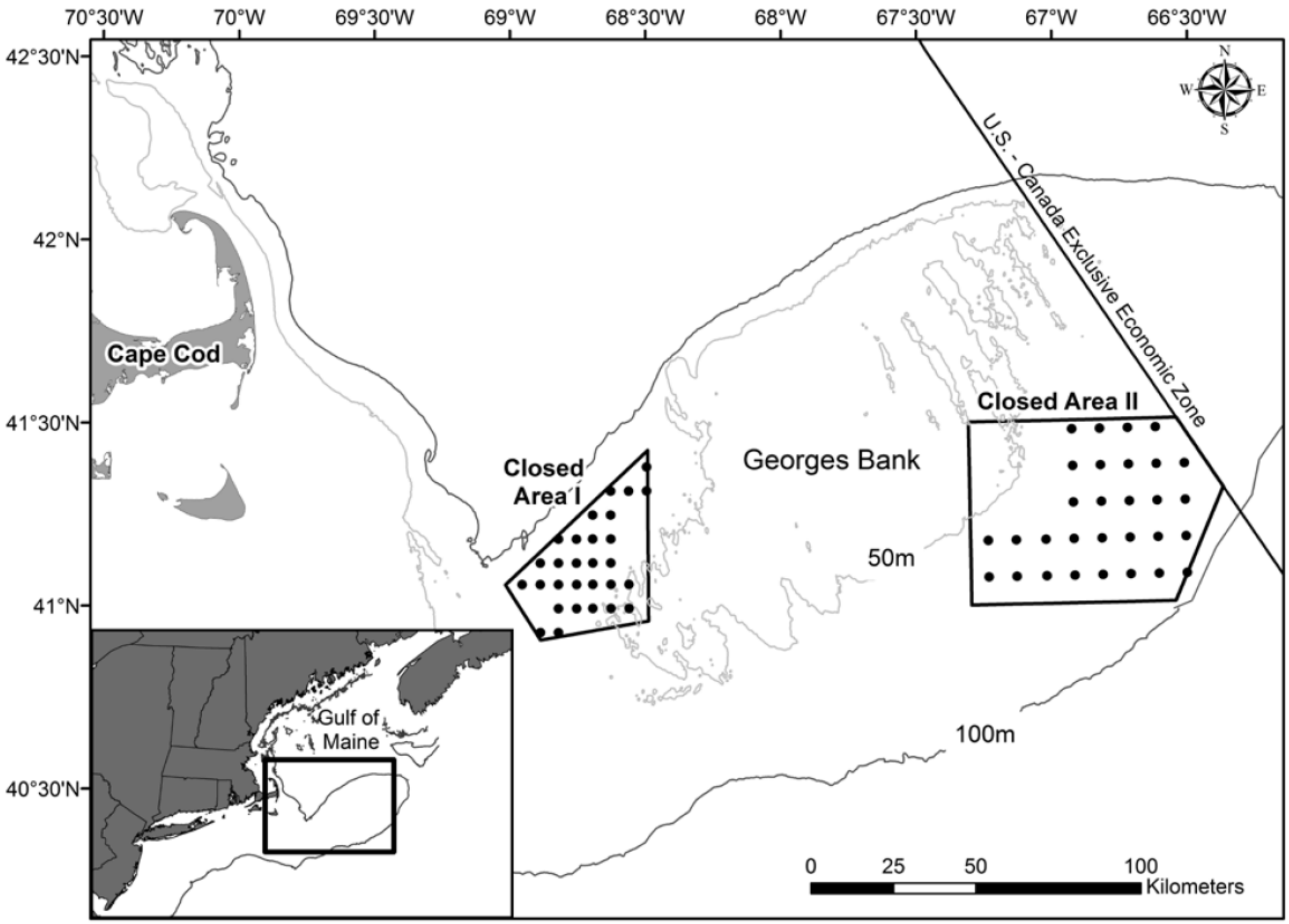




wildlife.state.nh.us/

Motivating example

Coonamessett Farm Foundation's seasonal bycatch survey



Simplest possible model

What is the mean catch of YT per survey tow?

- **# of YT per 30 minute tow**

Remember our GLM fitting steps!

1. **Specify distribution for response variable**
2. **Specify link function**
3. **Specify linear predictor**

Simplest possible model

What is the mean catch of YT per survey tow?

- **# of YT per 30 minute tow**

Remember our GLM fitting steps!

1. **Specify distribution for response variable**
   - **Counts -> Poisson:**

$$c_i \sim Poisson(\lambda_i)$$

2. **Specify link function**
   - **We'll go with the canonical link function -> log link**
     - **(? family in R for others)**
3. **Specify linear predictor**
   - **Intercept only**

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0$$

Fitting in R

Using the glm( ) command:

```
> glm0 = glm(catch~1,data=dat2,family=poisson(link="log"))
```

```
> summary(glm0)

Call:
glm(formula = catch ~ 1, family = poisson(link = "log"), data = dat2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.7625  -2.8709  -1.2612   0.6929  24.2451

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.95702    0.01284   152.4   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 8485.4  on 856  degrees of freedom
Residual deviance: 8485.4  on 856  degrees of freedom
AIC: 10983

Number of Fisher Scoring iterations: 6

> exp(1.957)
[1] 7.078061
> summary(dat2$catch)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.000   4.000   7.078   9.000 143.000
```

Our model:

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0$$

Fitting in R

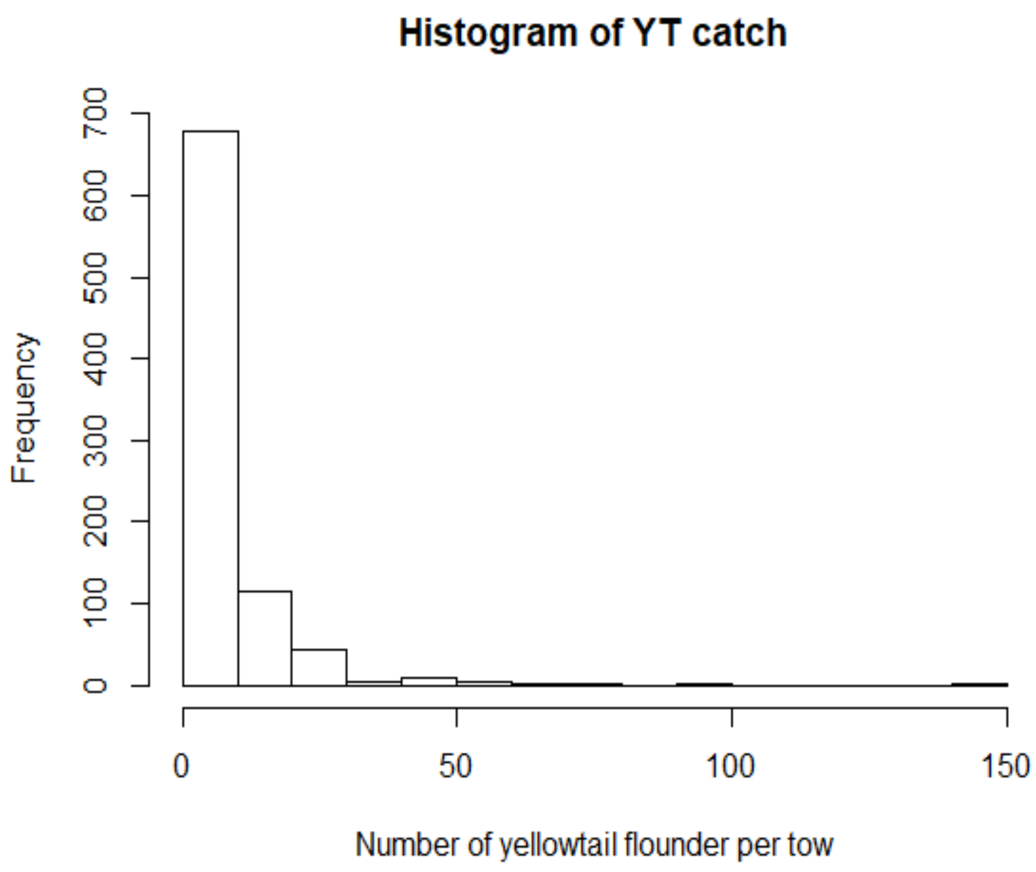
Using the gam( ) command in package 'mgcv':

```
> m0 = gam(catch~1,data=dat2,family=poisson(link="log"))
```

```
> summary(m0)

Family: poisson
Link function: log

Formula:
catch ~ 1

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.95702    0.01284   152.4   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =      0   Deviance explained = -2.14e-13%
UBRE = 8.9036  Scale est. = 1         n = 857
> |
```

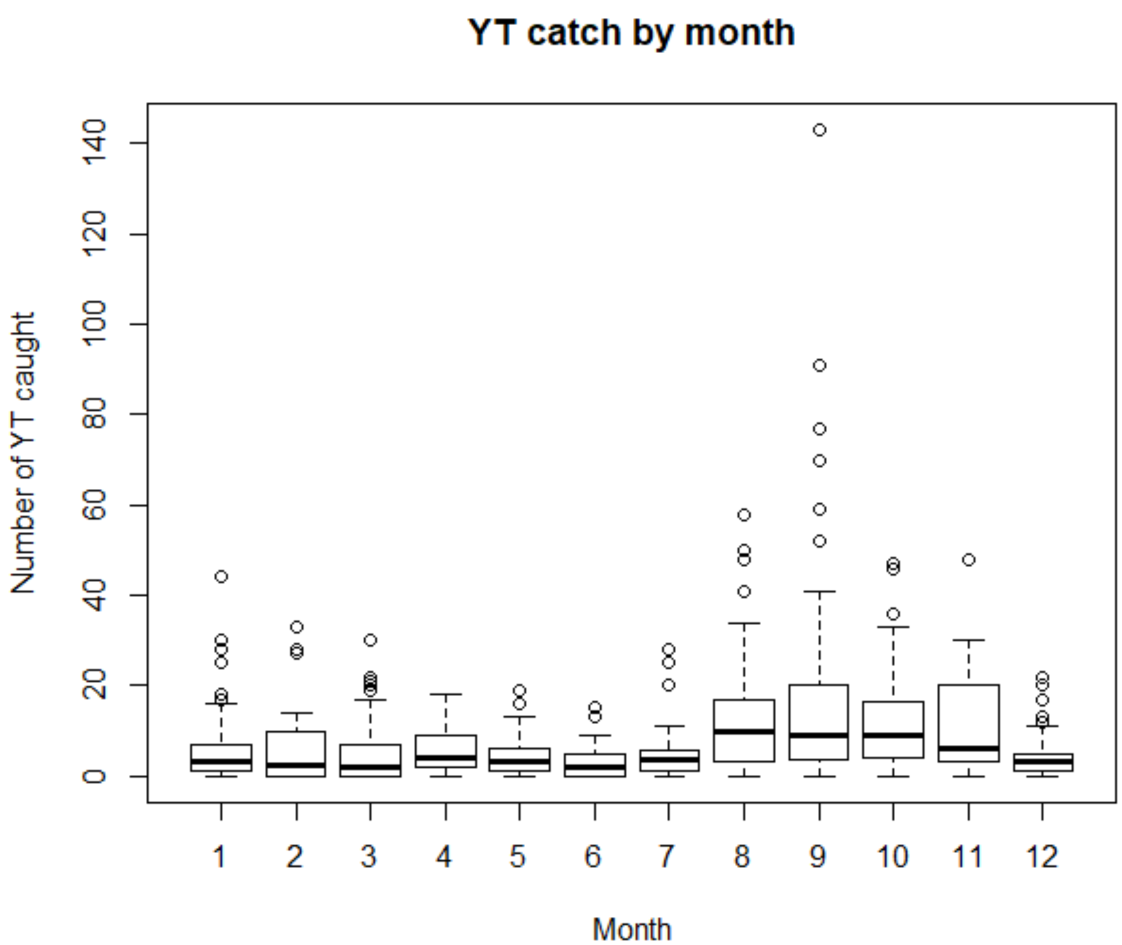
That was pretty boring…

There are very few situations where an intercept-only model will be informative (except as a baseline for model selection).



Specifying the linear predictor

We might expect that bycatch rates vary seasonally due to YT movements or factors impacting their response time (e.g. water temperature).



Specifying the linear predictor

A logical first step might be to include month as a factor (i.e. a categorical variable).

```
> m1 = gam(catch~as.factor(Month),data=dat2,family=poisson(link="log"))
```

```
> summary(m1)

Family: poisson
Link function: log

Formula:
catch ~ as.factor(Month)

Parametric coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.763268   0.043895  40.170  < 2e-16 ***
as.factor(Month)2   0.133852   0.083227   1.608 0.107775
as.factor(Month)3  -0.290487   0.062077  -4.679 2.88e-06 ***
as.factor(Month)4  -0.008156   0.062381  -0.131 0.895974
as.factor(Month)5  -0.284858   0.076986  -3.700 0.000215 ***
as.factor(Month)6  -0.679581   0.075407  -9.012  < 2e-16 ***
as.factor(Month)7  -0.259190   0.075037  -3.454 0.000552 ***
as.factor(Month)8   0.774320   0.057565  13.451  < 2e-16 ***
as.factor(Month)9   0.994294   0.051456  19.323  < 2e-16 ***
as.factor(Month)10  0.749496   0.057449  13.046  < 2e-16 ***
as.factor(Month)11  0.661535   0.069833   9.473  < 2e-16 ***
as.factor(Month)12 -0.382544   0.068703  -5.568 2.58e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  0.145   Deviance explained = 22.3%
UBRE = 6.7241  Scale est. = 1         n = 857
>
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$
$$\log(\lambda_i) = \beta_0 + \beta_{i,Monthi}$$

Interpretation of results

A logical first step might be to include month as a factor (i.e. a categorical variable).

```
> m1 = gam(catch~as.factor(Month),data=dat2,family=poisson(link="log"))
```

```
> summary(m1)

Family: poisson
Link function: log

Formula:
catch ~ as.factor(Month)

Parametric coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.763268   0.043895  40.170  < 2e-16 ***
as.factor(Month)2   0.133852   0.083227   1.608 0.107775
as.factor(Month)3  -0.290487   0.062077  -4.679 2.88e-06 ***
as.factor(Month)4  -0.008156   0.062381  -0.131 0.895974
as.factor(Month)5  -0.284858   0.076986  -3.700 0.000215 ***
as.factor(Month)6  -0.679581   0.075407  -9.012  < 2e-16 ***
as.factor(Month)7  -0.259190   0.075037  -3.454 0.000552 ***
as.factor(Month)8   0.774320   0.057565  13.451  < 2e-16 ***
as.factor(Month)9   0.994294   0.051456  19.323  < 2e-16 ***
as.factor(Month)10  0.749496   0.057449  13.046  < 2e-16 ***
as.factor(Month)11  0.661535   0.069833   9.473  < 2e-16 ***
as.factor(Month)12 -0.382544   0.068703  -5.568 2.58e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  0.145   Deviance explained = 22.3%
UBRE = 6.7241  Scale est. = 1         n = 857
>
```

Interpretation:

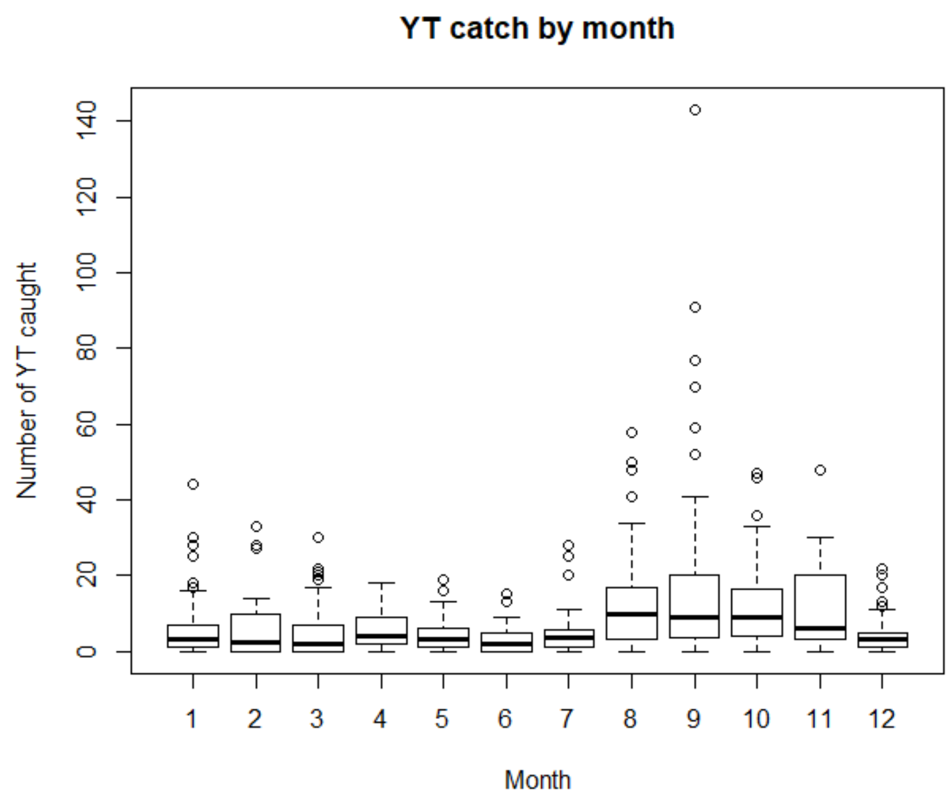
- **To predict value in each month, add coefficient to the intercept**
  - **Intercept corresponds to January**
- **Explained more of the observed variation than the intercept-only model.**

Interpretation of results

A logical first step might be to include month as a factor (i.e. a categorical variable).

```
> m1 = gam(catch~as.factor(Month),data=dat2,family=poisson(link="log"))
```

```
> summary(m1)

Family: poisson
Link function: log

Formula:
catch ~ as.factor(Month)

Parametric coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.763268   0.043895  40.170  < 2e-16 ***
as.factor(Month)2   0.133852   0.083227   1.608 0.107775
as.factor(Month)3  -0.290487   0.062077  -4.679 2.88e-06 ***
as.factor(Month)4  -0.008156   0.062381  -0.131 0.895974
as.factor(Month)5  -0.284858   0.076986  -3.700 0.000215 ***
as.factor(Month)6  -0.679581   0.075407  -9.012  < 2e-16 ***
as.factor(Month)7  -0.259190   0.075037  -3.454 0.000552 ***
as.factor(Month)8   0.774320   0.057565  13.451  < 2e-16 ***
as.factor(Month)9   0.994294   0.051456  19.323  < 2e-16 ***
as.factor(Month)10  0.749496   0.057449  13.046  < 2e-16 ***
as.factor(Month)11  0.661535   0.069833   9.473  < 2e-16 ***
as.factor(Month)12 -0.382544   0.068703  -5.568 2.58e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


R-sq.(adj) =  0.145   Deviance explained = 22.3%
UBRE = 6.7241  Scale est. = 1         n = 857
>
```



Specifying a continuous, linear seasonal effect

Based on the boxplot, it might make sense to model YT catch as a linear function of month.

```
> m2 = gam(catch~Month,data=dat2,family=poisson(link="log"))
```

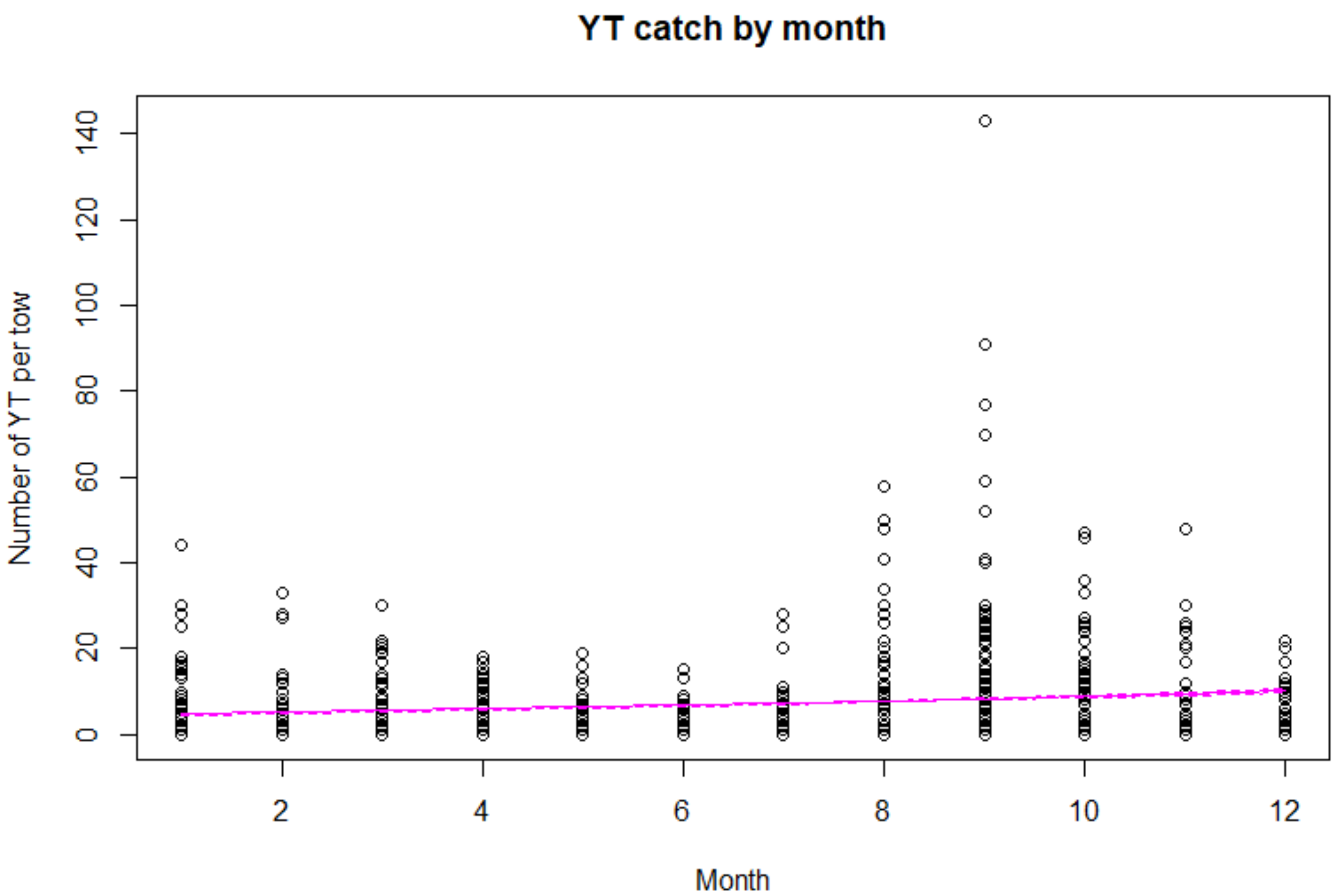
```
> summary(m2)

Family: poisson
Link function: log

Formula:
catch ~ Month

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 1.504082   0.029246   51.43   <2e-16 ***
Month       0.068112   0.003724   18.29   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


R-sq.(adj) =  0.0211   Deviance explained = 3.96%
UBRE =  8.514  Scale est. = 1         n = 857
>
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

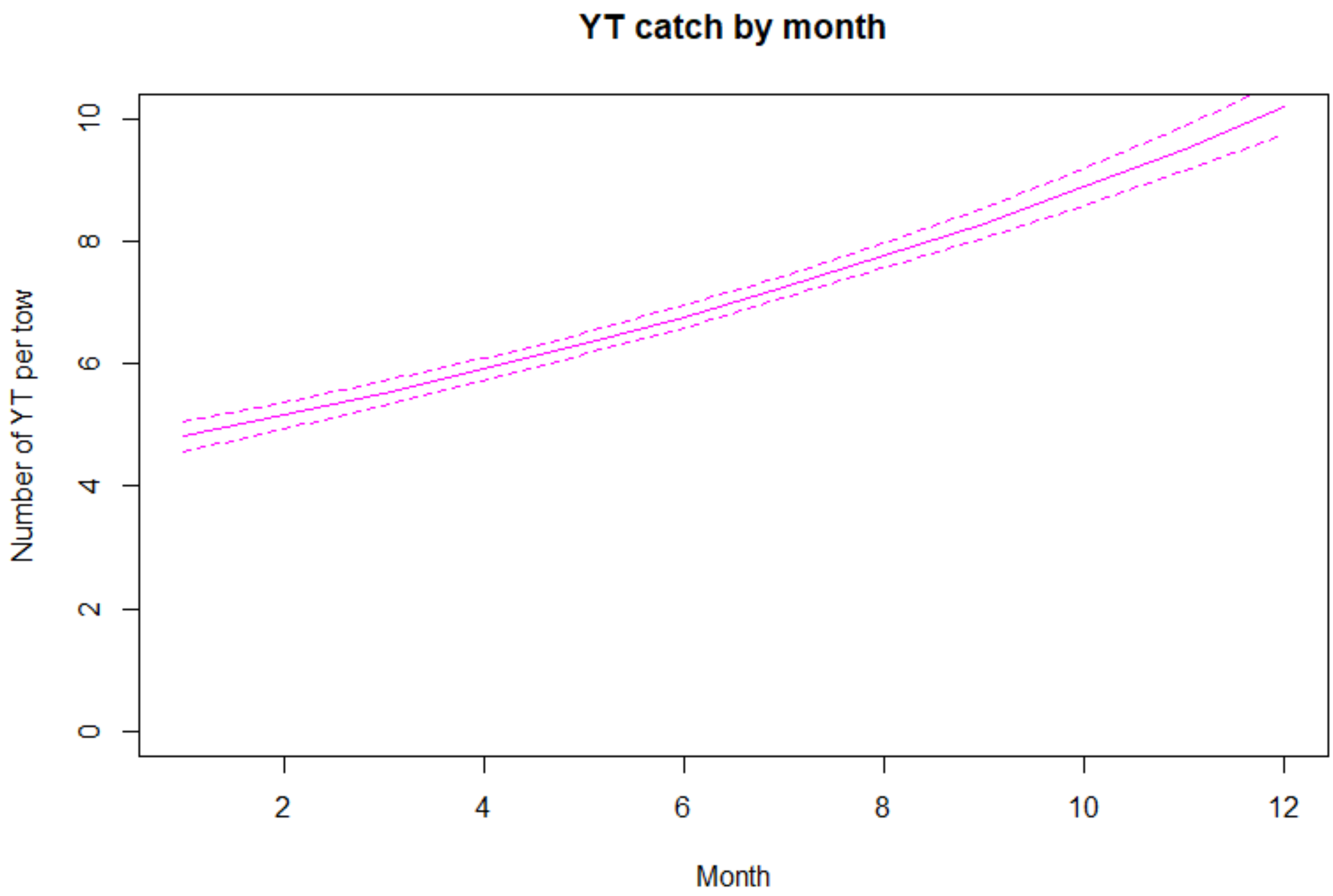
$$\log(\lambda_i) = \beta_0 + \beta_1 \text{Month}_i$$

Predicted relationship



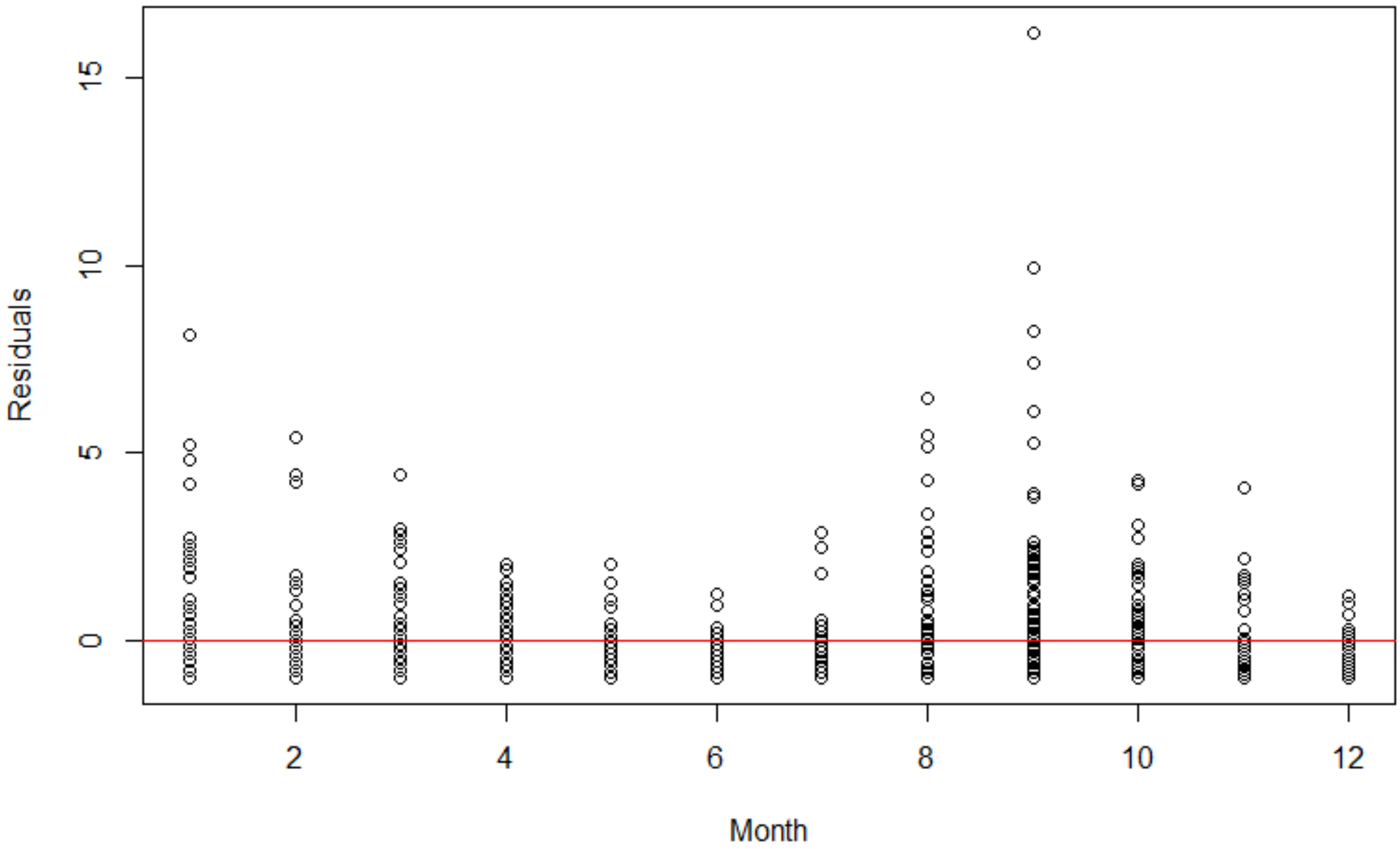
Predicted relationship

Zoomed in (and not plotting points for tows):



Predicted relationship

Not surprisingly, residual plot doesn't look great, either:



What do we do when things aren't linear?

Fit a polynomial: Many animals exhibit seasonal cycles – maybe a 3rd order polynomial will do?

```
> m3 = gam(catch~poly(Month,3),data=dat2,family=poisson(link="log"))
```

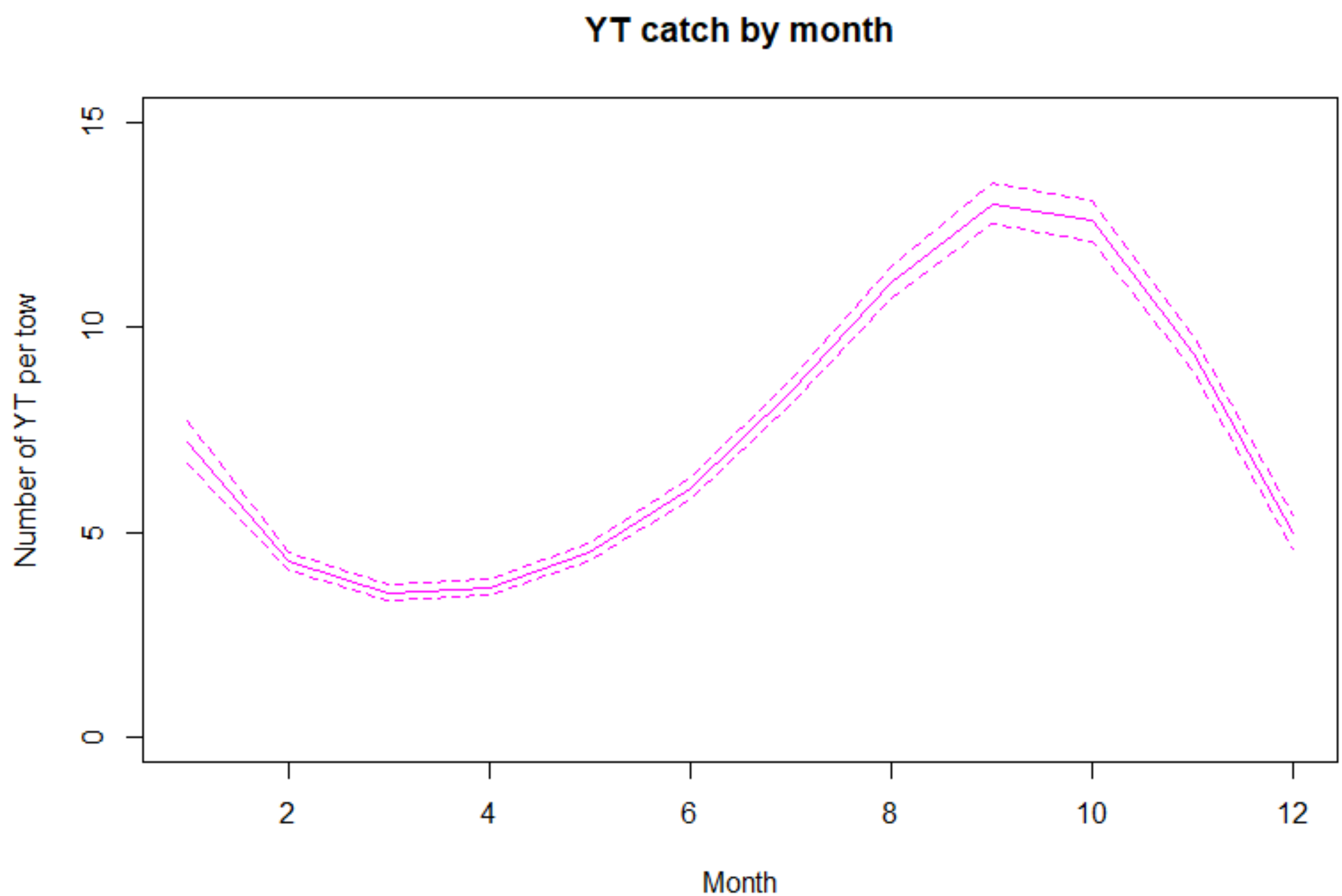
```
> summary(m3)

Family: poisson
Link function: log

Formula:
catch ~ poly(Month, 3)

Parametric coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       1.84683    0.01424 129.688  < 2e-16 ***
poly(Month, 3)1   6.74500    0.39915  16.899  < 2e-16 ***
poly(Month, 3)2  -1.78638    0.39403  -4.534 5.8e-06 ***
poly(Month, 3)3 -11.75134    0.39744 -29.568  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


R-sq.(adj) =  0.116   Deviance explained = 15.5%
UBRE = 7.3736  Scale est. = 1         n = 857
> |
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + \beta_1 \text{Month}_i + \beta_2 \text{Month}_i^2 + \beta_3 \text{Month}_i^3$$

Predicted relationship



What if the nonlinear relationship is more complex?

Fit a generalized additive model: s( ) notation

```
> m4 = gam(catch~s(Month),data=dat2,family=poisson(link="log"))
```

```
> summary(m4)

Family: poisson
Link function: log

Formula:
catch ~ s(Month)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.80825    0.01469   123.1   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Approximate significance of smooth terms:
           edf Ref.df Chi.sq p-value
s(Month) 8.865  8.994   1861  <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) =  0.146   Deviance explained = 22.1%
UBRE = 6.7365  Scale est. = 1         n = 857
> |
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{Month}_i)$$

gam( ) optimizes smoothness selection for you

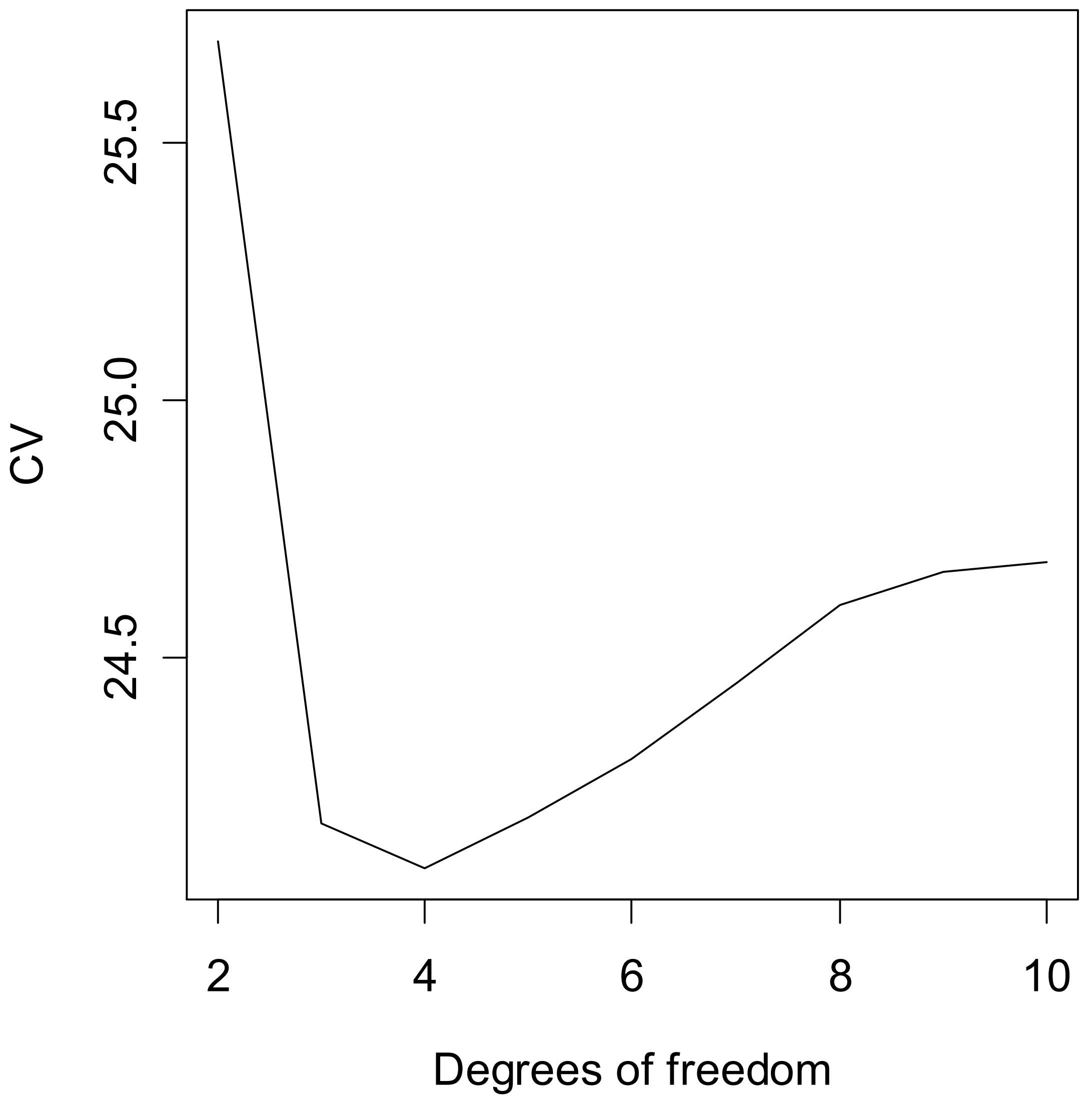
- **Automatically determines the degrees of freedom for every section of the smoothing function**

$$CV(\lambda) = \frac{1}{n}\sum_{1=1}^{n}\left(\left(Y_i - \widehat{f_\lambda^{-i}}(X_i)\right)\right)^2$$

* The -*i* means the ith observation was removed & λ is the amount of smoothing

- **GCV is generalized cross validation & is a modified version of cross validation that finds an optimal parameter value based on cross validation.**
- **A GCV (or UBRE) score will be included in the output**
  - **The lower the value, the better the fit (similar to AIC).**
- **Can also fit via maximum likelihood by specifying method="ML" (recommended)**

gam( ) optimizes smoothness selection for you



Interpreting gam( ) output

Fit a generalized additive model: s( ) notation

```
> m4 = gam(catch~s(Month),data=dat2,family=poisson(link="log"))
```

```
> summary(m4)

Family: poisson
Link function: log

Formula:
catch ~ s(Month)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.80825    0.01469   123.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
           edf Ref.df Chi.sq p-value
s(Month) 8.865  8.994   1861  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.146   Deviance explained = 22.1%
UBRE = 6.7365  Scale est. = 1         n = 857
>
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{Month}_i)$$

Interpreting gam( ) output

Fit a generalized additive model: s( ) notation

```
> m4 = gam(catch~s(Month),data=dat2,family=poisson(link="log"))
```

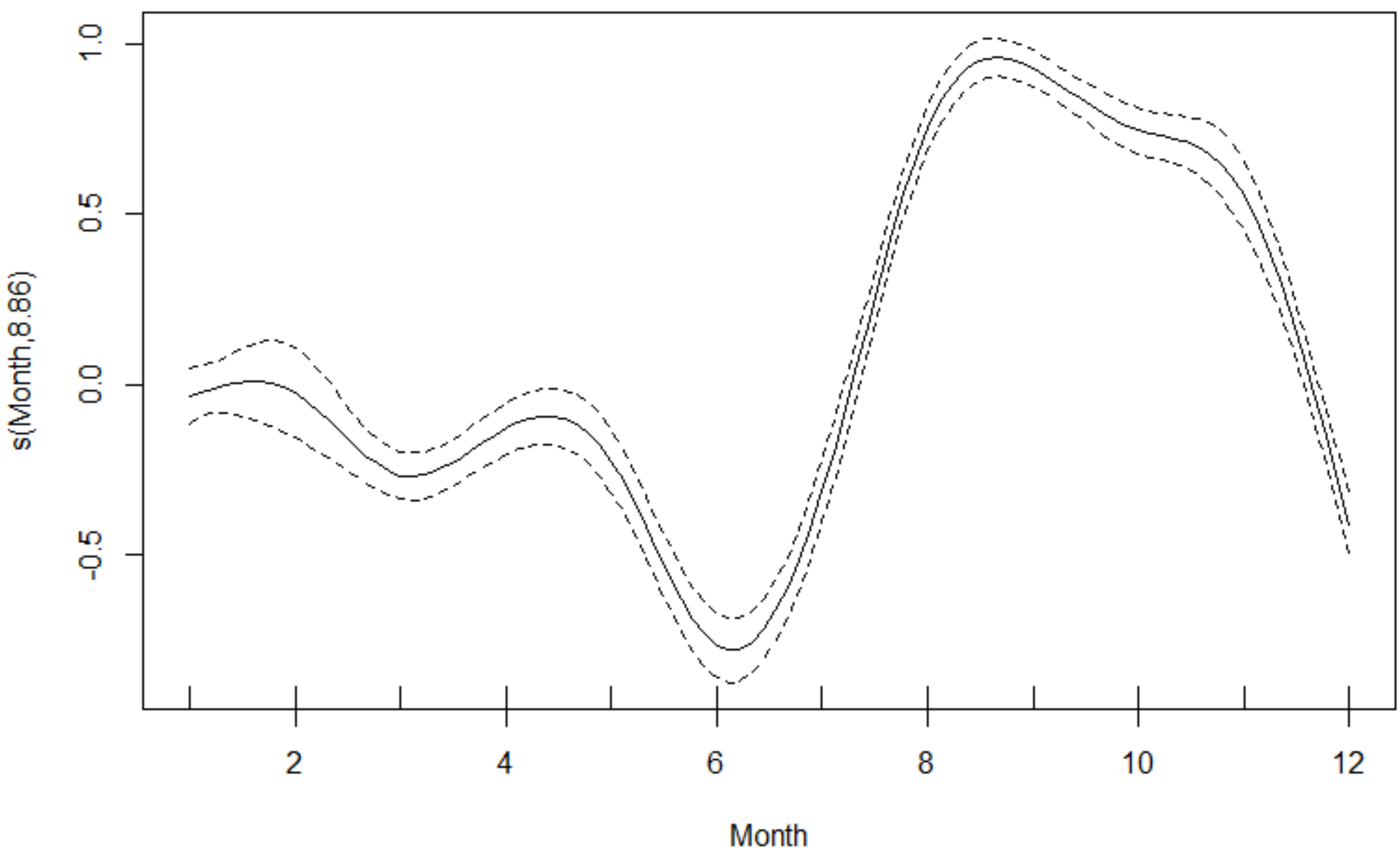
```
> summary(m4)

Family: poisson
Link function: log

Formula:
catch ~ s(Month)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.80825    0.01469   123.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
           edf Ref.df Chi.sq p-value
s(Month) 8.865  8.994   1861  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.146   Deviance explained = 22.1%
UBRE = 6.7365  Scale est. = 1         n = 857
> |
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{ Month}_i\ )$$

Predicted relationship



Spline selection: you've got options!

Fit a generalized additive model: s(, bs =) notation

```
> m5 = gam(catch~s(Month,bs='cc'),data=dat2,family=poisson(link="log"))
```

```
> summary(m3)

Family: poisson
Link function: log

Formula:
catch ~ Month + I(Month^2)

Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.129041   0.054723  20.632   <2e-16 ***
Month        0.212075   0.017542  12.090   <2e-16 ***
I(Month^2)  -0.010490   0.001243  -8.439   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  0.0332   Deviance explained = 4.82%
UBRE = 8.4306  Scale est. = 1         n = 857
>
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{ Month}_i\ )$$

Spline selection: you've got options!

Fit a generalized additive model: s(, bs =) notation

```
> m5 = gam(catch~s(Month,bs='cc'),data=dat2,family=poisson(link="log"))
```

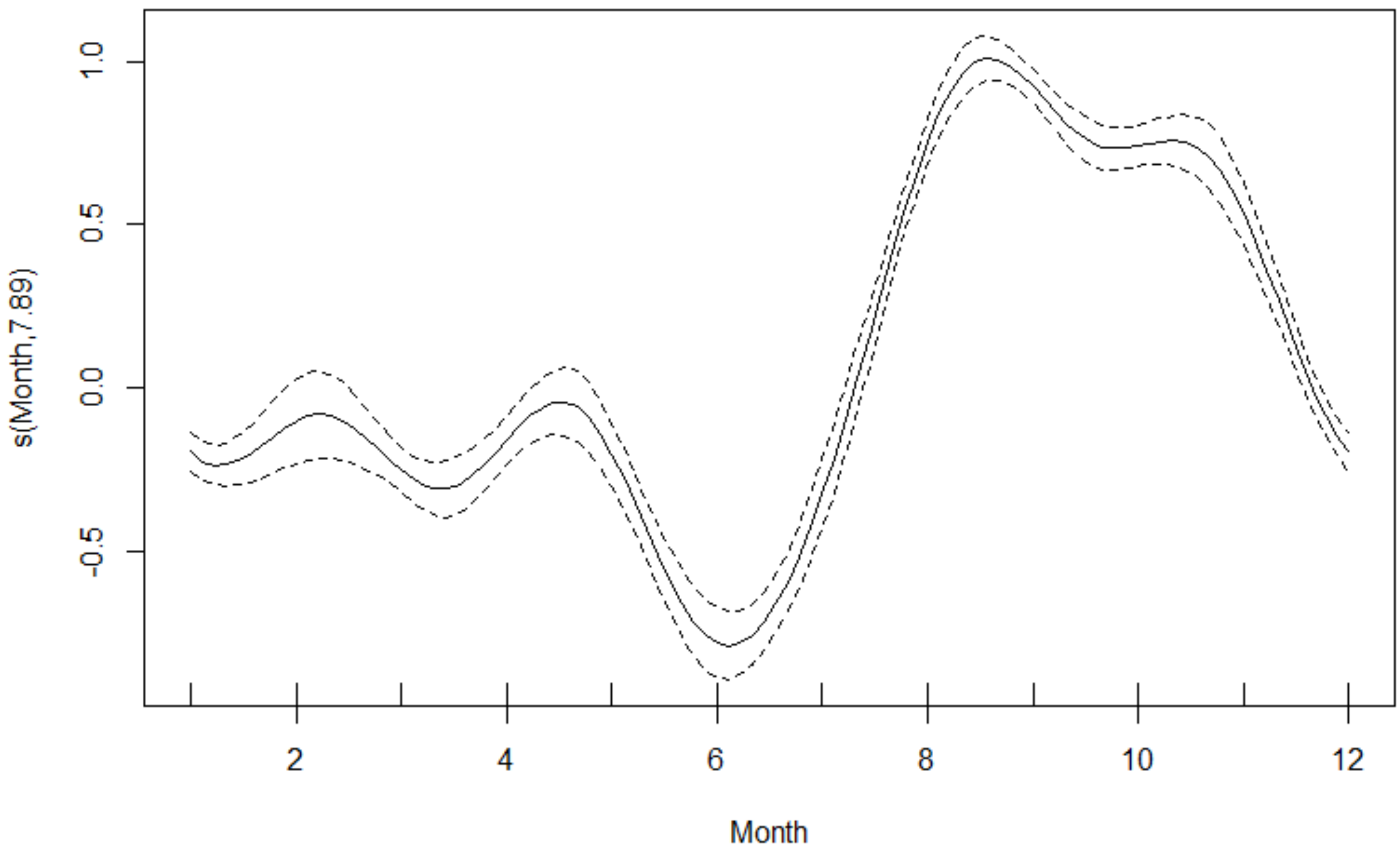
```
> summary(m5)

Family: poisson
Link function: log

Formula:
catch ~ s(Month, bs = "cc")

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.81270    0.01463   123.9   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
           edf Ref.df Chi.sq p-value
s(Month) 7.892      8   1837  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.145   Deviance explained = 21.6%
UBRE = 6.7843  Scale est. = 1         n = 857
> 
```

Our model is still:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{ Month}_i\ )$$

Predicted relationship



Including interaction terms in a smoother

Fit a generalized additive model: s(, by =) notation

```
> m6 = gam(catch~s(Month,by=Lat),data=dat2,family=poisson(link="log"))
```

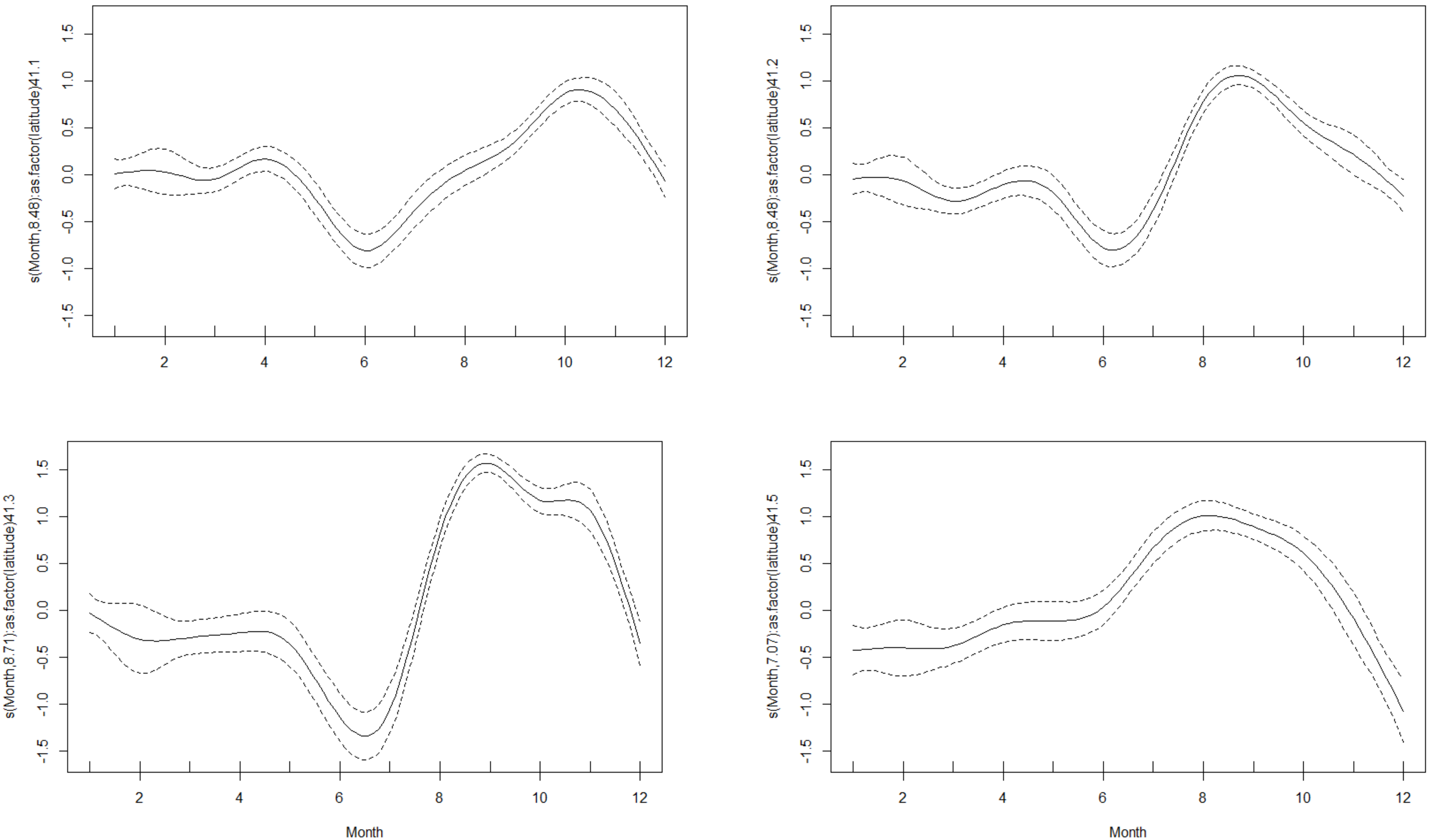
```
> summary(m6)

Family: poisson
Link function: log

Formula:
catch ~ s(Month, by = as.factor(latitude))

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.76400    0.01525   115.7   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Approximate significance of smooth terms:
                                   edf Ref.df Chi.sq p-value
s(Month):as.factor(latitude)41.1 8.477  8.921  288.7  <2e-16 ***
s(Month):as.factor(latitude)41.2 8.483  8.919  589.9  <2e-16 ***
s(Month):as.factor(latitude)41.3 8.712  8.973 1370.9  <2e-16 ***
s(Month):as.factor(latitude)41.4 8.902  8.997  651.5  <2e-16 ***
s(Month):as.factor(latitude)41.5 7.072  8.066  330.2  <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) =  0.208   Deviance explained = 29.8%
UBRE = 6.0504  Scale est. = 1         n = 857
> |
```

Our model is now:

$$c_i \sim Poisson(\lambda_i)$$

$$\log(\lambda_i) = \beta_0 + f(\text{Month}_i * Latitude_i)$$

Predicted relationships



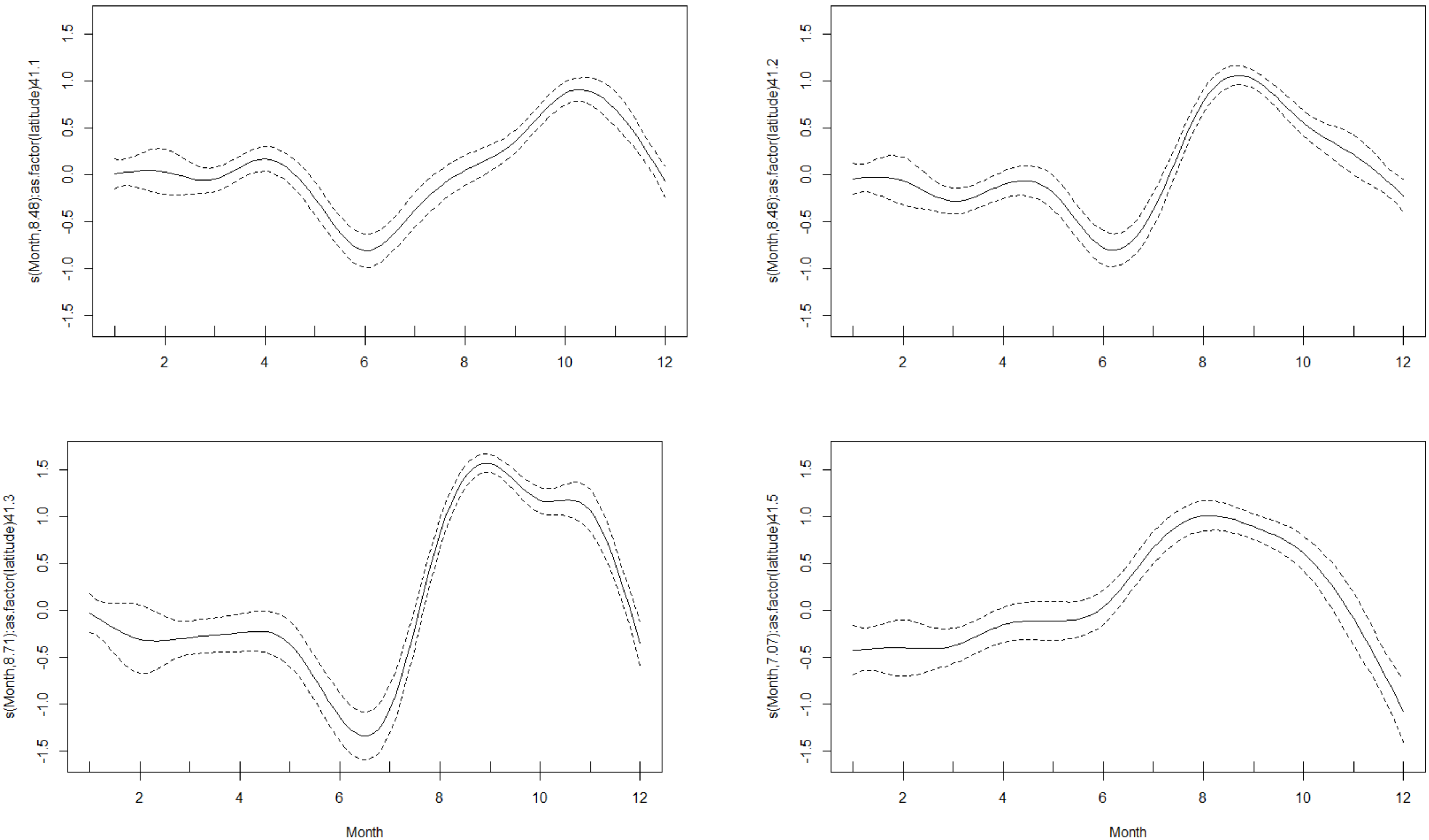
Model selection and validation

How do we assess fit?

- **Significance testing (not my favorite):**
  - Add or remove explanatory variables based on F- or likelihood ratio tests
    - Do not use default R output p-values!
- **Information theoretic approaches**
  - Measure predictive 'loss'
  - Akaike Information Criteria (AIC)
    - Need to be careful with edf – the debate rages on
- **Best approach (for now):**
  - Selection via AIC backed up with cross validation
- **Model validation**
  - Inspect residual and other diagnostic plots carefully

```
> AIC(m0,m1,m2,m3,m4,m5,m6)
          df       AIC
m0  1.000000 10983.282
m1 12.000000  9115.437
m2  2.000000 10649.363
m3  4.000000  9672.105
m4  9.864632  9126.083
m5  8.891544  9166.999
m6 42.645039  8538.063
```

Is all this wiggliness a good idea?

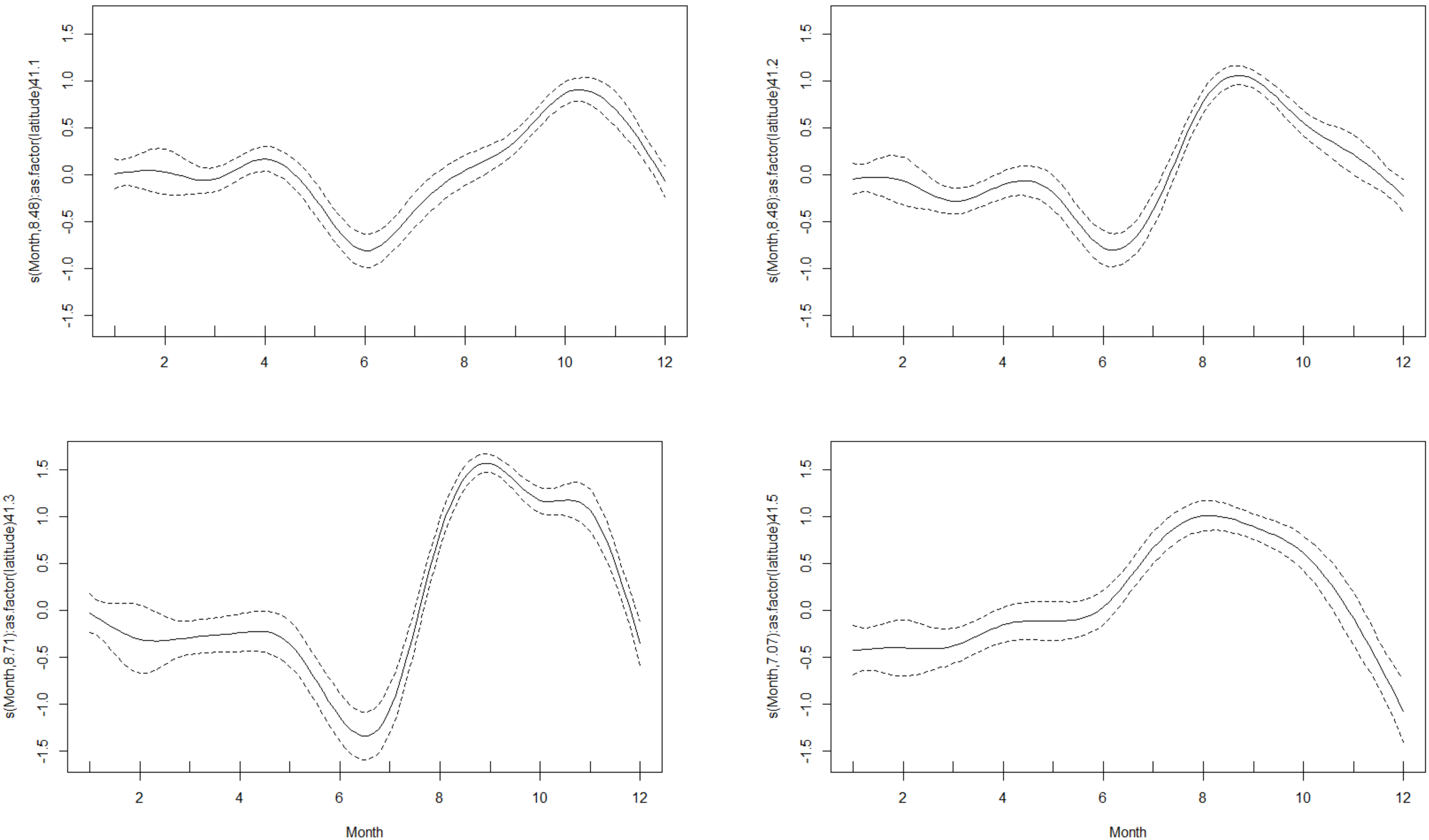


Words of caution

You may be tempted to use GAMs for everything (GAMania)

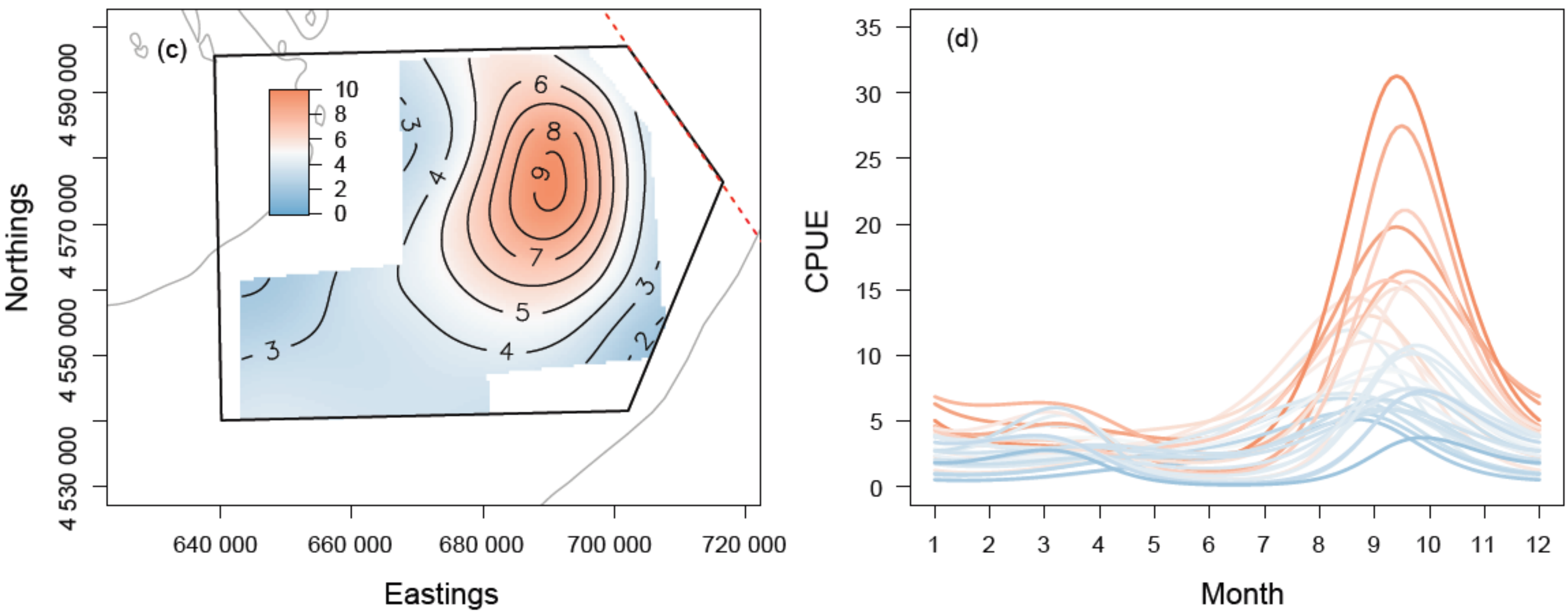
- Tendency to overfit data (i.e., be too wiggly), even when using penalized splines
  - Can limit predictive usefulness
- A lot of times, a well formulated polynomial can do almost as good of a job fitting to the data
  - Have more informative parameter estimates/greater predictive power
- **Importance of model validation**
  - Inspect residual and other diagnostic plots carefully
  - If we had time to do this, we would have discovered a lot of problems with our YT model.
- **Use your biological intuition!**

Coming back to our YT example



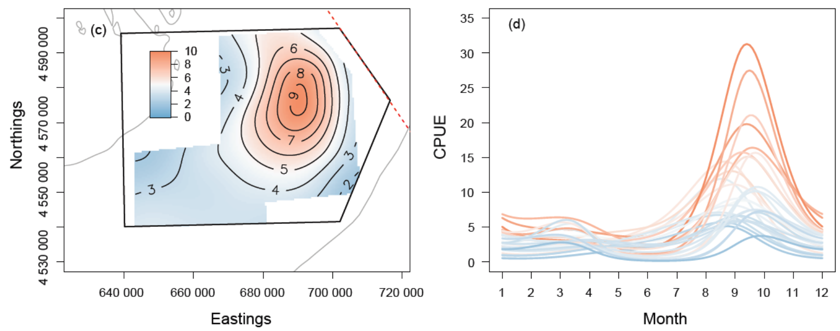
Extensions

Spatial models



Extensions

Spatiotemporal models – when response varies over space and time



Zero-inflated models – when response contains many zeros

- Probably appropriate for our YT example

Mixed effects models/hierarchical models – incredibly useful

- When observations are correlated
- Or when you are interested in a phenomena that is not directly observable, but can be inferred from your data

Useful References

Hastie, T.J. and R.J. Tibshirani. 1990. Generalized Additive Models.: Chapman and Hall. New York.

Venables, W. N. and C. M. Dichmont. 2004. GLMs, GAMs and GLMMs: an overview of theory for applications in fisheries research. Fisheries Research, 2004.

Wood, S.N., 2006. Generalized Additive Models: An Introduction with R. Chapman & Hall, London.

Zuur, A. et al. 2009. Mixed Effects Models and Extensions in Ecology with R. Springer-Verlag New York.

Next week

2/28: Matrix Algebra Review

3/01: Lab 7 (writing your own functions)

3/02: Principal Components Analysis