GAMs to Analyze Plankton Community Based on Environmental Variables

Curtis C. Bohlen, Casco Bay Estuary Partnership

7/21/2022

Contents

Introduction

This notebook provides analyses using GAMs, based on model selection informed by closer analysis of model collinearity and concurvity.

In this Notebook, I emphasize physical environmental variables. The base model includes the following Fixed Effects predictors:

- Temperature
- Salinity
- log(Turb)
- log(Chl)
- log1p(Fish)

Year is included in the model as a random effect largely to reduce unexplained variance in the model.

This means these models omit:

- Discharge (highly collinear with other predictors),
- Oxygen Saturation (Incomplete data and Highly collinear with Temperature),
- Season (Highly correlated with multiple predictors, especially Temperature)
- Station (Moderately correlated with Salinity and Temperature)
- Sample Event (inclusion as a random factor often led to over-specified models and it seldom proved important)

Load Libraries

```
library(tidyverse)
#> -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
#> v ggplot2 3.3.6     v purrr   0.3.4
#> v tibble  3.1.7     v dplyr   1.0.9
#> v tidyr   1.2.0     v stringr 1.4.0
#> v readr   2.1.2     v forcats 0.5.1
#> -- Conflicts ------------------------------------------ tidyverse_conflicts() --
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
library(readxl)
library(mgcv)      # for GAM models
#> Loading required package: nlme
#>
#> Attaching package: 'nlme'
#> The following object is masked from 'package:dplyr':
#>
#>     collapse
#> This is mgcv 1.8-40. For overview type 'help("mgcv-package")'.
library(emmeans)   # For extracting useful "marginal" model summaries
```

Set Graphics Theme

This sets `ggplot()`graphics for no background, no grid lines, etc. in a clean format suitable for (some) publications.

```
theme_set(theme_classic())
```

Input Data

Folder References

```
data_folder <- "Original_Data"
```

Load Data

```
filename.in <- "penob.station.data EA 3.12.20.xlsx"
file_path <- file.path(data_folder, filename.in)
station_data <- read_excel(file_path,
                           sheet="Final", col_types = c("skip", "date",
                                                        "numeric", "text", "numeric",
                                                        "text", "skip", "skip",
                                                        "skip",
                                                        rep("numeric", 10),
                                                        "text",
                                                        rep("numeric", 47),
                                                        "text",
```

```
                                                rep("numeric", 12))) %>%
  rename_with(~ gsub(" ", "_", .x)) %>%
  rename_with(~ gsub("\\.", "_", .x)) %>%
  rename_with(~ gsub("\\?", "", .x)) %>%
  rename_with(~ gsub("%", "pct", .x)) %>%
  rename_with(~ gsub("_Abundance", "", .x)) %>%
  filter(! is.na(date))
#> New names:
#> * `` -> `...61`
```

```
names(station_data)[10:12]
#> [1] "discharge_week_cftpersec" "discharg_day"
#> [3] "discharge_week_max"
names(station_data)[10:12] <- c('disch_wk', 'disch_day', 'disch_max')
```

Station names are arbitrary, and Erin previously expressed interest in renaming them from Stations 2, 4, 5 and 8 to Stations 1,2,3,and 4.

The `factor()` function by default sorts levels before assigning numeric codes, so a convenient way to replace the existing station codes with sequential numbers is to create a factor and extract the numeric indicator values with `as.numeric()`.

```
station_data <- station_data %>%
  mutate(station = factor(as.numeric(factor(station))))
head(station_data)
#> # A tibble: 6 x 76
#>   date                year month month_num season riv_km station station_num
#>   <dttm>              <dbl> <chr>     <dbl> <chr>   <dbl> <fct>         <dbl>
#> 1 2013-05-28 00:00:00  2013 May           5 Spring  22.6  1                 1
#> 2 2013-05-28 00:00:00  2013 May           5 Spring  13.9  2                 2
#> 3 2013-05-28 00:00:00  2013 May           5 Spring   8.12 3                 3
#> 4 2013-05-28 00:00:00  2013 May           5 Spring   2.78 4                 4
#> 5 2013-07-25 00:00:00  2013 July          7 Summer  22.6  1                 1
#> 6 2013-07-25 00:00:00  2013 July          7 Summer  13.9  2                 2
#> # ... with 68 more variables: depth <dbl>, disch_wk <dbl>, disch_day <dbl>,
#> #   disch_max <dbl>, tide_height <dbl>, Full_Moon <dbl>, Abs_Moon <dbl>,
#> #   Spring_or_Neap <chr>, ave_temp_c <dbl>, ave_sal_psu <dbl>,
#> #   ave_turb_ntu <dbl>, ave_do_mgperl <dbl>, ave_DO_Saturation <dbl>,
#> #   ave_chl_microgperl <dbl>, sur_temp <dbl>, sur_sal <dbl>, sur_turb <dbl>,
#> #   sur_do <dbl>, sur_chl <dbl>, bot_temp <dbl>, bot_sal <dbl>, bot_turb <dbl>,
#> #   bot_do <dbl>, bot_chl <dbl>, max_temp <dbl>, max_sal <dbl>, ...
```

Subsetting to Desired Data Columns

I base selection of predictor variables here on the ones used in the manuscript.

```
base_data <- station_data %>%
  rename(Date = date,
         Station = station,
         Year = year) %>%
  select(-c(month, month_num)) %>%
  mutate(Month = factor(as.numeric(format(Date, format = '%m')),
```

```
                                            levels = 1:12,
                                            labels = month.abb),
         DOY = as.numeric(format(Date,format = '%j')),
         season = factor(season, levels = c('Spring', 'Summer', 'Fall')),
         is_sp_up = season == 'Spring' & Station == 1,
         Yearf = factor(Year)) %>%
  rename(Season = season,
         Density = combined_density,
         Temp = ave_temp_c,
         Sal = ave_sal_psu,
         Turb = sur_turb,
         AvgTurb = ave_turb_ntu,
         DOsat = ave_DO_Saturation,
         Chl = ave_chl_microgperl,
         Fish = `___61`,
         RH = Herring
         ) %>%
  select(Date, Station, Year, Yearf, Month, Season, is_sp_up, DOY, riv_km,
         disch_wk, disch_day, disch_max,
         Temp, Sal, Turb, AvgTurb, DOsat, Chl,
         Fish, RH,
         Density, H, SEI,
         Acartia, Balanus, Eurytemora, Polychaete, Pseudocal, Temora) %>%
  arrange(Date, Station)
head(base_data)
#> # A tibble: 6 x 29
#>   Date                Station  Year Yearf Month Season is_sp_up   DOY riv_km
#>   <dttm>              <fct>   <dbl> <fct> <fct> <fct>  <lgl>    <dbl>  <dbl>
#> 1 2013-05-28 00:00:00 1        2013 2013  May   Spring TRUE       148  22.6
#> 2 2013-05-28 00:00:00 2        2013 2013  May   Spring FALSE      148  13.9
#> 3 2013-05-28 00:00:00 3        2013 2013  May   Spring FALSE      148   8.12
#> 4 2013-05-28 00:00:00 4        2013 2013  May   Spring FALSE      148   2.78
#> 5 2013-07-25 00:00:00 1        2013 2013  Jul   Summer FALSE      206  22.6
#> 6 2013-07-25 00:00:00 2        2013 2013  Jul   Summer FALSE      206  13.9
#> # ... with 20 more variables: disch_wk <dbl>, disch_day <dbl>, disch_max <dbl>,
#> #   Temp <dbl>, Sal <dbl>, Turb <dbl>, AvgTurb <dbl>, DOsat <dbl>, Chl <dbl>,
#> #   Fish <dbl>, RH <dbl>, Density <dbl>, H <dbl>, SEI <dbl>, Acartia <dbl>,
#> #   Balanus <dbl>, Eurytemora <dbl>, Polychaete <dbl>, Pseudocal <dbl>,
#> #   Temora <dbl>
```

```
rm(station_data)
```

Complete Cases

This drops only two samples, one for missing Zooplankton data, one for missing fish data. We need this reduced data set to run The `step()` function. It makes little sense to try stepwise model selection if each time you add or remove a variable, the sample you are studying changes. Since fish is never an important predictor, we will want need to refit models after stepwise elimination to use the most complete possible data set.

```
complete_data <- base_data %>%
  select(Season, Station, Yearf,
```

```
         is_sp_up, Temp, Sal, Turb, Chl, Fish, RH,
         Density, H,
         Acartia, Balanus, Eurytemora, Polychaete, Pseudocal, Temora) %>%
  filter(complete.cases(.))
```

Reduced Data

The low salinity spring samples are doing something rather different, and they complicate model fitting. Models are far better behaved if we exclude a few extreme samples. These are low salinity low zooplankton samples. We have two complementary ways to specify which samples to omit, without just omitting "outliers". The first is to restrict modeling to "marine" samples over a certain salinity, and the other is to omit spring upstream samples, which include most of the problematic samples.

```
drop_low <- complete_data %>%
  filter(Sal > 10)    # Pulls three samples, including one fall upstream sample
                      # a fourth low salinity sample lacks zooplankton data
#drop_sp_up <- complete_data %>%
#  filter(! is_sp_up)  # drops four samples
```

Models of Fish Abundance

Model 1

```
fish_gam <- gam(log1p(Fish) ~
                  Station +
                  s(Temp, bs="ts", k = 5) +
                  s(Sal, bs="ts", k = 5) +
                  s(log(Turb), bs="ts", k = 5) +
                  s(log(Chl), bs="ts", k = 5) +
                  s(log1p(Density), bs="ts", k = 5) +
                  s(Yearf, bs = 're'),
                data = complete_data, family = 'gaussian')
summary(fish_gam)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ Station + s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   4.8898     0.4735  10.328 1.37e-13 ***
#> Station2     -0.7040     0.6662  -1.057    0.296
#> Station3     -0.9525     0.6493  -1.467    0.149
#> Station4     -0.7792     0.7098  -1.098    0.278
#> ---
```

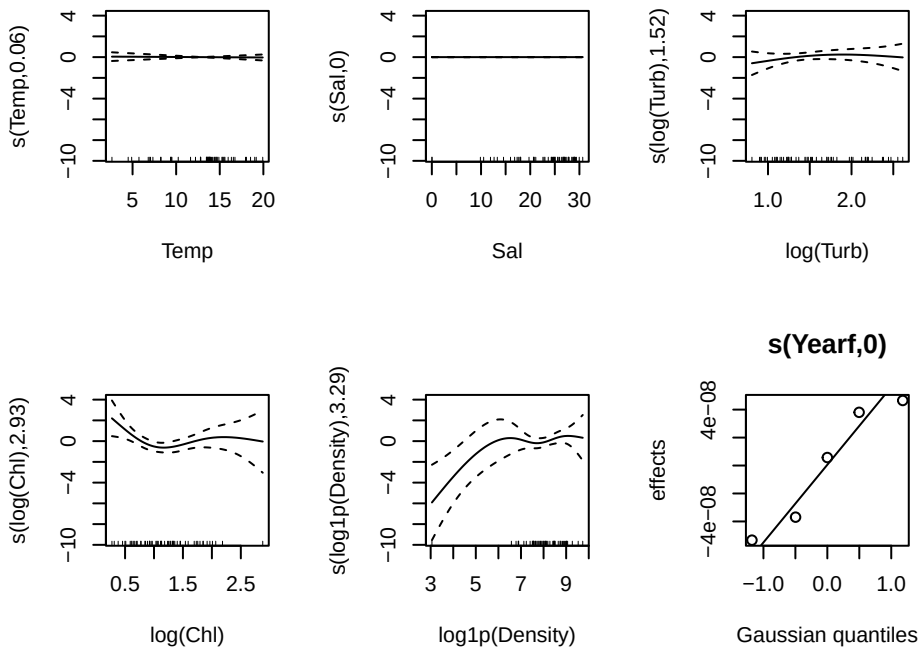
```
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                          edf Ref.df     F p-value
#> s(Temp)           6.095e-02      4 0.013  0.3440
#> s(Sal)            1.246e-07      4 0.000  0.9534
#> s(log(Turb))      1.521e+00      4 0.347  0.3922
#> s(log(Chl))       2.929e+00      4 2.151  0.0329 *
#> s(log1p(Density)) 3.289e+00      4 2.991  0.0112 *
#> s(Yearf)          3.793e-07      4 0.000  0.5569
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.183   Deviance explained = 33.8%
#> GCV = 3.3124  Scale est. = 2.6385    n = 58
```

Note that the model only explains on the order of 33% of the variance.

```
anova(fish_gam)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ Station + s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric Terms:
#>         df    F p-value
#> Station  3 0.78   0.511
#>
#> Approximate significance of smooth terms:
#>                          edf    Ref.df     F p-value
#> s(Temp)           6.095e-02 4.000e+00 0.013  0.3440
#> s(Sal)            1.246e-07 4.000e+00 0.000  0.9534
#> s(log(Turb))      1.521e+00 4.000e+00 0.347  0.3922
#> s(log(Chl))       2.929e+00 4.000e+00 2.151  0.0329
#> s(log1p(Density)) 3.289e+00 4.000e+00 2.991  0.0112
#> s(Yearf)          3.793e-07 4.000e+00 0.000  0.5569
```

Plot GAM Results

```
oldpar <- par(mfrow = c(2,3))
plot(fish_gam)
```




```
par(oldpar)
```

The GAM fit is highly influenced by a low Density sample with very few fish.

Concurvidity Analysis

A common recommendation is that values over 0.8 are problematic. We have a few values close to that cutoff. This analysis, however, is sometimes misleading with shrinkage estimators, as some terms are nearly removed from the model. Nevertheless, this model is troubling.

We could get away with including Station in a linear model, but with the GAMs, it appears the model including Station has too many problems with concurvity.

```
concurvity(fish_gam)
#>          para   s(Temp)    s(Sal) s(log(Turb)) s(log(Chl)) s(log1p(Density))
#> worst       1 0.8328234 0.9111731    0.7079354   0.7639046         0.8231068
#> observed    1 0.7734183 0.8661907    0.4656816   0.6025788         0.7964007
#> estimate    1 0.7232478 0.8391191    0.5833858   0.6973428         0.7627393
#>           s(Yearf)
#> worst    1.0000000
#> observed 0.3955044
#> estimate 0.6366590
```

Simplified Model

```
fish_gam_2 <- update( fish_gam, .~.-Station)
concurvity(fish_gam_2)
#>          para   s(Temp)    s(Sal) s(log(Turb)) s(log(Chl)) s(log1p(Density))
#> worst       1 0.7616888 0.7818932    0.5653240   0.6808175         0.7995049
#> observed    1 0.6213389 0.6934565    0.4304950   0.5831642         0.7942603
#> estimate    1 0.5858850 0.6778488    0.4340869   0.6215817         0.7262430
#>           s(Yearf)
#> worst    1.0000000
#> observed 0.3598406
#> estimate 0.6112068
```

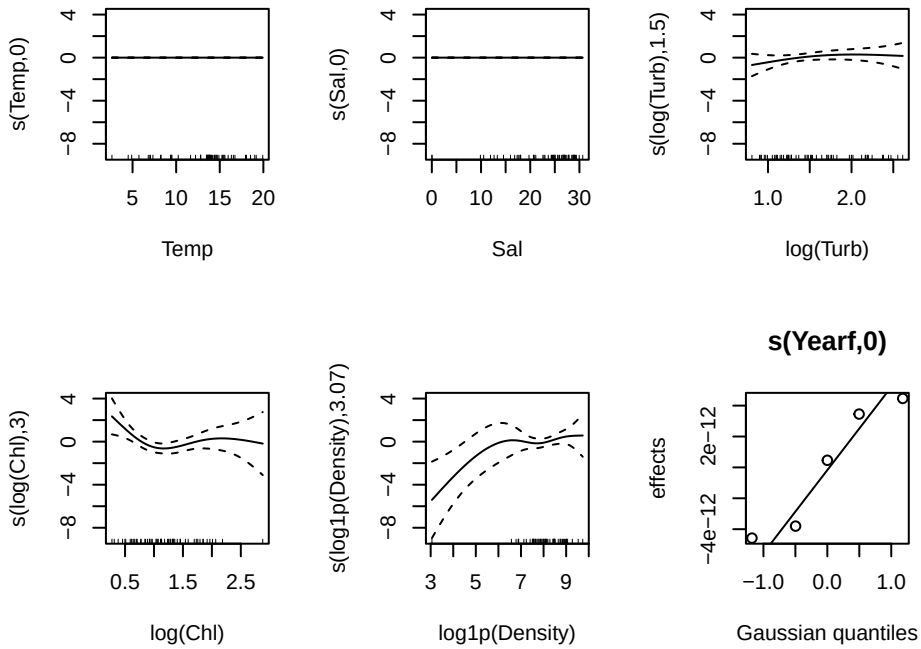
```
summary(fish_gam_2)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) +
#>     s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) +
#>     s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   4.2733     0.2125   20.11   <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(Temp)            1.194e-08      4 0.000  0.8663
#> s(Sal)             6.304e-09      4 0.000  0.4111
#> s(log(Turb))       1.498e+00      4 0.465  0.2795
#> s(log(Chl))        3.004e+00      4 2.423  0.0220 *
#> s(log1p(Density)) 3.068e+00      4 2.627  0.0157 *
#> s(Yearf)           3.816e-11      5 0.000  0.6200
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.189   Deviance explained = 29.7%
#> GCV = 3.0733  Scale est. = 2.6192    n = 58
```

```
anova(fish_gam_2)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) +
#>     s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) +
#>     s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                          edf    Ref.df     F p-value
```

```
#> s(Temp)           1.194e-08 4.000e+00 0.000  0.8663
#> s(Sal)            6.304e-09 4.000e+00 0.000  0.4111
#> s(log(Turb))      1.498e+00 4.000e+00 0.465  0.2795
#> s(log(Chl))       3.004e+00 4.000e+00 2.423  0.0220
#> s(log1p(Density)) 3.068e+00 4.000e+00 2.627  0.0157
#> s(Yearf)          3.816e-11 5.000e+00 0.000  0.6200
```

Plot GAM Results

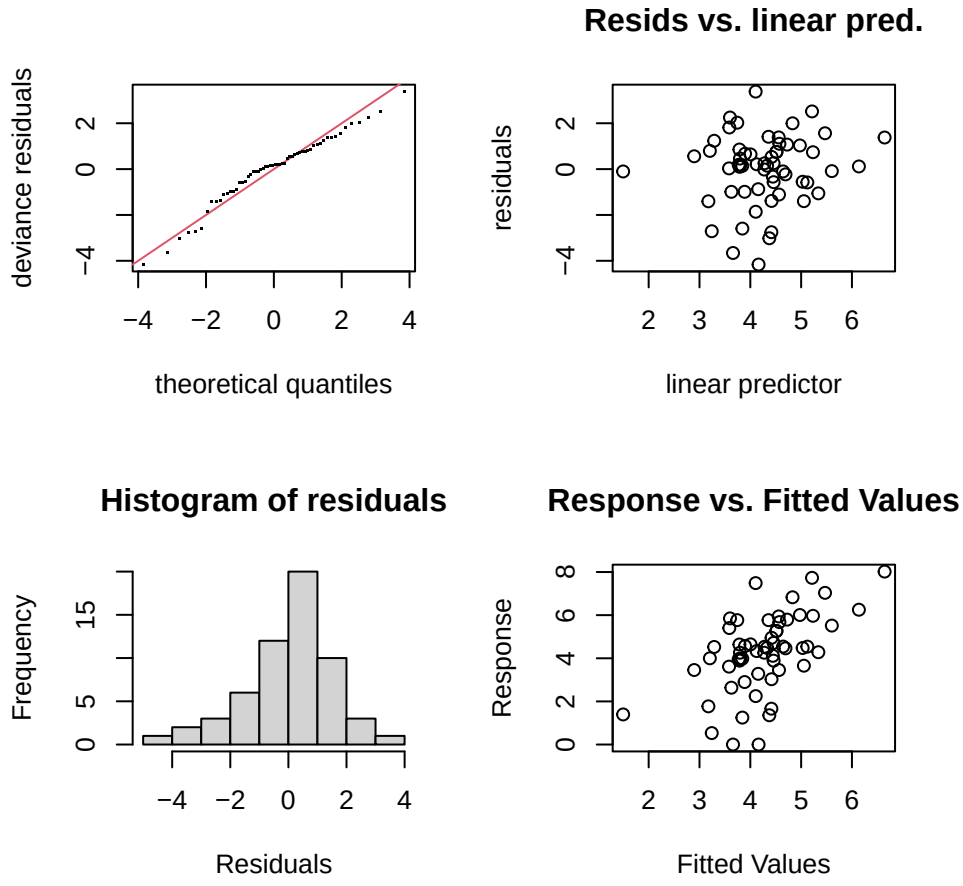
```
oldpar <- par(mfrow = c(2,3))
plot(fish_gam_2)
```




```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(fish_gam_2)
```




```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 20 iterations.
#> The RMS GCV score gradient at convergence was 1.16717e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                        k'      edf k-index p-value
#> s(Temp)           4.00e+00 1.19e-08    0.87   0.150
#> s(Sal)            4.00e+00 6.30e-09    0.81   0.045 *
#> s(log(Turb))      4.00e+00 1.50e+00    0.96   0.325
#> s(log(Chl))       4.00e+00 3.00e+00    1.00   0.385
#> s(log1p(Density)) 4.00e+00 3.07e+00    1.05   0.580
#> s(Yearf)          5.00e+00 3.82e-11      NA      NA
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
par(oldpar)
```

The model is pretty good, with slightly skewed and slightly heavy tails to the residuals and one moderate outlier.

Model on Reduced Data

We refit to data that omits samples where salinity was below 10 PSU. Based on our prior analysis, we omit Station as a predictor again, to avoid concurvity issues.

This reduced data set drops he lowest fish abundance sample in the data set, because it happened to coincide with a low salinity (and low plankton density sample.

```
fish_gam_no_low <- gam(log1p(Fish) ~
                       s(Temp, bs="ts", k = 5) +
                       s(Sal, bs="ts", k = 5) +
                       s(log(Turb), bs="ts", k = 5) +
                       s(log(Chl), bs="ts", k = 5) +
                       s(log1p(Density), bs="ts", k = 5) +
                       s(Yearf, bs = 're'),
                       data = drop_low, family = 'gaussian')
summary(fish_gam_no_low)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) +
#>     s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) +
#>     s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   4.3263     0.2214   19.54   <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(Temp)            2.679e-10      4 0.00  0.7365
#> s(Sal)             3.205e-10      4 0.00  0.4881
#> s(log(Turb))       1.700e+00      4 1.01  0.0923 .
#> s(log(Chl))        2.913e+00      4 2.10  0.0370 *
#> s(log1p(Density)) 9.469e-11      4 0.00  0.5653
#> s(Yearf)           3.162e-11      5 0.00  0.6826
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =   0.17   Deviance explained = 24.1%
#> GCV = 3.0036  Scale est. = 2.697     n = 55
```

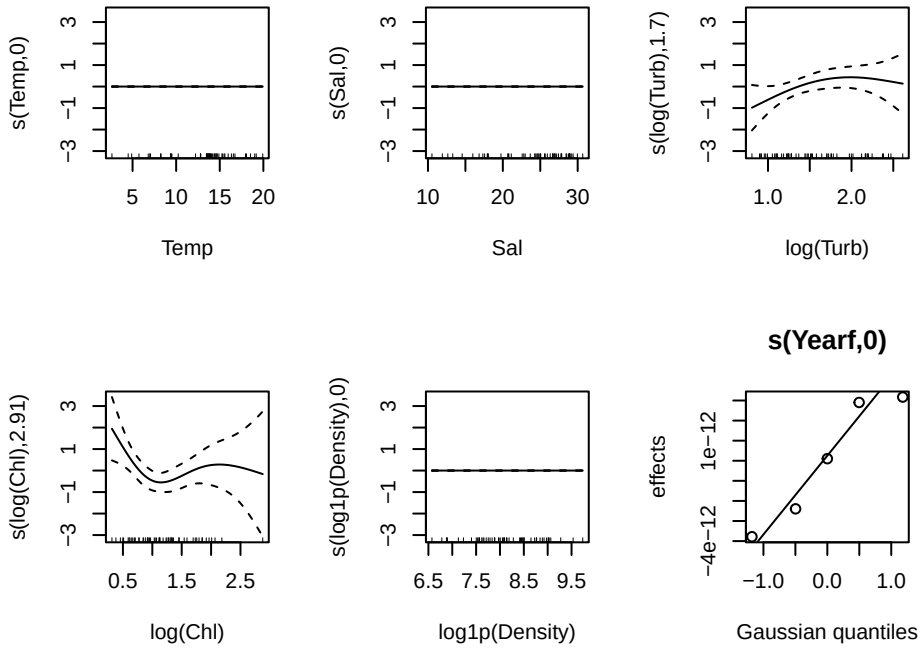
This model explains under 25% of the variance. The equivalent linear model failed to identify any statistically significant predictors, which highlights the importance of not quite linear relationships.

```
anova(fish_gam_no_low)
#>
#> Family: gaussian
#> Link function: identity
#>
```

```
#> Formula:
#> log1p(Fish) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) +
#>     s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) +
#>     s(log1p(Density), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                          edf    Ref.df    F p-value
#> s(Temp)           2.679e-10 4.000e+00 0.00  0.7365
#> s(Sal)            3.205e-10 4.000e+00 0.00  0.4881
#> s(log(Turb))      1.700e+00 4.000e+00 1.01  0.0923
#> s(log(Chl))       2.913e+00 4.000e+00 2.10  0.0370
#> s(log1p(Density)) 9.469e-11 4.000e+00 0.00  0.5653
#> s(Yearf)          3.162e-11 5.000e+00 0.00  0.6826
```

Plot GAM Results

```
oldpar <- par(mfrow = c(2,3))
plot(fish_gam_no_low)
```

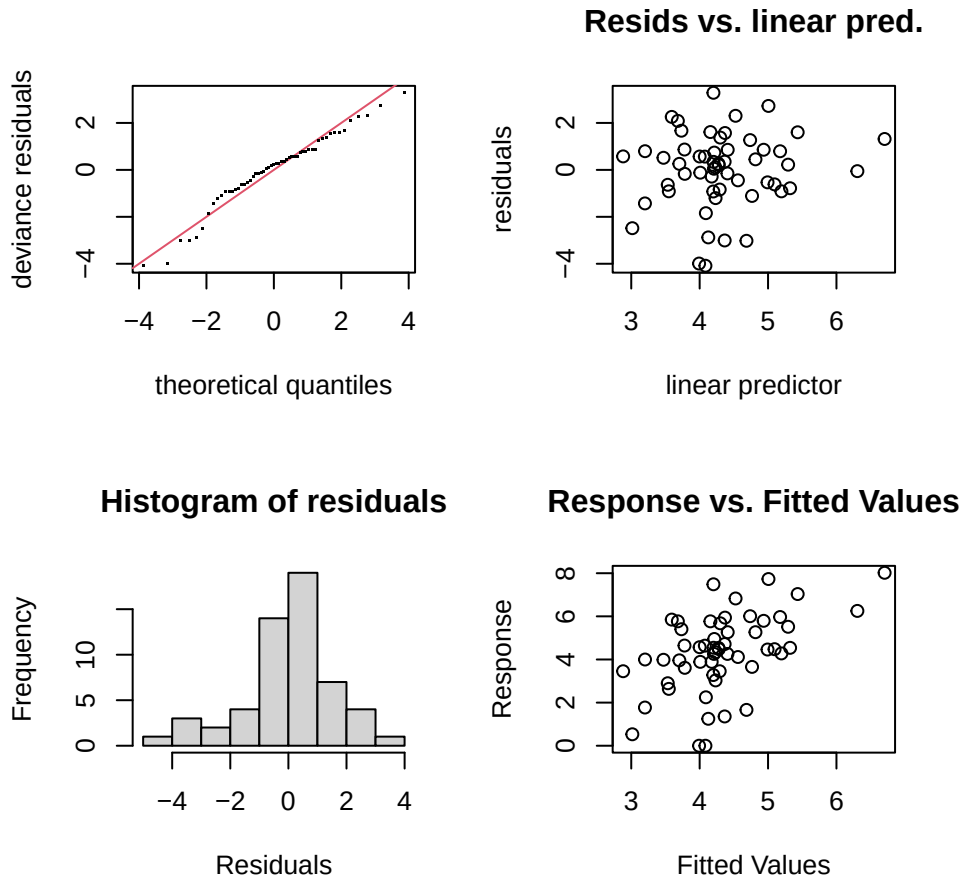



```
par(oldpar)
```

Overall, fish abundance is high at very low chlorophyll and drops off at log(chlorophyll) ~= 1, or Chlorophyll ~ 2.7. Fish also tends to be less abundant under low turbidity conditions, but the relationship has high uncertainty.

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(fish_gam_no_low)
```




```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 26 iterations.
#> The RMS GCV score gradient at convergence was 1.325503e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                         k'      edf k-index p-value
#> s(Temp)           4.00e+00 2.68e-10    0.91   0.225
#> s(Sal)            4.00e+00 3.20e-10    0.83   0.085 .
#> s(log(Turb))      4.00e+00 1.70e+00    0.97   0.345
#> s(log(Chl))       4.00e+00 2.91e+00    0.95   0.335
#> s(log1p(Density)) 4.00e+00 9.47e-11    0.98   0.395
```

```
#> s(Yearf)          5.00e+00 3.16e-11      NA      NA
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
par(oldpar)
```

That model is fairly robust, although it shows some signs of kurtosis.

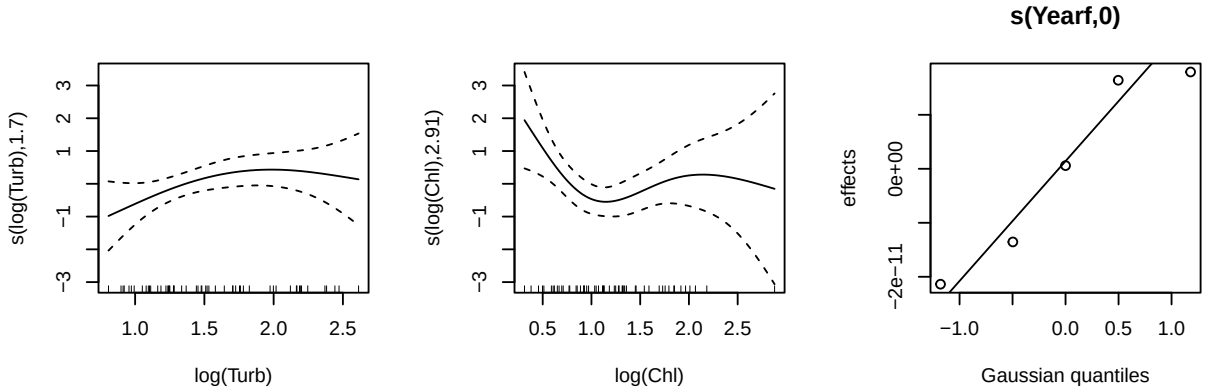
Reduced Complexity Model

I drop Temperature, Salinity and Zooplankton density, none of which are significant individually or collectively in various intermediate models not shown)

What remains significant is chlorophyll.

```
fish_gam_reduced <- gam(log1p(Fish) ~
                      #s(Temp, bs="ts", k = 5) +
                      #s(Sal, bs="ts", k = 5) +
                      s(log(Turb), bs="ts", k = 5) +
                      s(log(Chl), bs="ts", k = 5)  +
                      #s(log1p(Density), bs="ts", k = 5)  +
                      s(Yearf, bs = 're'),
                      data = drop_low, family = 'gaussian')
summary(fish_gam_reduced)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Fish) ~ s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   4.3263     0.2215   19.54   <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(log(Turb)) 1.70e+00      4 1.010  0.0924 .
#> s(log(Chl))  2.91e+00      4 2.089  0.0374 *
#> s(Yearf)     1.79e-10      5 0.000  0.6835
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =   0.17   Deviance explained = 24.1%
#> GCV = 3.0039  Scale est. = 2.6975    n = 55
```

```
oldpar <- par(mfrow = c(1,3))
plot(fish_gam_reduced)
```




```
par(oldpar)
```

The reduced model has no impact on the basic conclusions of the model.

Total Zooplankton Density

I fit the simplified model without Station. The full model has the same concurvity problems as before, and here the model fails to converge. While I could alter the convergence criteria to search for a solution, we know the model that includes Station will have concurvity problems, so there is little point.

Model 1

```
density_gam<- gam(log(Density) ~
                          s(Temp, bs="ts", k = 5) +
                          s(Sal, bs="ts", k = 5) +
                          s(log(Turb), bs="ts", k = 5) +
                          s(log(Chl), bs="ts", k = 5) +
                          s(log1p(Fish),bs="ts", k = 5) +
                          s(Yearf, bs = 're'),
                        data = complete_data, family = 'gaussian')
concurvity(density_gam)
#>          para   s(Temp)    s(Sal) s(log(Turb)) s(log(Chl)) s(log1p(Fish))
#> worst       1 0.6687582 0.7446731    0.4388267   0.6815834      0.4367056
#> observed    1 0.5202641 0.4756895    0.3992914   0.6303849      0.3264736
#> estimate    1 0.5372910 0.5754949    0.3616421   0.6194483      0.3402522
#>           s(Yearf)
#> worst    1.0000000
#> observed 0.4408282
#> estimate 0.5359525
```

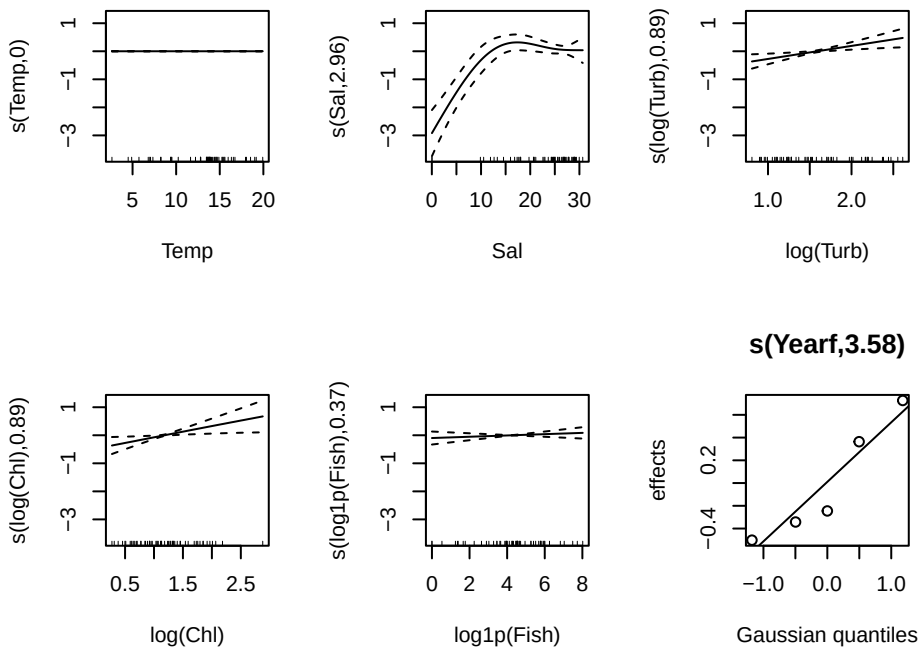
```
summary(density_gam)
#>
#> Family: gaussian
#> Link function: identity
```

```
#>
#> Formula:
#> log(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   8.0049     0.2464   32.49   <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(Temp)        2.466e-10      4  0.000  0.48812
#> s(Sal)         2.965e+00      4 18.779 1.28e-06 ***
#> s(log(Turb))   8.949e-01      4  2.730  0.00246 **
#> s(log(Chl))    8.941e-01      4  3.642  0.01008 *
#> s(log1p(Fish)) 3.706e-01      4  0.192  0.16214
#> s(Yearf)       3.584e+00      4  9.786 2.27e-06 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.656   Deviance explained = 70.9%
#> GCV = 0.3922  Scale est. = 0.32655   n = 58
```

```
anova(density_gam)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                      edf    Ref.df      F  p-value
#> s(Temp)        2.466e-10 4.000e+00  0.000  0.48812
#> s(Sal)         2.965e+00 4.000e+00 18.779 1.28e-06
#> s(log(Turb))   8.949e-01 4.000e+00  2.730  0.00246
#> s(log(Chl))    8.941e-01 4.000e+00  3.642  0.01008
#> s(log1p(Fish)) 3.706e-01 4.000e+00  0.192  0.16214
#> s(Yearf)       3.584e+00 4.000e+00  9.786 2.27e-06
```

Plot the GAM

```
oldpar <- par(mfrow = c(2,3))
plot(density_gam)
```




```
par(oldpar)
```

The only significantly non-linear response is the response for salinity, which appears to be driven by a few very low salinity samples.

Model on Reduced Data

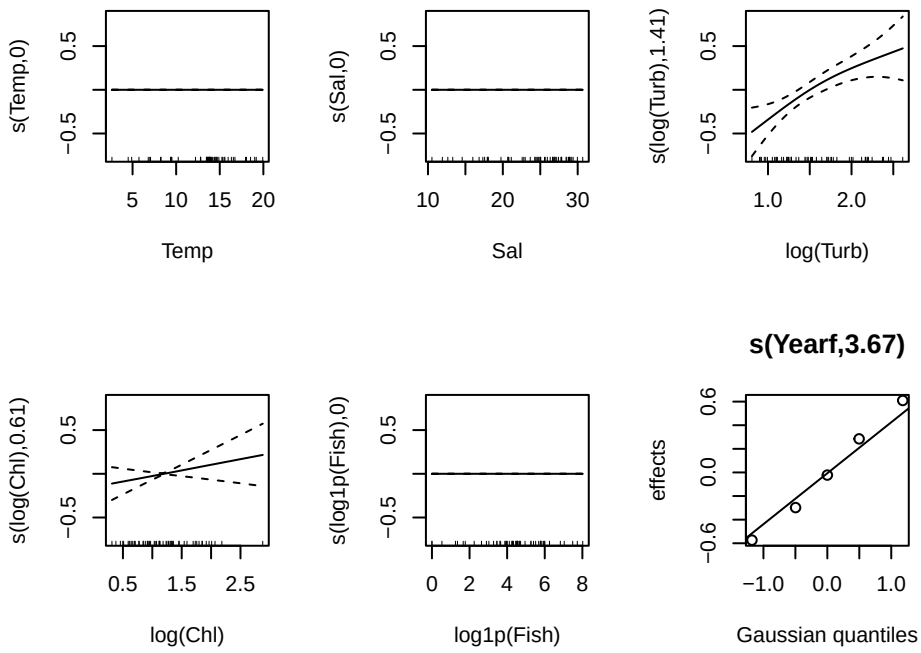
```
density_gam_no_low<- gam(log(Density) ~
                           s(Temp, bs="ts", k = 5) +
                           s(Sal, bs="ts", k = 5) +
                           s(log(Turb), bs="ts", k = 5) +
                           s(log(Chl), bs="ts", k = 5) +
                           s(log1p(Fish),bs="ts", k = 5) +
                           s(Yearf, bs = 're'),
                         data = drop_low, family = 'gaussian')
summary(density_gam_no_low)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
```

```
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   8.1283     0.2307   35.23   <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(Temp)        4.718e-10      4  0.00 0.299862
#> s(Sal)         1.471e-10      4  0.00 0.613212
#> s(log(Turb))   1.412e+00      4  6.18 0.000253 ***
#> s(log(Chl))    6.072e-01      4  0.83 0.122462
#> s(log1p(Fish)) 1.505e-10      4  0.00 0.645778
#> s(Yearf)       3.672e+00      4 10.52 1.63e-06 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.561   Deviance explained = 60.8%
#> GCV = 0.26018  Scale est. = 0.22853   n = 55
```

```
anova(density_gam_no_low)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df     F  p-value
#> s(Temp)        4.718e-10 4.000e+00  0.00 0.299862
#> s(Sal)         1.471e-10 4.000e+00  0.00 0.613212
#> s(log(Turb))   1.412e+00 4.000e+00  6.18 0.000253
#> s(log(Chl))    6.072e-01 4.000e+00  0.83 0.122462
#> s(log1p(Fish)) 1.505e-10 4.000e+00  0.00 0.645778
#> s(Yearf)       3.672e+00 4.000e+00 10.52 1.63e-06
```

Plot the GAM

```
oldpar <- par(mfrow = c(2,3))
plot(density_gam_no_low)
```



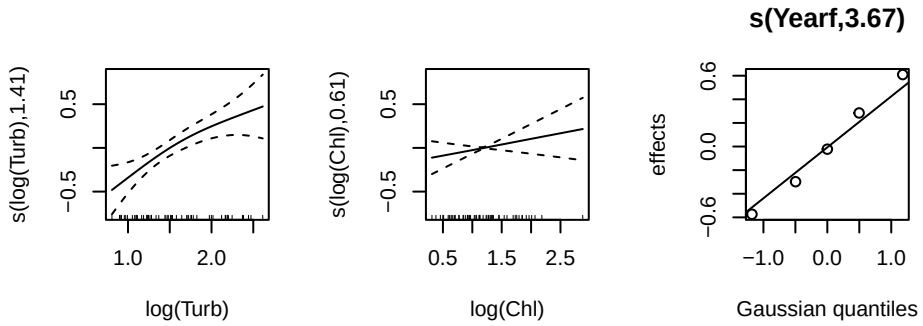

```
par(oldpar)
```

Reduced Complexity Model

```
density_gam_reduced<- gam(log(Density) ~
                            #s(Temp, bs="ts", k = 5) +
                            #s(Sal, bs="ts", k = 5) +
                            s(log(Turb), bs="ts", k = 5) +
                            s(log(Chl), bs="ts", k = 5) +
                            #s(log1p(Fish),bs="ts", k = 5) +
                            s(Yearf, bs = 're'),
                          data = drop_low, family = 'gaussian')
summary(density_gam_reduced)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log(Density) ~ s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   8.1283     0.2307   35.23   <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(log(Turb)) 1.4120      4  6.18 0.000253 ***
#> s(log(Chl))  0.6072      4  0.83 0.122462
#> s(Yearf)     3.6720      4 10.52 1.63e-06 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.561   Deviance explained = 60.8%
#> GCV = 0.26018  Scale est. = 0.22853   n = 55
```

```
oldpar <- par(mfrow = c(2,3))
plot(density_gam_reduced)
par(oldpar)
```



Again, results are little affected by removing non-significant terms. Even this simple model predicts on the order of 60% of the variance.

Shannon Diversity

Model 1

```
shannon_gam <- gam(H ~
                     s(Temp, bs="ts", k = 5) +
```

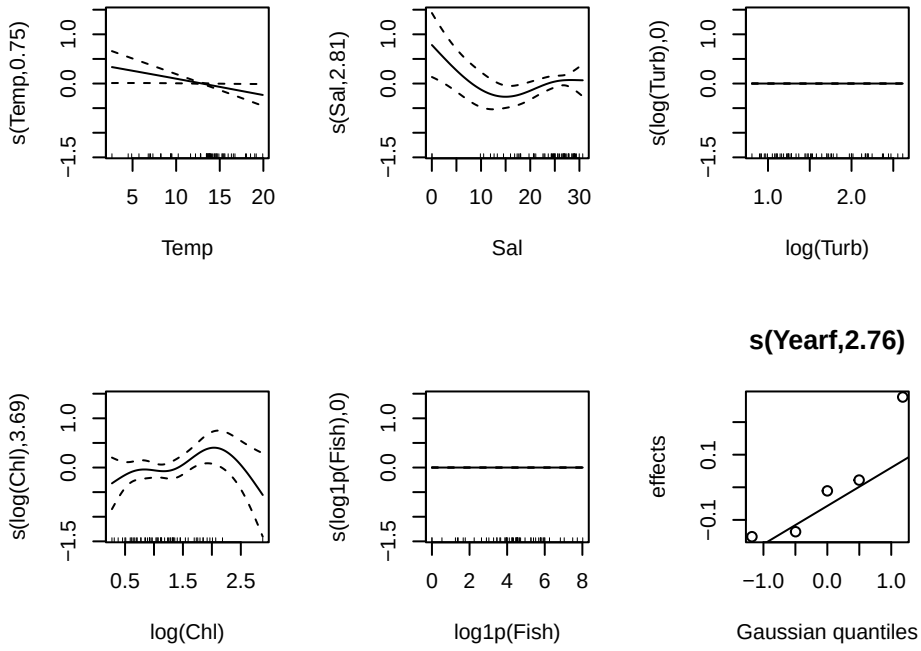
```
                    s(Sal, bs="ts", k = 5) +
                    s(log(Turb), bs="ts", k = 5) +
                    s(log(Chl), bs="ts", k = 5) +
                    s(log1p(Fish),bs="ts", k = 5) +
                    s(Yearf, bs = 're'),
                  data = complete_data, family = 'gaussian')
summary(shannon_gam)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> H ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) + s(log(Turb),
#>     bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) + s(log1p(Fish),
#>     bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   1.3561     0.1101   12.31 2.59e-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(Temp)        7.459e-01      4 1.604  0.0156 *
#> s(Sal)         2.813e+00      4 2.944  0.0244 *
#> s(log(Turb))   1.005e-08      4 0.000  0.9363
#> s(log(Chl))    3.693e+00      4 3.305  0.0639 .
#> s(log1p(Fish)) 1.101e-09      4 0.000  0.5674
#> s(Yearf)       2.762e+00      4 2.268  0.0162 *
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.387   Deviance explained = 49.5%
#> GCV = 0.2272  Scale est. = 0.18405   n = 58
```

```
anova(shannon_gam)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> H ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) + s(log(Turb),
#>     bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) + s(log1p(Fish),
#>     bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                     edf    Ref.df     F p-value
#> s(Temp)        7.459e-01 4.000e+00 1.604  0.0156
#> s(Sal)         2.813e+00 4.000e+00 2.944  0.0244
#> s(log(Turb))   1.005e-08 4.000e+00 0.000  0.9363
#> s(log(Chl))    3.693e+00 4.000e+00 3.305  0.0639
#> s(log1p(Fish)) 1.101e-09 4.000e+00 0.000  0.5674
```

```
#> s(Yearf)       2.762e+00 4.000e+00 2.268  0.0162
```

Plot the GAM

```
oldpar <- par(mfrow = c(2,3))
plot(shannon_gam)
```




```
par(oldpar)
```

Again, the relationship with salinity appear principally driven by a couple of very low salinity outliers.

Model on Reduced Data

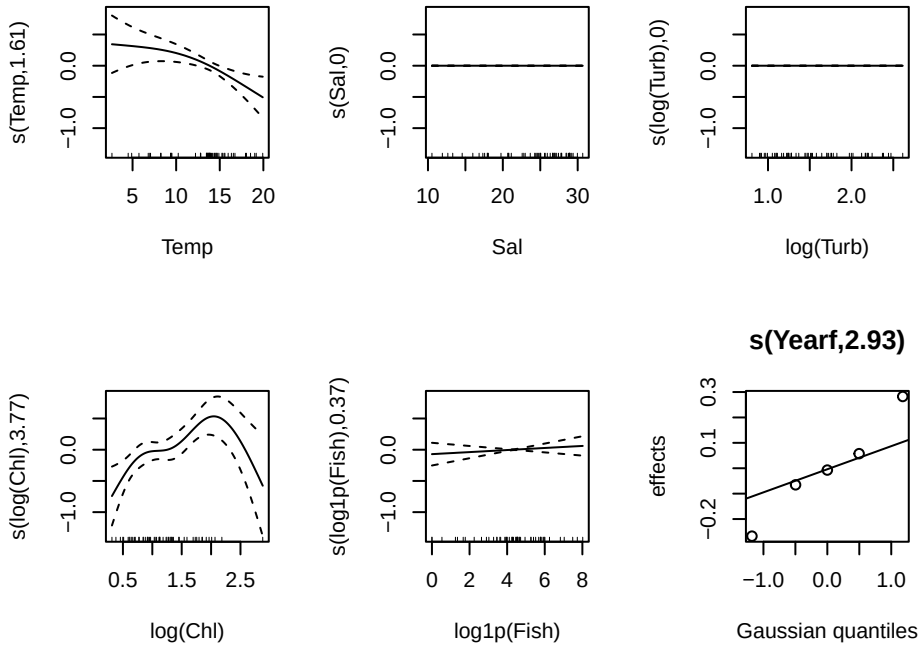
```
shannon_gam_no_low <- gam(H ~
                      s(Temp, bs="ts", k = 5) +
                      s(Sal, bs="ts", k = 5) +
                      s(log(Turb), bs="ts", k = 5) +
                      s(log(Chl), bs="ts", k = 5) +
                      s(log1p(Fish),bs="ts", k = 5) +
                      s(Yearf, bs = 're'),
                    data = drop_low, family = 'gaussian')
summary(shannon_gam_no_low)
#>
```

```
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> H ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) + s(log(Turb),
#>     bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) + s(log1p(Fish),
#>     bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   1.3310     0.1142   11.66  3.1e-15 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                       edf Ref.df      F  p-value
#> s(Temp)         1.615e+00      4  4.222 0.002901 **
#> s(Sal)          2.259e-08      4  0.000 0.257386
#> s(log(Turb))    1.369e-08      4  0.000 0.608480
#> s(log(Chl))     3.767e+00      4 11.002 0.000252 ***
#> s(log1p(Fish)) 3.675e-01      4  0.167 0.197576
#> s(Yearf)        2.929e+00      4  2.802 0.008131 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.417   Deviance explained = 51.1%
#> GCV =    0.2  Scale est. = 0.1648    n = 55
```

```
anova(shannon_gam_no_low)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> H ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts", k = 5) + s(log(Turb),
#>     bs = "ts", k = 5) + s(log(Chl), bs = "ts", k = 5) + s(log1p(Fish),
#>     bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df      F  p-value
#> s(Temp)         1.615e+00 4.000e+00  4.222 0.002901
#> s(Sal)          2.259e-08 4.000e+00  0.000 0.257386
#> s(log(Turb))    1.369e-08 4.000e+00  0.000 0.608480
#> s(log(Chl))     3.767e+00 4.000e+00 11.002 0.000252
#> s(log1p(Fish)) 3.675e-01 4.000e+00  0.167 0.197576
#> s(Yearf)        2.929e+00 4.000e+00  2.802 0.008131
```

Plot the GAM

```
oldpar <- par(mfrow = c(2,3))
plot(shannon_gam_no_low)
```

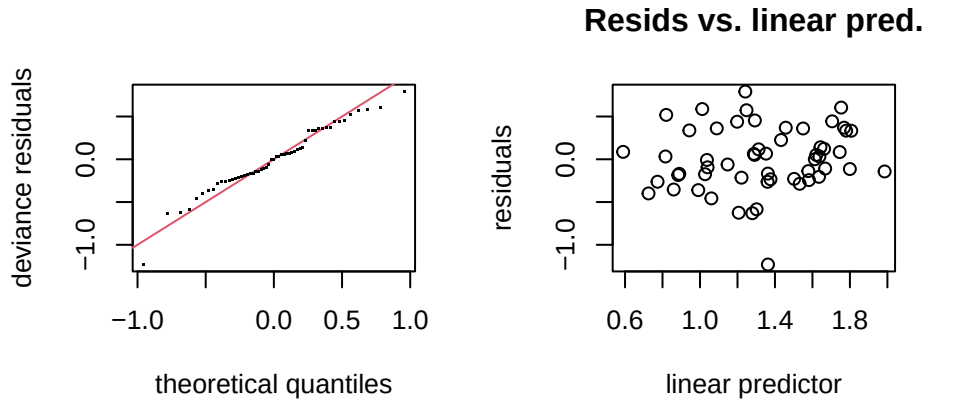


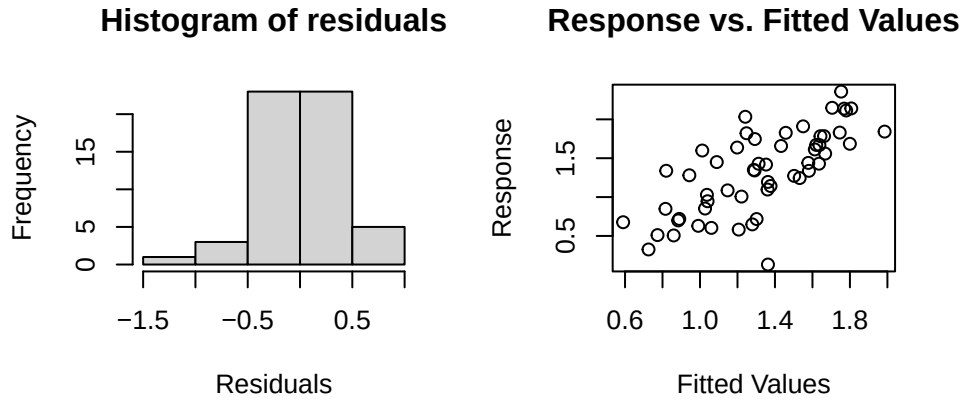

```
par(oldpar)
```

Plankton diversity is highest at low temperatures and at intermediate levels of chlorophyll. Neither of those patterns was uncovered by the linear model analysis.

Diagnostic Plots

```
oldpar <- par(mfrow = c(2,2))
gam.check(shannon_gam_no_low)
```






```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 21 iterations.
#> The RMS GCV score gradient at convergence was 2.198624e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 1.61e+00    1.00    0.43
#> s(Sal)         4.00e+00 2.26e-08    1.07    0.61
#> s(log(Turb))   4.00e+00 1.37e-08    1.35    1.00
#> s(log(Chl))    4.00e+00 3.77e+00    1.10    0.75
#> s(log1p(Fish)) 4.00e+00 3.67e-01    0.96    0.34
#> s(Yearf)       5.00e+00 2.93e+00      NA      NA
par(oldpar)
```

Model looks excellent.

Single Species Models

Model Choice

Our model alternatives are similar to the choices we had for the Total Density model. The problem is, we can't use any of the continuous data distributions in GAMS with zero values (at least relying on the canonical link functions) because (log(0) = -Inf; 1/0 = Inf, 1 / 0*0 = Inf). The easiest solution is to add some finite small quantity to the density data, and predict that. Here we predict log(Density + 1) using Gaussian models.

Automating Analysis of Separate Species

I'm going to automate analysis of all selected species by using a "nested" Tibble. This is a convenient alternative to writing a "for" loop to run multiple identical analyses.

I create a "long" data source, based on the reduced data set that omits low salinity samples.

```
spp_data <- drop_low %>%
  select(Yearf, Season, Station, Temp,
         Sal, Turb, Chl, Fish,
         Acartia, Balanus, Eurytemora, Polychaete, Pseudocal, Temora) %>%
  pivot_longer(-c(Yearf:Fish), names_to = 'Species', values_to = 'Density')
```

Next, I create a function to run the analysis. This function takes a data frame or tibble as an argument. The tibble mush have data columns with the correct names.

The initial model fits for some species had a lot of wiggles in them, to an extent that I thought did not make much scientific sense, so I decided to reduce the dimensionality of the GAM smoothers, by adding the parameter `k= 4`. Lowe numbers constrain the GAM to fit smoother lines.

```
my_gam <- function(.dat) {

  gam(log1p(Density) ~
        s(Temp, bs="ts", k = 5) +
        s(Sal, bs="ts", k = 5) +
        s(log(Turb), bs="ts", k = 5) +
        s(log(Chl), bs="ts", k = 5) +
        s(log1p(Fish),bs="ts", k = 5) +
        s(Yearf, bs = 're'),
      data = .dat, family = "gaussian")
}
```

Next, I create the nested tibble, and conduct the analysis on each species. . . .

```
spp_analysis <- spp_data %>%
  group_by(Species) %>%
  nest() %>%
  mutate(gam_mods = map(data, my_gam))
```

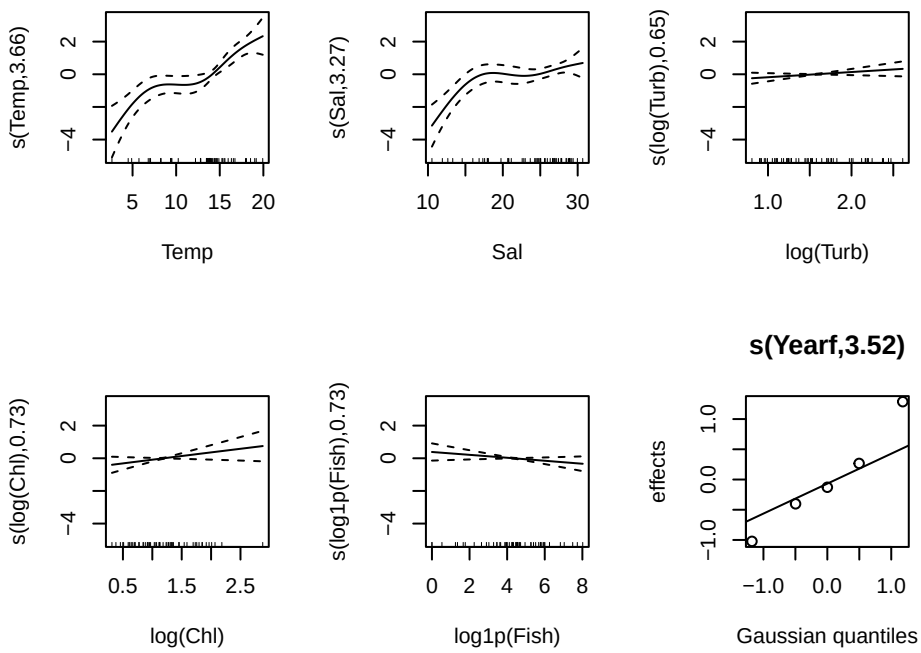
And finally, output the model results. I can do that in a "for" loop, but it's Awkward to look through a long list of output, so I step through each species in turn.

Acartia

```
spp = 'Acartia'
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)    6.598      0.371   17.78   <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(Temp)        3.6631      4 31.950  < 2e-16 ***
#> s(Sal)         3.2713      4  7.570 0.000232 ***
#> s(log(Turb))   0.6538      4  0.637 0.076037 .
#> s(log(Chl))    0.7323      4  1.316 0.055331 .
#> s(log1p(Fish)) 0.7316      4  0.610 0.080622 .
#> s(Yearf)       3.5153      4 11.237 6.14e-07 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.763   Deviance explained = 81.8%
#> GCV = 0.93657  Scale est. = 0.70553   n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                   edf Ref.df      F  p-value
#> s(Temp)        3.6631 4.0000 31.950  < 2e-16
#> s(Sal)         3.2713 4.0000  7.570 0.000232
#> s(log(Turb))   0.6538 4.0000  0.637 0.076037
#> s(log(Chl))    0.7323 4.0000  1.316 0.055331
#> s(log1p(Fish)) 0.7316 4.0000  0.610 0.080622
#> s(Yearf)       3.5153 4.0000 11.237 6.14e-07
```

Plot GAM

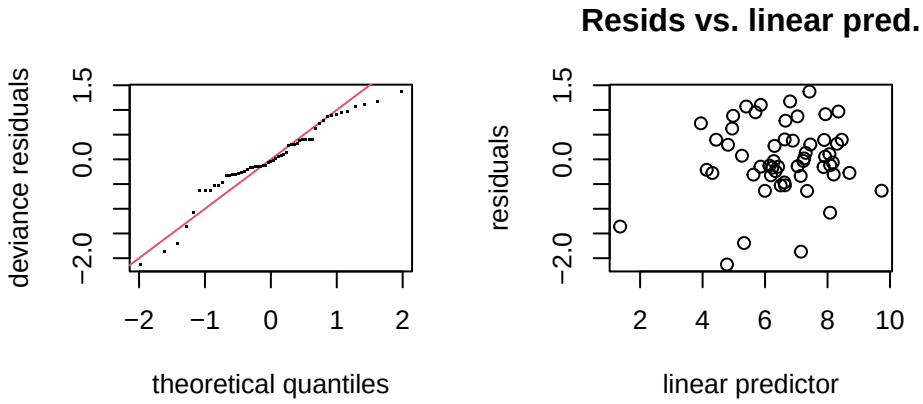
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```

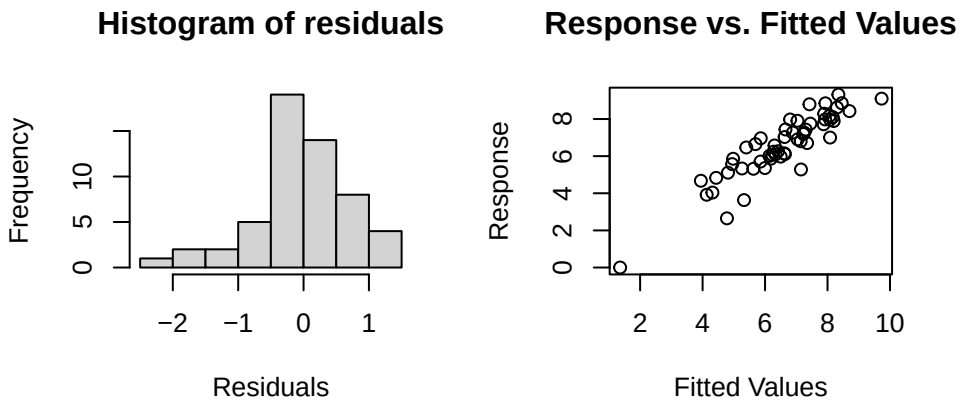



```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```





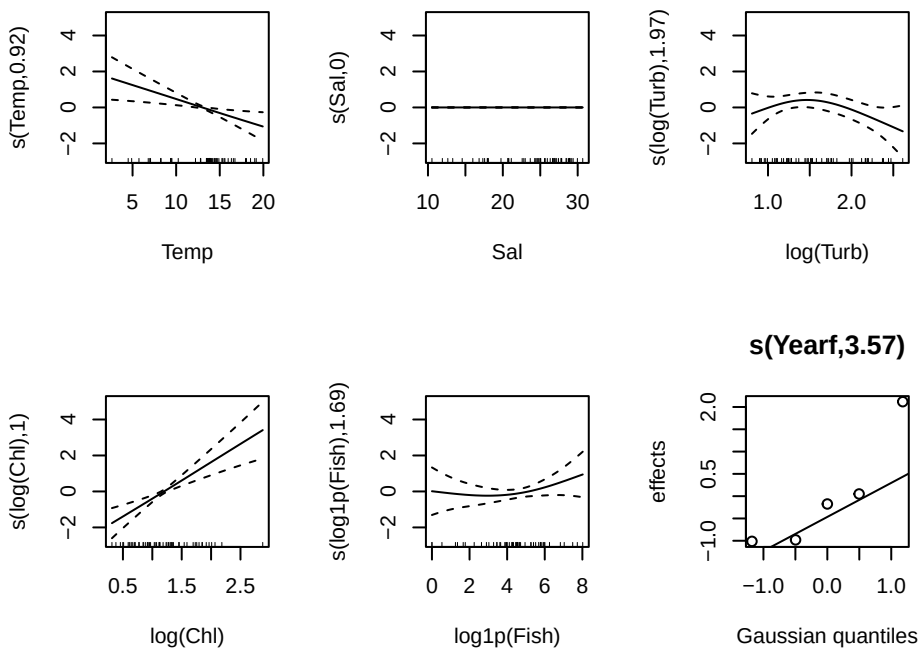

```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 14 iterations.
#> The RMS GCV score gradient at convergence was 1.761436e-06 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                   k'   edf k-index p-value
#> s(Temp)        4.000 3.663    1.28    0.97
#> s(Sal)         4.000 3.271    0.92    0.19
#> s(log(Turb))   4.000 0.654    1.22    0.95
#> s(log(Chl))    4.000 0.732    0.97    0.34
#> s(log1p(Fish)) 4.000 0.732    1.01    0.47
#> s(Yearf)       5.000 3.515      NA      NA
par(oldpar)
```

Balanus

```
spp = 'Balanus'
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   3.6930     0.6478   5.701 8.74e-07 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                       edf Ref.df      F  p-value
#> s(Temp)         9.192e-01      4  2.998  0.00414 **
#> s(Sal)          1.782e-10      4  0.000  0.52552
#> s(log(Turb))    1.967e+00      4  1.779  0.06016 .
#> s(log(Chl))     1.004e+00      4 14.125 2.07e-05 ***
#> s(log1p(Fish)) 1.686e+00      4  0.691  0.22444
#> s(Yearf)        3.568e+00      4  7.912 1.75e-05 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.581   Deviance explained = 65.2%
#> GCV = 2.7021  Scale est. = 2.2038    n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df      F  p-value
#> s(Temp)         9.192e-01 4.000e+00  2.998  0.00414
#> s(Sal)          1.782e-10 4.000e+00  0.000  0.52552
#> s(log(Turb))    1.967e+00 4.000e+00  1.779  0.06016
#> s(log(Chl))     1.004e+00 4.000e+00 14.125 2.07e-05
#> s(log1p(Fish)) 1.686e+00 4.000e+00  0.691  0.22444
#> s(Yearf)        3.568e+00 4.000e+00  7.912 1.75e-05
```

Plot GAM

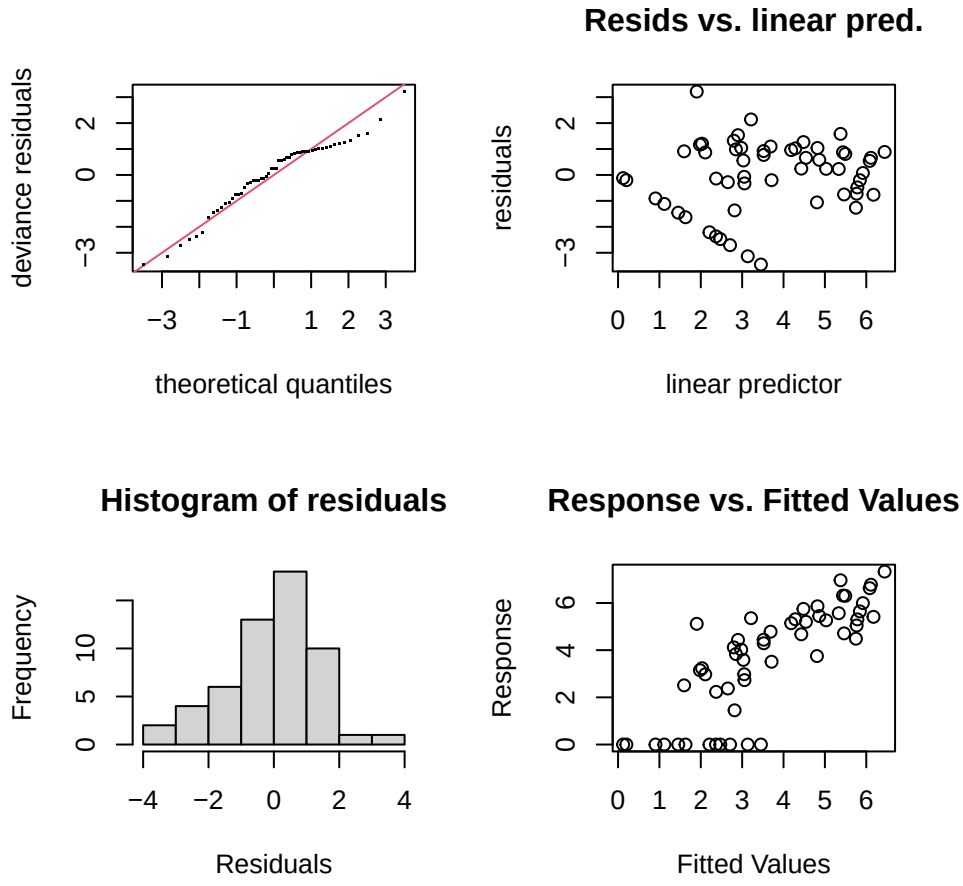
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```




```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```



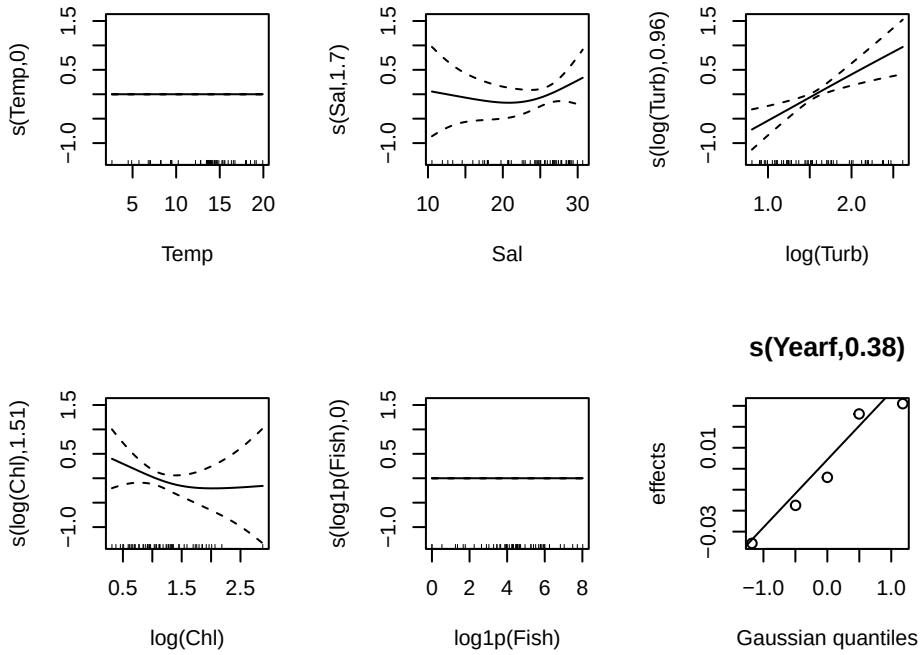

```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 16 iterations.
#> The RMS GCV score gradient at convergence was 2.501697e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 9.19e-01    0.96    0.34
#> s(Sal)         4.00e+00 1.78e-10    0.85    0.09 .
#> s(log(Turb))   4.00e+00 1.97e+00    0.88    0.17
#> s(log(Chl))    4.00e+00 1.00e+00    0.93    0.21
#> s(log1p(Fish)) 4.00e+00 1.69e+00    0.94    0.32
#> s(Yearf)       5.00e+00 3.57e+00      NA      NA
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
par(oldpar)
```

Eurytemora

```
spp =  "Eurytemora"
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   6.5275     0.1297   50.34   <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(Temp)         8.514e-10      4 0.000 0.522777
#> s(Sal)          1.698e+00      4 0.439 0.360992
#> s(log(Turb))    9.561e-01      4 3.326 0.000375 ***
#> s(log(Chl))     1.509e+00      4 0.541 0.241190
#> s(log1p(Fish)) 4.650e-10      4 0.000 0.340527
#> s(Yearf)        3.805e-01      4 0.101 0.368007
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.241   Deviance explained = 30.5%
#> GCV = 0.91936  Scale est. = 0.8267    n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df     F  p-value
#> s(Temp)         8.514e-10 4.000e+00 0.000 0.522777
#> s(Sal)          1.698e+00 4.000e+00 0.439 0.360992
#> s(log(Turb))    9.561e-01 4.000e+00 3.326 0.000375
#> s(log(Chl))     1.509e+00 4.000e+00 0.541 0.241190
#> s(log1p(Fish)) 4.650e-10 4.000e+00 0.000 0.340527
#> s(Yearf)        3.805e-01 4.000e+00 0.101 0.368007
```

Plot GAM

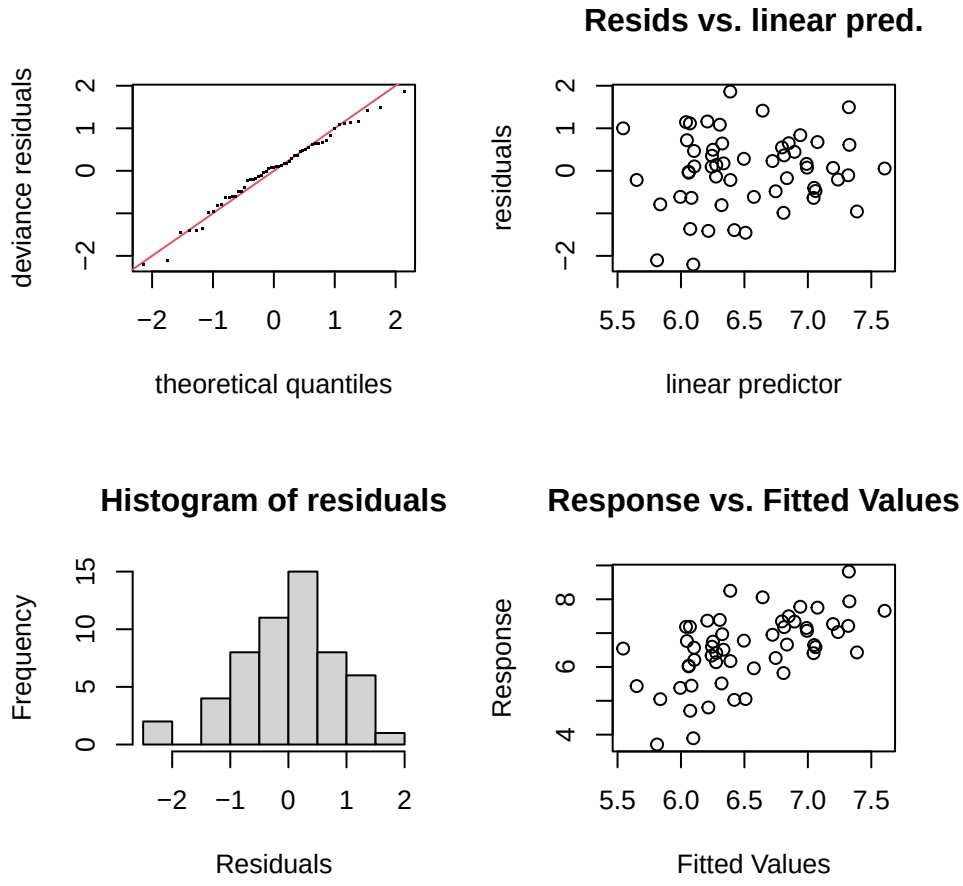
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```




```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```



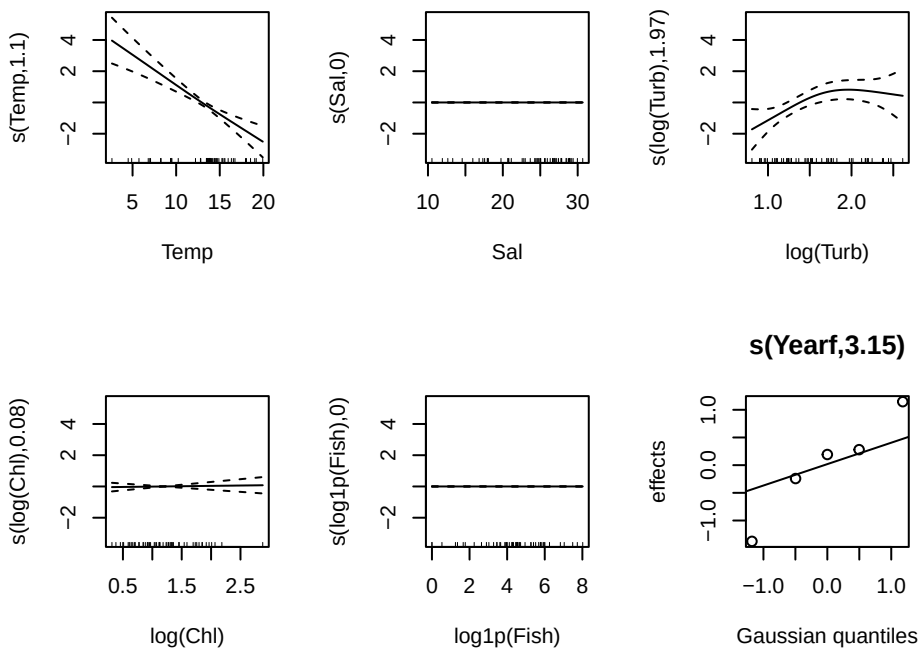

```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 17 iterations.
#> The RMS GCV score gradient at convergence was 1.009452e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 8.51e-10    1.06    0.60
#> s(Sal)         4.00e+00 1.70e+00    0.96    0.35
#> s(log(Turb))   4.00e+00 9.56e-01    0.91    0.20
#> s(log(Chl))    4.00e+00 1.51e+00    1.00    0.36
#> s(log1p(Fish)) 4.00e+00 4.65e-10    1.08    0.66
#> s(Yearf)       5.00e+00 3.80e-01      NA      NA
par(oldpar)
```

Polychaete

```
spp =  "Polychaete"
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)    3.080      0.537   5.736 6.46e-07 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                       edf Ref.df      F  p-value
#> s(Temp)         1.097e+00      4 10.566 6.27e-07 ***
#> s(Sal)          4.128e-10      4  0.000  0.51023
#> s(log(Turb))    1.970e+00      4  3.399  0.00422 **
#> s(log(Chl))     8.222e-02      4  0.024  0.29937
#> s(log1p(Fish)) 4.562e-10      4  0.000  0.67017
#> s(Yearf)        3.153e+00      4  3.277  0.00506 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.525   Deviance explained =   58%
#> GCV = 3.6818  Scale est. = 3.1929    n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df      F  p-value
#> s(Temp)         1.097e+00 4.000e+00 10.566 6.27e-07
#> s(Sal)          4.128e-10 4.000e+00  0.000  0.51023
#> s(log(Turb))    1.970e+00 4.000e+00  3.399  0.00422
#> s(log(Chl))     8.222e-02 4.000e+00  0.024  0.29937
#> s(log1p(Fish)) 4.562e-10 4.000e+00  0.000  0.67017
#> s(Yearf)        3.153e+00 4.000e+00  3.277  0.00506
```

Plot GAM

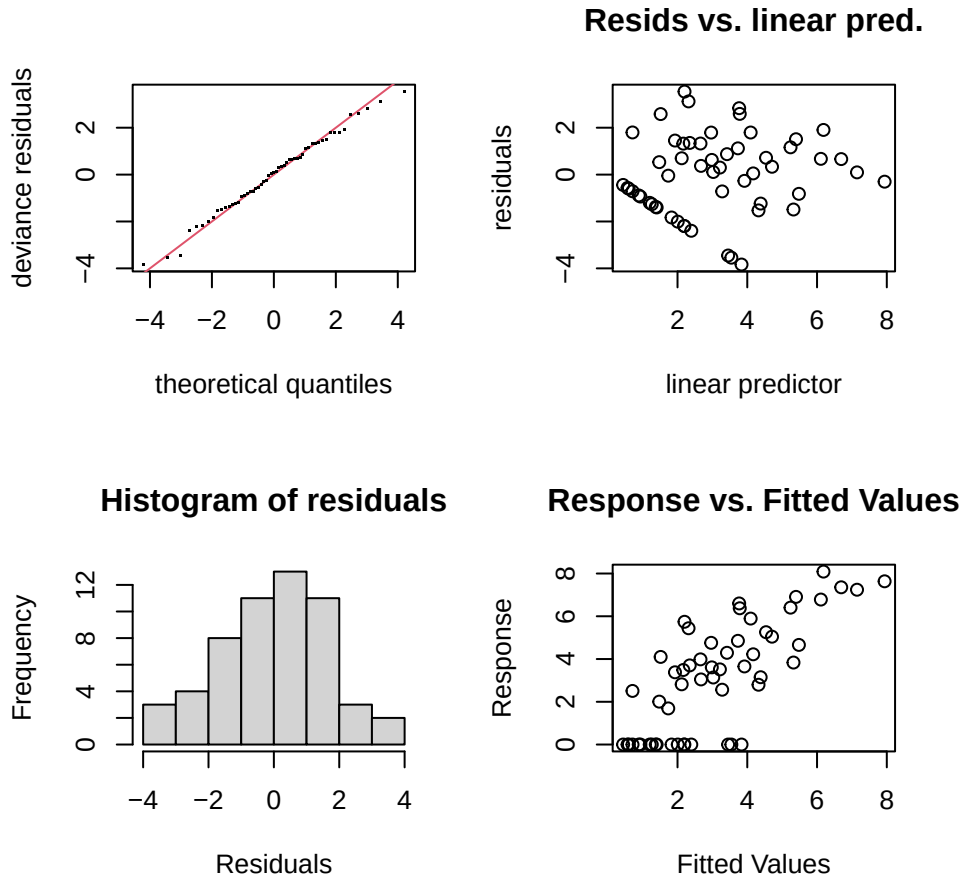
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```




```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```



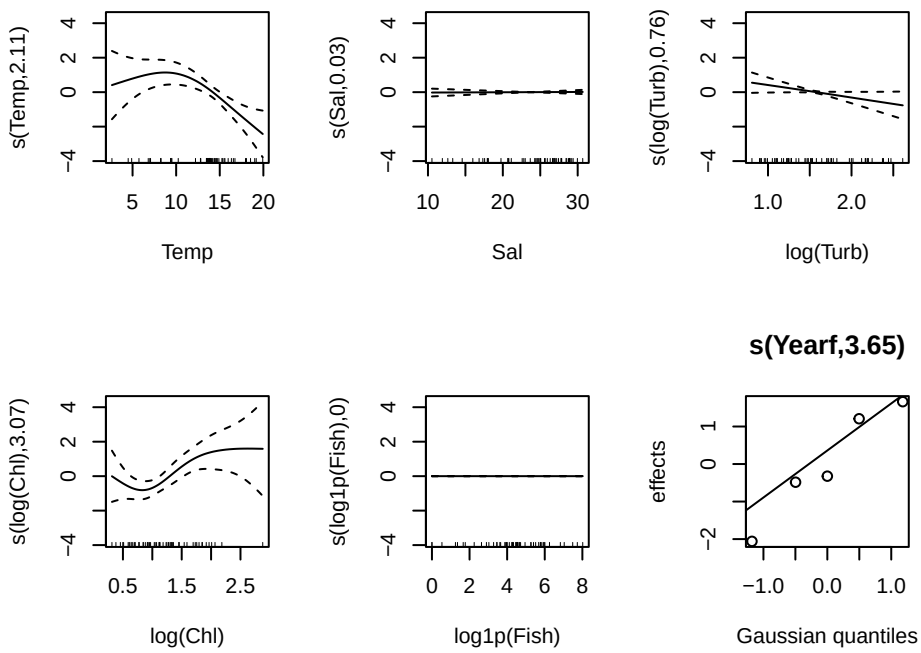

```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 20 iterations.
#> The RMS GCV score gradient at convergence was 2.115525e-07 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 1.10e+00    0.93    0.27
#> s(Sal)         4.00e+00 4.13e-10    1.05    0.55
#> s(log(Turb))   4.00e+00 1.97e+00    0.88    0.15
#> s(log(Chl))    4.00e+00 8.22e-02    0.86    0.08 .
#> s(log1p(Fish)) 4.00e+00 4.56e-10    1.10    0.79
#> s(Yearf)       5.00e+00 3.15e+00      NA      NA
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
par(oldpar)
```

Pseudocal

```
spp =  "Pseudocal"
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   4.6181     0.7274   6.349    1e-07 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                       edf Ref.df     F  p-value
#> s(Temp)         2.115e+00      4 9.700 0.000252 ***
#> s(Sal)          2.915e-02      4 0.009 0.247693
#> s(log(Turb))    7.645e-01      4 1.173 0.029377 *
#> s(log(Chl))     3.065e+00      4 8.038 0.007095 **
#> s(log1p(Fish)) 2.068e-09      4 0.000 0.919182
#> s(Yearf)        3.655e+00      4 8.628 9.41e-06 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.553   Deviance explained = 63.3%
#> GCV = 2.5172  Scale est. = 2.0308    n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df     F  p-value
#> s(Temp)         2.115e+00 4.000e+00 9.700 0.000252
#> s(Sal)          2.915e-02 4.000e+00 0.009 0.247693
#> s(log(Turb))    7.645e-01 4.000e+00 1.173 0.029377
#> s(log(Chl))     3.065e+00 4.000e+00 8.038 0.007095
#> s(log1p(Fish)) 2.068e-09 4.000e+00 0.000 0.919182
#> s(Yearf)        3.655e+00 4.000e+00 8.628 9.41e-06
```

Plot GAM

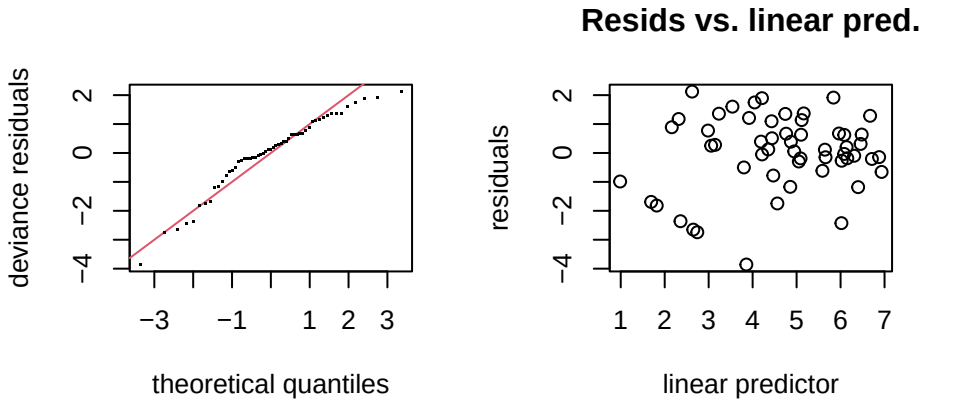
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```

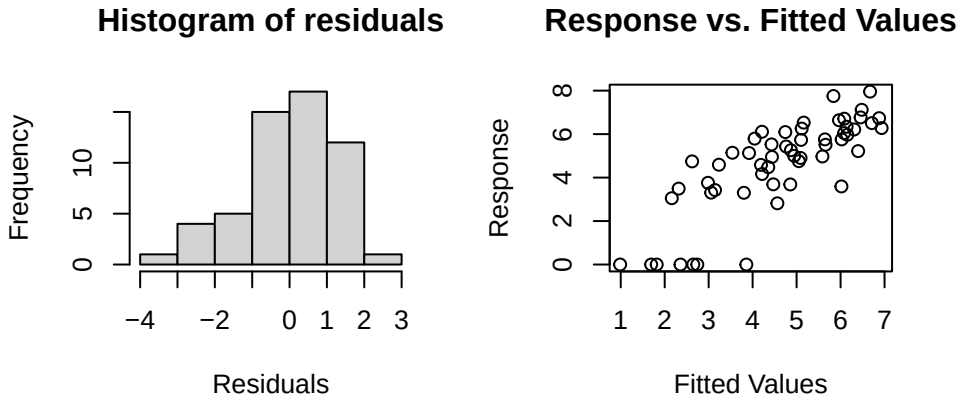



```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```





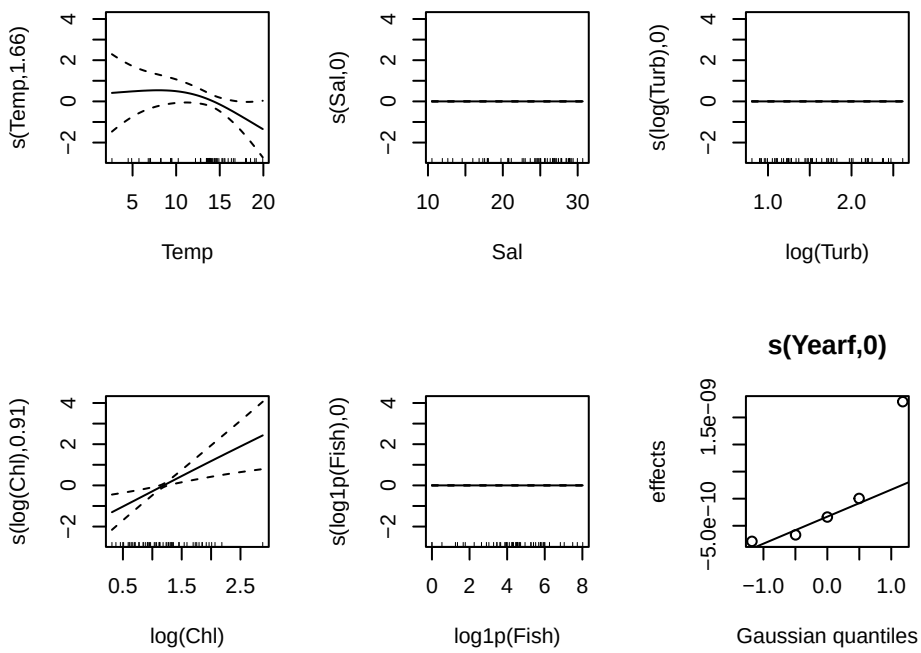

```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 23 iterations.
#> The RMS GCV score gradient at convergence was 9.518261e-08 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 2.11e+00    1.21    0.91
#> s(Sal)         4.00e+00 2.91e-02    1.04    0.60
#> s(log(Turb))   4.00e+00 7.64e-01    1.03    0.47
#> s(log(Chl))    4.00e+00 3.07e+00    1.13    0.76
#> s(log1p(Fish)) 4.00e+00 2.07e-09    0.97    0.37
#> s(Yearf)       5.00e+00 3.65e+00      NA      NA
par(oldpar)
```

Temora

```
spp =  "Temora"
mod <- spp_analysis$gam_mods[spp_analysis$Species == spp][[1]]
summary(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)   1.9074     0.2491   7.658 4.75e-10 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Approximate significance of smooth terms:
#>                       edf Ref.df     F p-value
#> s(Temp)         1.662e+00      4 1.076 0.07506 .
#> s(Sal)          2.580e-10      4 0.000 0.41314
#> s(log(Turb))    1.261e-10      4 0.000 0.70764
#> s(log(Chl))     9.065e-01      4 2.425 0.00141 **
#> s(log1p(Fish)) 1.918e-10      4 0.000 0.77314
#> s(Yearf)        7.146e-09      4 0.000 0.39809
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> R-sq.(adj) =  0.194   Deviance explained = 23.2%
#> GCV = 3.6492  Scale est. = 3.4124    n = 55
cat('\n')
anova(mod)
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> log1p(Density) ~ s(Temp, bs = "ts", k = 5) + s(Sal, bs = "ts",
#>     k = 5) + s(log(Turb), bs = "ts", k = 5) + s(log(Chl), bs = "ts",
#>     k = 5) + s(log1p(Fish), bs = "ts", k = 5) + s(Yearf, bs = "re")
#>
#> Approximate significance of smooth terms:
#>                       edf    Ref.df     F p-value
#> s(Temp)         1.662e+00 4.000e+00 1.076 0.07506
#> s(Sal)          2.580e-10 4.000e+00 0.000 0.41314
#> s(log(Turb))    1.261e-10 4.000e+00 0.000 0.70764
#> s(log(Chl))     9.065e-01 4.000e+00 2.425 0.00141
#> s(log1p(Fish)) 1.918e-10 4.000e+00 0.000 0.77314
#> s(Yearf)        7.146e-09 4.000e+00 0.000 0.39809
```

Plot GAM

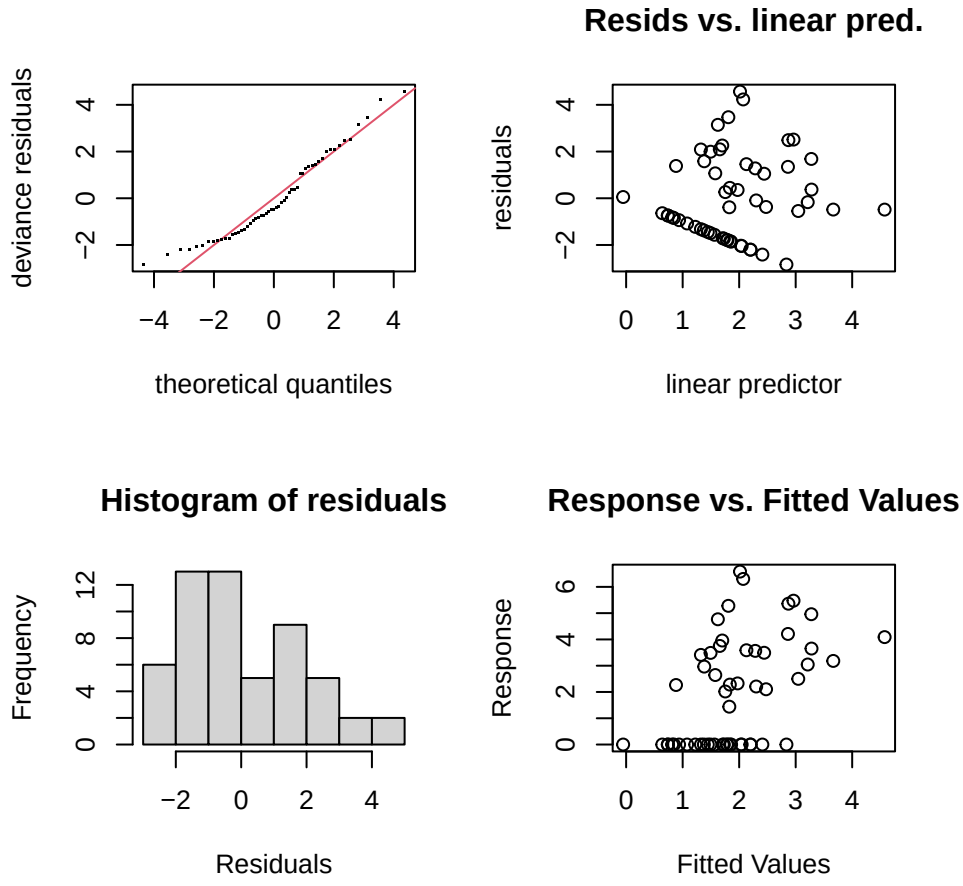
```
oldpar <- par(mfrow = c(2,3))
plot(mod)
```




```
par(oldpar)
```

Model Diagnostics

```
oldpar <- par(mfrow = c(2,2))
gam.check(mod)
```




```
#>
#> Method: GCV   Optimizer: magic
#> Smoothing parameter selection converged after 18 iterations.
#> The RMS GCV score gradient at convergence was 1.91419e-06 .
#> The Hessian was positive definite.
#> Model rank =  26 / 26
#>
#> Basis dimension (k) checking results. Low p-value (k-index<1) may
#> indicate that k is too low, especially if edf is close to k'.
#>
#>                     k'      edf k-index p-value
#> s(Temp)        4.00e+00 1.66e+00    1.12    0.74
#> s(Sal)         4.00e+00 2.58e-10    1.18    0.90
#> s(log(Turb))   4.00e+00 1.26e-10    1.20    0.90
#> s(log(Chl))    4.00e+00 9.07e-01    1.03    0.48
#> s(log1p(Fish)) 4.00e+00 1.92e-10    0.84    0.12
#> s(Yearf)       5.00e+00 7.15e-09      NA      NA
par(oldpar)
```