Mixed Effects Linear Models to Analyze Plankton Comunity Data

Curtis C. Bohlen, Casco Bay Estuary Partnership

7/21/2022

Contents

Introduction

This notebook reprises analyses using mixed model linear models instead of GAMs. The goal is to lower complexity of models, and gain access to some tools for automated model selection.

In practice, what I found using GAM analyses was that they seldom fit relationships that were not linear between predictors and response, so we gained little benefit from the added complexity of using GAM models. GAMs are complex to fit, especially (as here) when predictor variables are interrelated.

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

- Station

- 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)
- Sample Event (inclusion as a random factor often led to overspecified models and it seldom proved important)

I emphasize hierarchical linear models here because there are robust automated tools for stepwise model selection on linear hierarchical models, and the logic of applying consideration of collinearity is less confusing than the logic of applying "concurvity" to GAM models. Almost all GAM models we fit to the plankton data end up fitting only linear relationships anyway, so there should be little loss in model accuracy.

Load Libraries

```
library(lmerTest)  # Automatically loads lme4
#> Loading required package: lme4
#> Loading required package: Matrix
#>
#> Attaching package: 'lmerTest'
#> The following object is masked from 'package:lme4':
#>
#>     lmer
#> The following object is masked from 'package:stats':
#>
#>     step
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 tidyr::expand() masks Matrix::expand()
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
#> x tidyr::pack()   masks Matrix::pack()
#> x tidyr::unpack() masks Matrix::unpack()
library(readxl)
library(car)
#> Loading required package: carData
#>
#> Attaching package: 'car'
#> The following object is masked from 'package:dplyr':
#>
#>     recode
#> The following object is masked from 'package:purrr':
#>
#>     some
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()`.

```r
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.

```r
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
```

Model of Fish Abundance

```
fish_lmer <- lmer(log1p(Fish) ~ Station +
                     Temp +
                     Sal +
                     log(Turb) +
                     log(Chl) +
                     log1p(Density) +
                     (1 | Yearf),
                   data = base_data, na.action = na.omit,
                 REML = TRUE)
#> boundary (singular) fit: see help('isSingular')
summary(fish_lmer)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula:
#> log1p(Fish) ~ Station + Temp + Sal + log(Turb) + log(Chl) + log1p(Density) +
#>     (1 | Yearf)
#>    Data: base_data
#>
#> REML criterion at convergence: 229.9
#>
#> Scaled residuals:
#>      Min      1Q   Median       3Q      Max
#> -2.21140 -0.53493  0.00498  0.68506  2.39279
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.00     0.000
#>  Residual             3.28     1.811
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>                Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept)     1.73121    2.26961 49.00000   0.763    0.449
#> Station2       -1.12929    1.08611 49.00000  -1.040    0.304
#> Station3       -1.40461    0.98480 49.00000  -1.426    0.160
#> Station4       -1.38003    1.07274 49.00000  -1.286    0.204
#> Temp           -0.07888    0.08813 49.00000  -0.895    0.375
#> Sal             0.02158    0.05846 49.00000   0.369    0.714
#> log(Turb)      -0.05450    0.64021 49.00000  -0.085    0.933
#> log(Chl)       -0.12674    0.54168 49.00000  -0.234    0.816
#> log1p(Density)  0.53607    0.28616 49.00000   1.873    0.067 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.148
#> Station3   -0.229  0.759
#> Station4   -0.292  0.756  0.774
#> Temp       -0.331  0.578  0.581  0.601
#> Sal        -0.039 -0.675 -0.549 -0.542 -0.328
```

```
#> log(Turb)    -0.353 -0.109  0.143  0.186  0.202  0.318
#> log(Chl)      0.064 -0.206 -0.166 -0.277 -0.362 -0.092 -0.018
#> lg1p(Dnsty) -0.576  0.062 -0.055  0.005 -0.172 -0.285 -0.396 -0.027
#> optimizer (nloptwrap) convergence code: 0 (OK)
#> boundary (singular) fit: see help('isSingular')
```

Note that the impact of the Year random factor is so small that `lmer()` fit it as having exactly zero variance. But the term is still in the model, and the model includes adjustments for annual means. That still helps reduce model variance. In effect, the model is the equivalent of a linear model that includes Yearf as a factor.

```
anova(fish_lmer)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>                Sum Sq Mean Sq NumDF DenDF F value  Pr(>F)
#> Station        7.0771  2.3590     3    49  0.7193 0.54527
#> Temp           2.6275  2.6275     1    49  0.8011 0.37513
#> Sal            0.4468  0.4468     1    49  0.1362 0.71365
#> log(Turb)      0.0238  0.0238     1    49  0.0072 0.93251
#> log(Chl)       0.1796  0.1796     1    49  0.0547 0.81598
#> log1p(Density) 11.5100 11.5100    1    49  3.5095 0.06699 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Stepwise Model Selection

The `lmerTest` package includes a backward elimination algorithm that first searches for random effects that provide little explanatory power (by likelihood ratio test), then for fixed effects that can be dropped.

```
(fish_step_0 <- step(fish_lmer,
                 #reduce.random = FALSE,  # add to not drop random terms
                 ))
#> Backward reduced random-effect table:
#>
#>            Eliminated npar  logLik    AIC LRT Df Pr(>Chisq)
#> <none>                  11 -114.95 251.9
#> (1 | Yearf)          1  10 -114.95 249.9   0  1          1
#>
#> Backward reduced fixed-effect table:
#>                Eliminated Df Sum of Sq    RSS     AIC F value  Pr(>F)
#> log(Turb)               1  1    0.0238 160.73  75.118  0.0072 0.93251
#> log(Chl)                2  1    0.1819 160.91  73.184  0.0566 0.81292
#> Sal                     3  1    0.5200 161.43  71.371  0.1648 0.68647
#> Station                 4  3   10.9333 172.36  69.172  1.1739 0.32863
#> Temp                    5  1    0.3385 172.70  67.285  0.1080 0.74368
#> log1p(Density)          6  1   11.3703 184.07  68.984  3.6869 0.05994 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Model found:
#> log1p(Fish) ~ 1
fish_step_0 <- get_model(fish_step_0)
```

Stepwise elimination takes us to a model that lacks **any** predictors. In essence, the best prediction we can come up with for the abundance of fish based on AIC is.... The mean abundance of fish.

Fish abundance is associated with zooplankton density, with higher fish abundance occurring with higher plankton density, but the association hovers at the edge of statistical significance, depending on the details of the model.

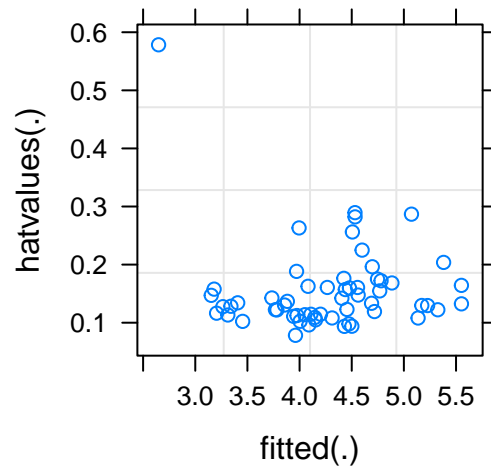## Model Diagnostics (Full Model)

```
oldpar <- par(mfrow = c(2,2))
plot(fish_lmer, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
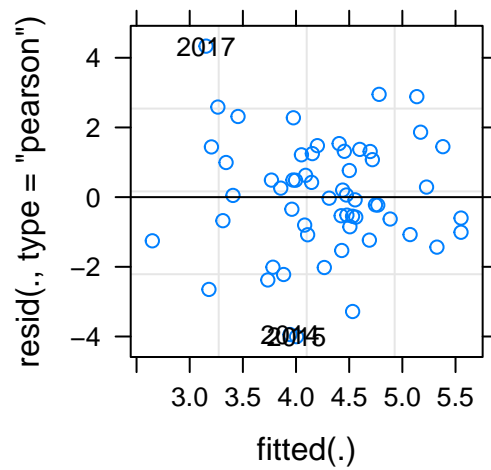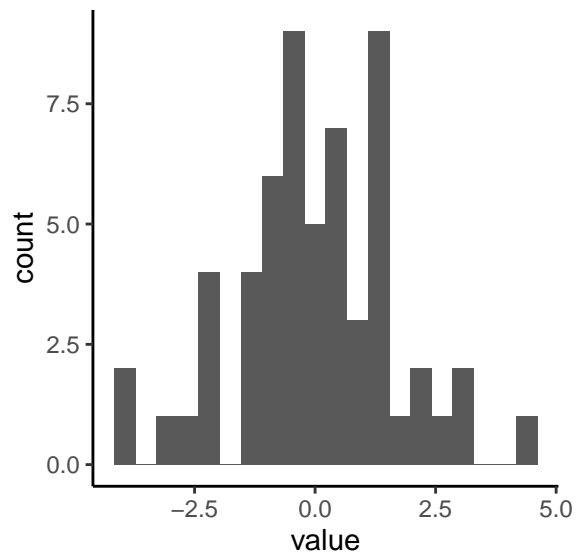


```
plot(fish_lmer, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
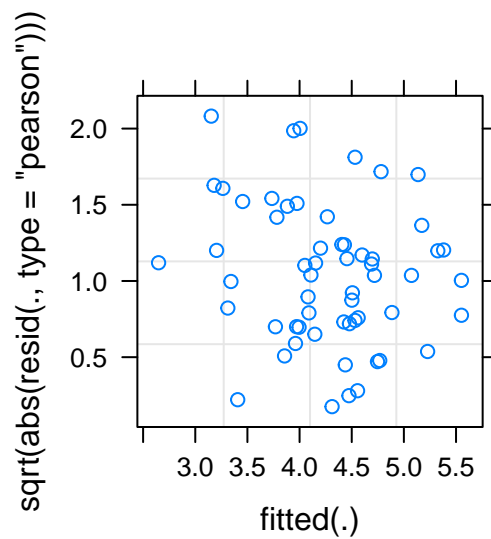
```
plot(fish_lmer, hatvalues(.) ~ fitted(.))
```



```
as_tibble(resid(fish_lmer)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```



```
par(oldpar)
```

The Regression model has one high leverage point. As we have come to expect, that is a high discharge, low salinity spring sample.

```
complete_data[which(hatvalues(fish_lmer)>.5),]
#> # A tibble: 1 x 18
#>   Season Station Yearf is_sp_up  Temp   Sal  Turb   Chl  Fish    RH Density
#>   <fct>  <fct>   <fct> <lgl>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
```

```
#> 1 Spring 1      2014  TRUE     7.33  0.03  7.4  1.31  3.03    0    20.3
#> # ... with 7 more variables: H <dbl>, Acartia <dbl>, Balanus <dbl>,
#> #   Eurytemora <dbl>, Polychaete <dbl>, Pseudocal <dbl>, Temora <dbl>
```

The impact of the high leverage term is obvioous in a graphic.

Graphic Review of Data

There is a lot of noise in this relationship.

```
base_data %>%
  ggplot(aes(log1p(Density), log1p(Fish))) +
  geom_point(aes(color = Station), size = 2) +
  geom_smooth(method = 'lm', formula = y~x) +
  facet_wrap(~Season)
#> Warning: Removed 2 rows containing non-finite values (stat_smooth).
#> Warning: Removed 2 rows containing missing values (geom_point).
```
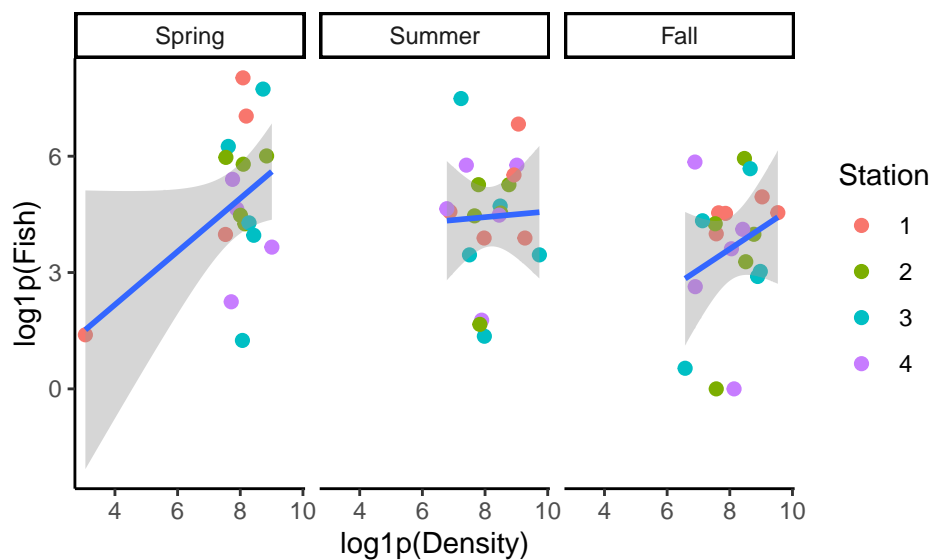

Model on Reduced Data

```
fish_lmer_no_low <- lmer(log1p(Fish) ~ Station +
                     Temp +
                     Sal +
                     log(Turb) +
                     log(Chl) +
                     log1p(Density) +
                     (1 | Yearf),
                   data = drop_low, na.action = na.omit, REML = FALSE)
#> boundary (singular) fit: see help('isSingular')
summary(fish_lmer_no_low)
#> Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
```

```
#>    method [lmerModLmerTest]
#> Formula:
#> log1p(Fish) ~ Station + Temp + Sal + log(Turb) + log(Chl) + log1p(Density) +
#>      (1 | Yearf)
#>    Data: drop_low
#>
#>      AIC      BIC   logLik deviance df.resid
#>    234.9    257.0   -106.4    212.9       44
#>
#> Scaled residuals:
#>      Min       1Q   Median       3Q      Max
#> -2.35213 -0.52015  0.00501  0.70812  2.29842
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.000    0.000
#>  Residual             2.809    1.676
#> Number of obs: 55, groups:  Yearf, 5
#>
#> Fixed effects:
#>                 Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept)      5.14893    3.13585 55.00000   1.642    0.106
#> Station2        -1.19974    1.00752 55.00000  -1.191    0.239
#> Station3        -1.44516    0.91655 55.00000  -1.577    0.121
#> Station4        -1.48029    1.00059 55.00000  -1.479    0.145
#> Temp            -0.08730    0.08176 55.00000  -1.068    0.290
#> Sal             -0.01576    0.06304 55.00000  -0.250    0.803
#> log(Turb)        0.13584    0.63759 55.00000   0.213    0.832
#> log(Chl)         0.04070    0.52963 55.00000   0.077    0.939
#> log1p(Density)   0.18745    0.36664 55.00000   0.511    0.611
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.139
#> Station3   -0.183  0.756
#> Station4   -0.252  0.753  0.776
#> Temp       -0.271  0.578  0.579  0.600
#> Sal        -0.289 -0.573 -0.449 -0.428 -0.248
#> log(Turb)  -0.086 -0.111  0.156  0.189  0.177  0.230
#> log(Chl)    0.169 -0.192 -0.158 -0.268 -0.360 -0.231  0.015
#> lg1p(Dnsty) -0.763  0.086 -0.029  0.039 -0.084 -0.009 -0.446 -0.098
#> optimizer (nloptwrap) convergence code: 0 (OK)
#> boundary (singular) fit: see help('isSingular')
```

And the resulting model shows no signs of a statistically significant connection to any predictors.

Stepwise Evaluation of Submodels

```
(fish_step_1 <- step(fish_lmer_no_low,
                #reduce.random = FALSE,  # add to not drop random terms
                ))
```

```
#> Backward reduced random-effect table:
#>
#>              Eliminated npar  logLik    AIC        LRT Df Pr(>Chisq)
#> <none>                  11 -106.44 234.88
#> (1 | Yearf)         1  10 -106.44 232.88 5.6843e-14  1          1
#>
#> Backward reduced fixed-effect table:
#>              Eliminated Df Sum of Sq     RSS    AIC F value Pr(>F)
#> log(Chl)            1  1    0.0166 154.49 72.804  0.0049 0.9443
#> log(Turb)           2  1    0.1262 154.62 70.849  0.0384 0.8455
#> Sal                 3  1    0.2498 154.87 68.937  0.0776 0.7818
#> log1p(Density)      4  1    1.4419 156.31 67.447  0.4562 0.5026
#> Temp                5  1    5.8390 162.15 67.464  1.8678 0.1778
#> Station             6  3   13.3372 175.48 65.812  1.3983 0.2540
#>
#> Model found:
#> log1p(Fish) ~ 1
fish_step_1 <- get_model(fish_step_1)
```

Again, the final model includes no significant association between any of the predictors and fish abundance.

Model of Zooplankton Density

```
density_lmer <- lmer(log(Density) ~
                       Station +
                       Temp +
                       Sal +
                       log(Turb) +
                       log(Chl) +
                       log1p(Fish) +
                       (1 | Yearf),
                     data = complete_data, na.action = na.omit)
anova(density_lmer)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>            Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
#> Station    1.7574  0.5858     3 45.805  1.0436 0.382301
#> Temp       0.0143  0.0143     1 48.590  0.0254 0.874002
#> Sal        5.2955  5.2955     1 46.457  9.4340 0.003555 **
#> log(Turb)  3.3045  3.3045     1 46.166  5.8870 0.019216 *
#> log(Chl)   0.6516  0.6516     1 48.907  1.1608 0.286586
#> log1p(Fish) 1.7119 1.7119     1 45.623  3.0498 0.087479 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Salinity and Turbidity are significant predictors.

Stepwise Model Selection

```
(density_step_0 <- step(density_lmer,
                    #reduce.random = FALSE,  # add to not drop random terms
                 ))
#> Backward reduced random-effect table:
#>
#>            Eliminated npar   logLik    AIC     LRT Df Pr(>Chisq)
#> <none>                  11 -75.937 173.88
#> (1 | Yearf)          0   10 -80.160 180.32 8.4451  1    0.00366 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Backward reduced fixed-effect table:
#> Degrees of freedom method: Satterthwaite
#>
#>            Eliminated Sum Sq Mean Sq NumDF  DenDF F value     Pr(>F)
#> Temp                1 0.0143  0.0143    1 48.590  0.0254 0.8740017
#> Station             2 2.5984  0.8661    3 46.428  1.5781 0.2073682
#> log(Chl)            3 0.9552  0.9552    1 51.635  1.6695 0.2020762
#> log1p(Fish)         4 2.2109  2.2109    1 50.491  3.7945 0.0569903 .
#> Sal                 0 8.5973  8.5973    1 51.682 14.0309 0.0004539 ***
#> log(Turb)           0 3.9984  3.9984    1 51.652  6.5255 0.0136227 *
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Model found:
#> log(Density) ~ Sal + log(Turb) + (1 | Yearf)
density_step_0 <- get_model(density_step_0)
```

So, the stepwise process retains Year, salinity and turbidity as predictors of zooplankton density.

Model Diagnostics (Full Model)

```
plot(density_lmer, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
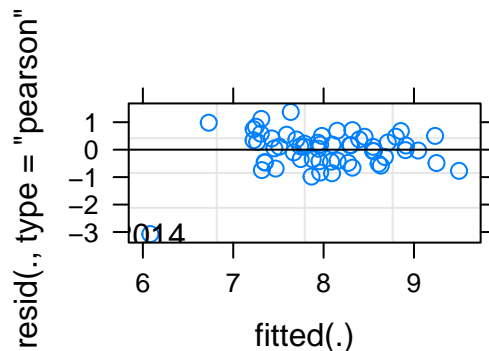
```
plot(density_lmer, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
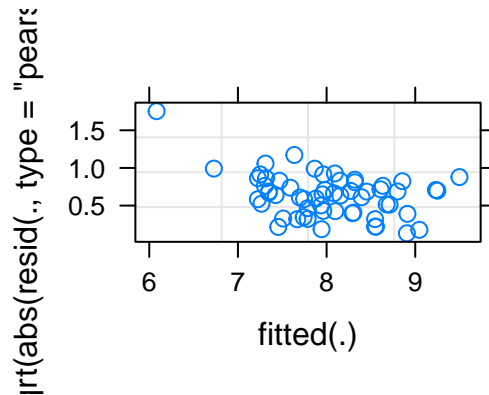


```
plot(density_lmer, hatvalues(.) ~ fitted(.))
```
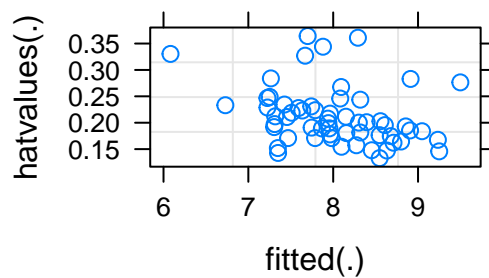


```
as_tibble(resid(density_lmer)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
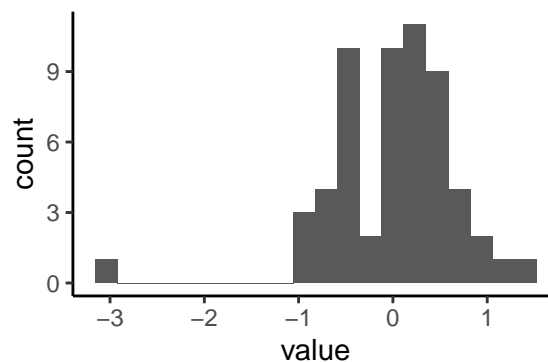

This time we have no very high leverage points, although some are on the high side for a model with this many parameters. But we have one obvious outlier, from 2014.

```
outlier <- which(resid(density_lmer) < -3)
complete_data[outlier,]
#> # A tibble: 1 x 18
```

```
#>   Season Station Yearf is_sp_up   Temp   Sal   Turb   Chl  Fish    RH Density
#>   <fct>  <fct>   <fct> <lgl>     <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>   <dbl>
#> 1 Spring 1       2014  TRUE       7.33  0.03   7.4  1.31  3.03     0    20.3
#> # ... with 7 more variables: H <dbl>, Acartia <dbl>, Balanus <dbl>,
#> #   Eurytemora <dbl>, Polychaete <dbl>, Pseudocal <dbl>, Temora <dbl>
```

The outlier is a spring, low salinity sample from Station 1.

Graphic Review of Data

```
base_data %>%
  ggplot(aes(Sal, log1p(Density))) +
  geom_point(aes(color = Station), size = 2) +
  geom_smooth(method = 'lm', formula = y~x) +
  facet_wrap(~Season)
```
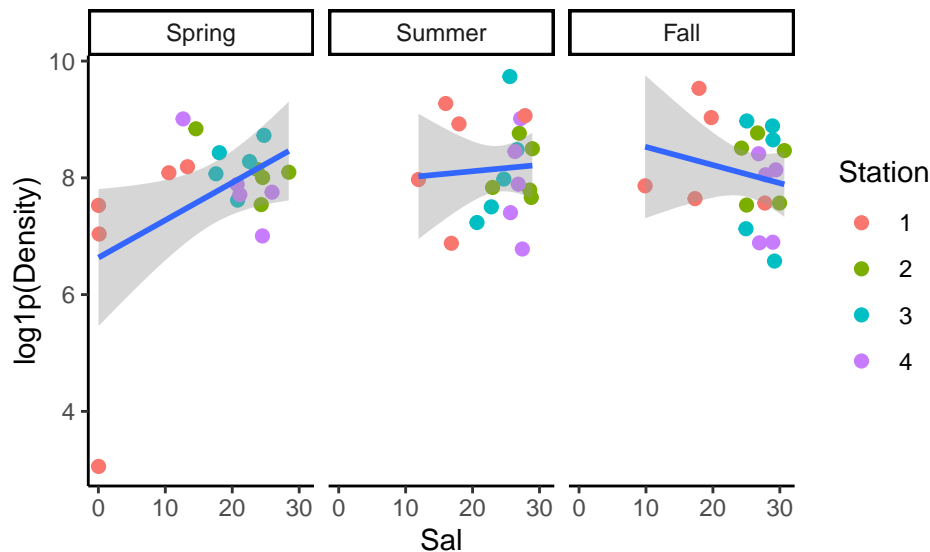

Model on Reduced Data

```
density_lmer_no_low <- lmer(log(Density) ~
                       Station +
                       Temp +
                       Sal +
                       log(Turb) +
                       log(Chl) +
                       log1p(Fish) +
                       (1 | Yearf),
                    data = drop_low, na.action = na.omit)
anova(density_lmer_no_low)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>          Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
```

```
#> Station      2.14305 0.71435    3 42.367  3.5613 0.021920 *
#> Temp         0.48304 0.48304    1 43.717  2.4081 0.127916
#> Sal          0.10945 0.10945    1 42.282  0.5457 0.464179
#> log(Turb)    1.52413 1.52413    1 42.642  7.5983 0.008562 **
#> log(Chl)     1.36545 1.36545    1 44.233  6.8073 0.012340 *
#> log1p(Fish)  0.00056 0.00056    1 42.143  0.0028 0.957925
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This model produces slightly different results from the full data. In the absence of the handful of low salinity samples, salinity no longer provides much information about plankton density. Instead, both Turbidity and Chlorophyll appear important.

Stepwise Model Selection

Stepwise model selection confirms that conclusion.

```
(density_step_no_low <- step(density_lmer_no_low))
#> Backward reduced random-effect table:
#>
#>             Eliminated npar  logLik    AIC    LRT Df Pr(>Chisq)
#> <none>                 11 -50.479 122.96
#> (1 | Yearf)        0   10 -63.564 147.13 26.17  1  3.126e-07 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Backward reduced fixed-effect table:
#> Degrees of freedom method: Satterthwaite
#>
#>             Eliminated  Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
#> log1p(Fish)         1 0.00056 0.00056     1 42.143   0.0028 0.957925
#> Sal                 2 0.10972 0.10972     1 43.286   0.5599 0.458356
#> Temp                3 0.43029 0.43029     1 45.856   2.2157 0.143462
#> Station             0 1.92806 0.64269     3 45.099   3.1840 0.032696 *
#> log(Turb)           0 2.17855 2.17855     1 45.434  10.7929 0.001968 **
#> log(Chl)            0 1.29550 1.29550     1 45.906   6.4182 0.014776 *
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Model found:
#> log(Density) ~ Station + log(Turb) + log(Chl) + (1 | Yearf)
density_step_no_low <- get_model(density_step_no_low)
```

Graphic Summaries

Compare Stations

```
drop_low %>%
  ggplot(aes(Station, log1p(Density))) +
```

```
  stat_summary(size = 1, color = 'grey50') +
  geom_point(aes(color = Season), size = 2, alpha = .75)
#> No summary function supplied, defaulting to `mean_se()`
```

Data (dropping low salinity samples)
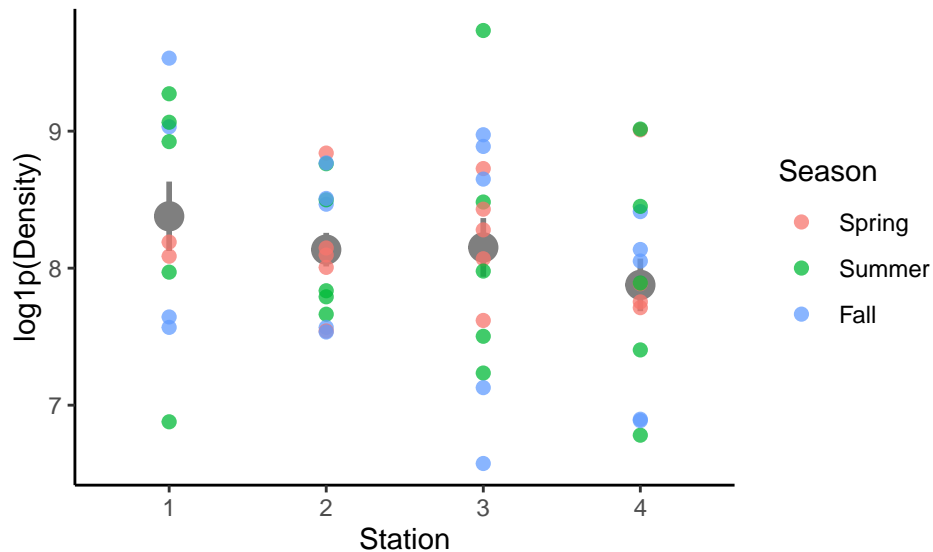
Marginal (Modeled) Means This shows comparisons of "marginal", "least squares" or "adjusted" means. These are modeled means, estimated for what would be seen at average levels of all the other predictors. They are a product not only of the data but also of the model. Error bars are 95% confidence intervals.

```
station_emms <- emmeans(density_lmer_no_low, ~Station, type = 'response')
plot(station_emms) +
  coord_flip() +
  xlab('Predicted Zooplankton Abundance')
```
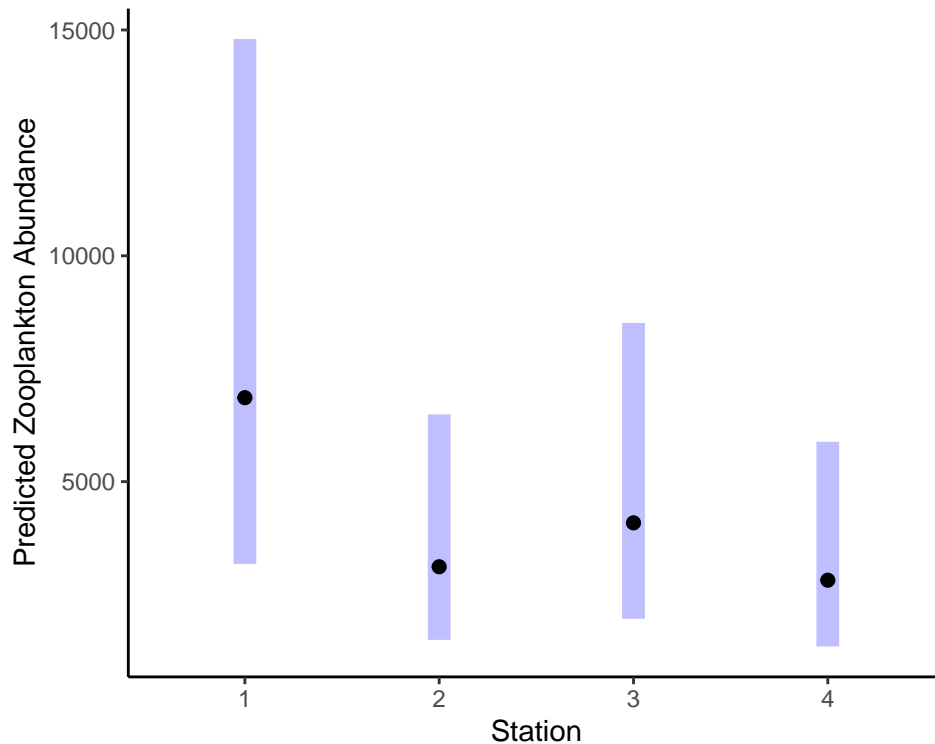
```
pairs(station_emms)
#>  contrast             ratio    SE    df null t.ratio p.value
#>  Station1 / Station2 2.202 0.638 42.7    1   2.724  0.0444
#>  Station1 / Station3 1.678 0.454 42.9    1   1.915  0.2370
#>  Station1 / Station4 2.435 0.741 43.2    1   2.923  0.0271
#>  Station2 / Station3 0.762 0.136 42.0    1  -1.517  0.4367
#>  Station2 / Station4 1.106 0.213 42.3    1   0.523  0.9531
#>  Station3 / Station4 1.451 0.254 42.2    1   2.125  0.1618
#>
#> Degrees-of-freedom method: kenward-roger
#> P value adjustment: tukey method for comparing a family of 4 estimates
#> Tests are performed on the log scale
```

Station 1 has higher zooplankton diversity than Stations 2 or 4.

Turbidity

```
drop_low %>%
  ggplot(aes(log(Turb), log1p(Density))) +
  geom_point(aes(color = Season), size = 2, alpha = .75) +
  geom_smooth(method = 'lm')
#> `geom_smooth()` using formula 'y ~ x'
```

Data
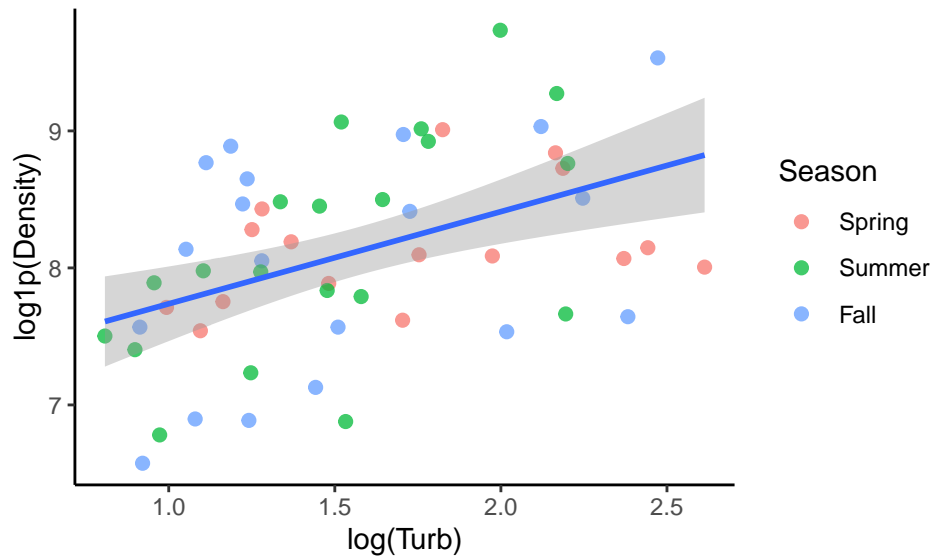
```
station_emms <- emmeans(density_lmer_no_low, ~Turb, at = list(Turb = 1:15),
                        type = 'response')
plot(station_emms) +
  coord_flip() +
  xlab("Estimated Zooplankton Density")
```

Model
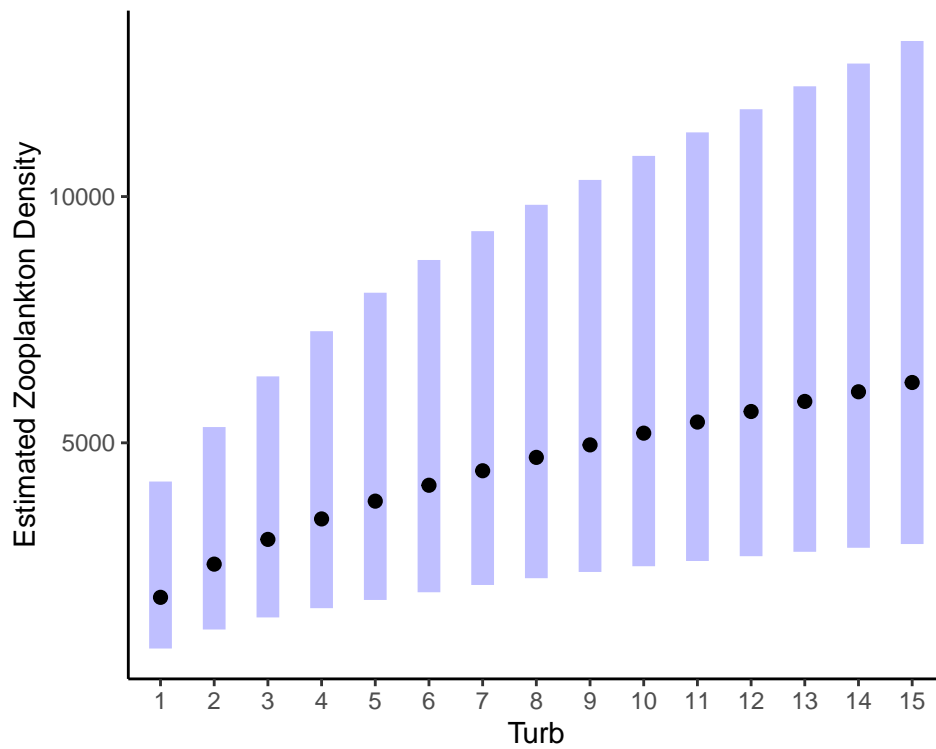
Chlorophyll

Data There is not much pattern with chlorophyll in the raw data. I show it here by Station, because it may differ – but we did not fit interaction terms.

```
drop_low %>%
  ggplot(aes(log(Chl), log1p(Density))) +
  geom_point(aes(color = Station), size = 2, alpha = .75) +
  geom_smooth(method = 'lm') +
  facet_wrap(~Station)
#> `geom_smooth()` using formula 'y ~ x'
```
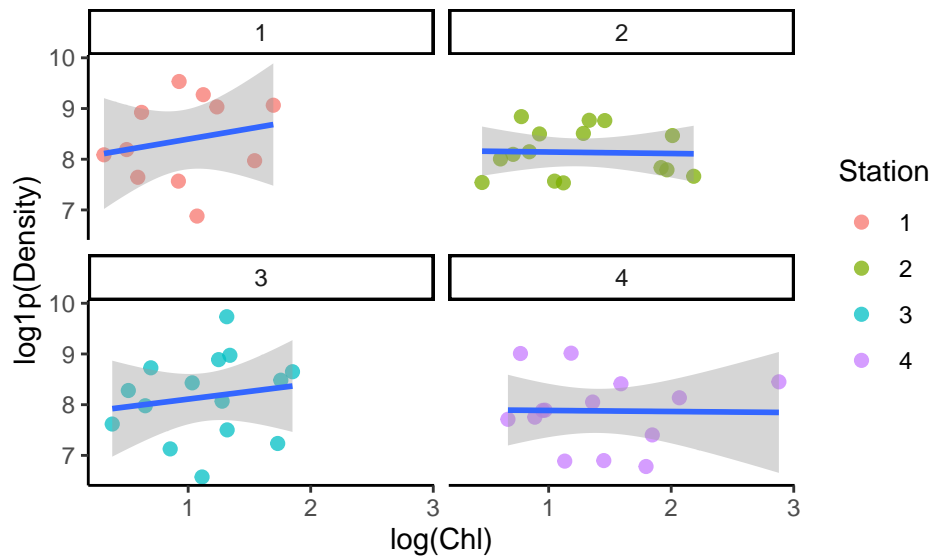

Model But the relationship emerges in the model, presumably because it is hidden or confounded by other predictors in the model.

```
station_emms <- emmeans(density_lmer_no_low, ~Chl, at = list(Chl = 1:20),
                        type = 'response')
plot(station_emms) +
  coord_flip() +
  xlab("Estimated Zooplankton Density")
```
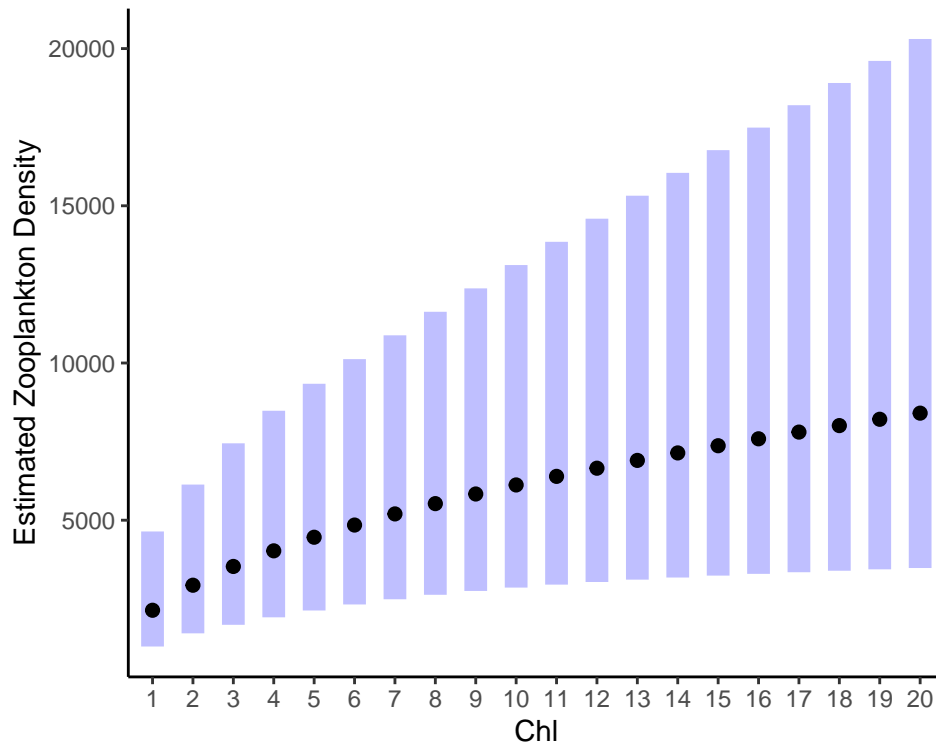
Shannon Diversity

```
shannon_lmer <- lmer(H ~ Station +
                         Temp +
                         Sal +
                         log(Turb) +
                         log(Chl) +
                         log1p(Fish) +
                         (1 | Yearf),
                      data = complete_data, na.action = na.omit)
anova(shannon_lmer)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>              Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
#> Station     0.98117 0.32706     3 46.909  1.2833 0.2911
#> Temp        0.14639 0.14639     1 46.517  0.5744 0.4523
#> Sal         0.44368 0.44368     1 48.553  1.7409 0.1932
#> log(Turb)   0.01654 0.01654     1 47.978  0.0649 0.8000
#> log(Chl)    0.54991 0.54991     1 43.450  2.1578 0.1491
#> log1p(Fish) 0.08348 0.08348     1 47.149  0.3276 0.5698
```

So this model returns no statistically significant predictors. Lets check on this with `step()`.

Stepwise Model Selection

```
(shannon_step <- step(shannon_lmer,))
#> Backward reduced random-effect table:
#>
#>            Eliminated npar  logLik    AIC     LRT Df Pr(>Chisq)
#> <none>                 11 -54.512 131.02
#> (1 | Yearf)        1   10 -55.225 130.45 1.4268  1     0.2323
#>
#> Backward reduced fixed-effect table:
#>            Eliminated Df Sum of Sq    RSS      AIC F value Pr(>F)
#> log(Turb)           1  1   0.00022 13.628 -68.002  0.0008 0.9775
#> log1p(Fish)         2  1   0.10592 13.734 -69.553  0.3886 0.5359
#> Temp                3  1   0.17767 13.912 -70.807  0.6598 0.4204
#> log(Chl)            4  1   0.54639 14.458 -70.573  2.0423 0.1590
#> Sal                 5  1   0.75702 15.215 -69.613  2.7751 0.1016
#> Station             6  3   1.90301 17.118 -68.778  2.2513 0.0928 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Model found:
#> H ~ 1
shannon_step <- get_model(shannon_step)
```

So, even after model simplification, we see no statistically significant predictors.

Model Diagnostics

```
plot(shannon_lmer, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
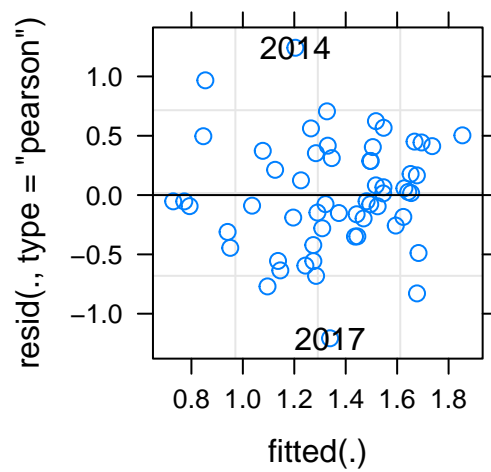
```
plot(shannon_lmer, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
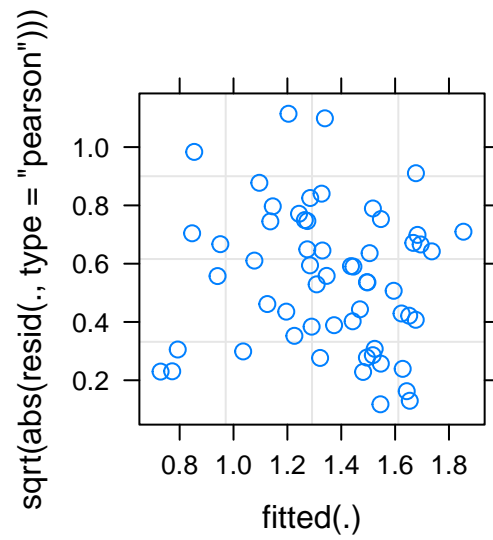


```
plot(shannon_lmer, hatvalues(.) ~ fitted(.))
```
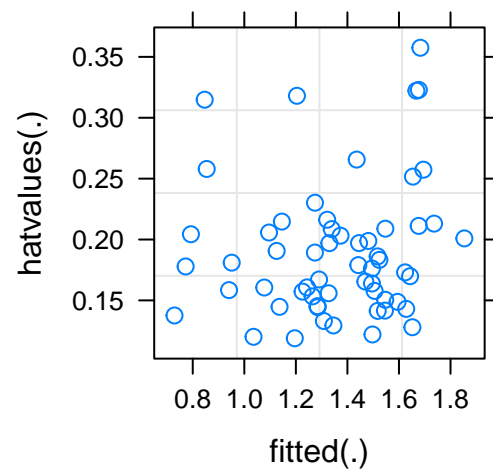


```
plot(shannon_lmer, cooks.distance(.) ~ fitted(.))
```
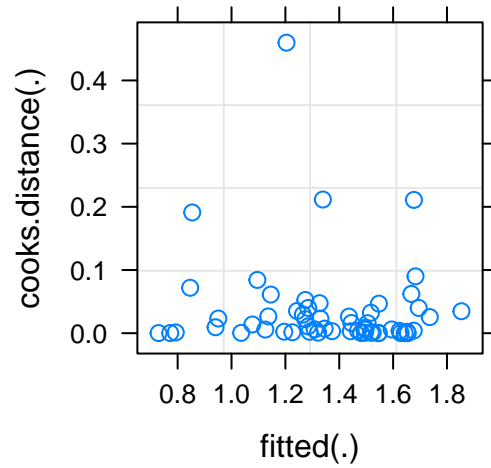
```
as_tibble(resid(shannon_lmer)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
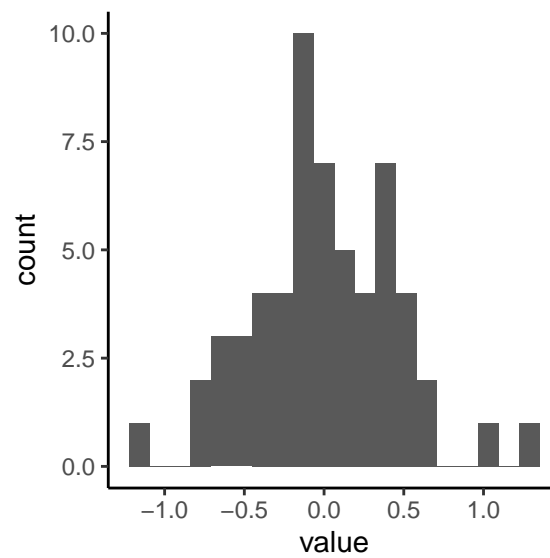

This model is not dreadful, although he one outlier may be problematic.

Model on Reduced Data

```
shannon_lmer_no_low <- lmer(H ~ Station +
                              Temp +
                              Sal +
                              log(Turb) +
                              log(Chl) +
                              log1p(Fish) +
                              (1 | Yearf),
```

```
                            data = drop_low, na.action = na.omit)
anova(shannon_lmer_no_low)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>              Sum Sq Mean Sq NumDF  DenDF F value  Pr(>F)
#> Station     1.23530 0.41177     3 43.823  2.1158 0.11190
#> Temp        0.00213 0.00213     1 44.920  0.0109 0.91716
#> Sal         0.06569 0.06569     1 44.083  0.3375 0.56422
#> log(Turb)   0.00293 0.00293     1 45.208  0.0150 0.90296
#> log(Chl)    0.17845 0.17845     1 41.209  0.9169 0.34387
#> log1p(Fish) 0.66696 0.66696     1 43.395  3.4271 0.07095 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fish abundance is marginally significant here.

Stepwise Model Selection

```
(shannon_step_no_low <- step(shannon_lmer_no_low, reduce.random = TRUE))
#> Backward reduced random-effect table:
#>
#>             Eliminated npar  logLik    AIC    LRT Df Pr(>Chisq)
#> <none>                   11 -46.041 114.08
#> (1 | Yearf)          1   10 -47.248 114.50 2.4142  1     0.1202
#>
#> Backward reduced fixed-effect table:
#>             Eliminated Df Sum of Sq    RSS     AIC F value   Pr(>F)
#> Temp                 1  1    0.0005 10.238 -76.467  0.0023 0.961971
#> log(Turb)            2  1    0.0133 10.251 -78.396  0.0610 0.806067
#> Sal                  3  1    0.0686 10.320 -80.029  0.3214 0.573393
#> log(Chl)             4  1    0.3910 10.711 -79.983  1.8565 0.179261
#> log1p(Fish)          5  1    0.6786 11.390 -78.605  3.1678 0.081182 .
#> Station              0  3    3.8875 15.277 -68.454  5.8025 0.001718 **
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Model found:
#> H ~ Station
shannon_step_no_low <- get_model(shannon_step_no_low)
```

Reducing model complexity using stepwise model selection by AIC leads to a model that retains Station as a fairly robust predictor, even though it was NOT identified as important in the full model. Station is partially collinear with Salinity and Chlorophyll, so it is possible that including too many terms in the initial model hid a simpler pattern.
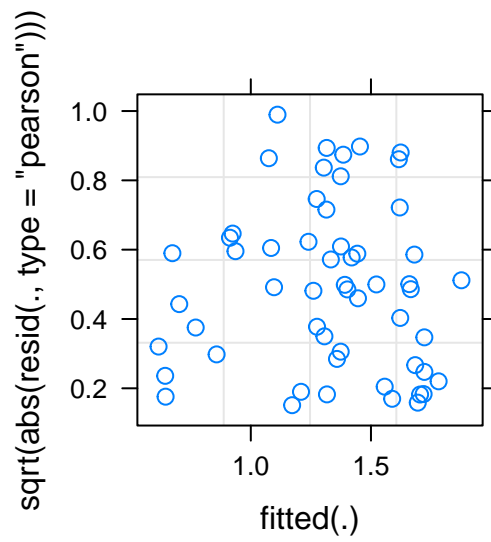
Model Diagnostics (full Model)

```
plot(shannon_lmer_no_low, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
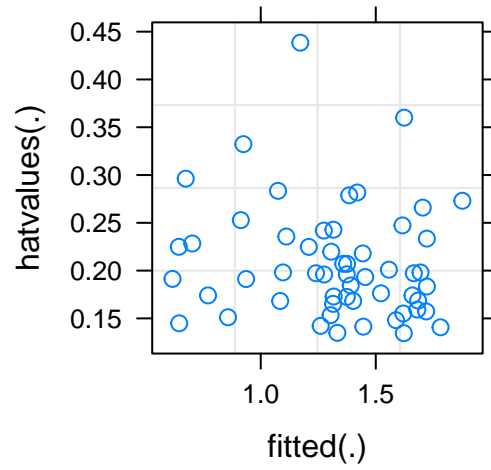```
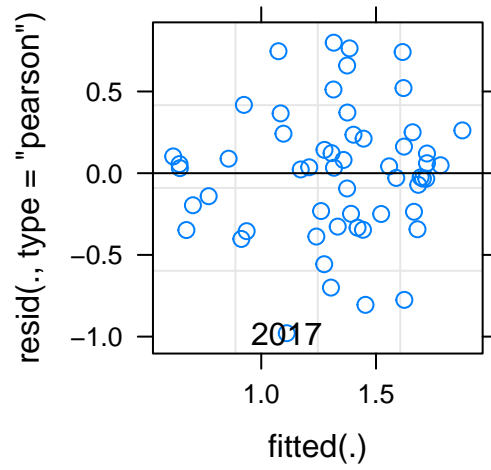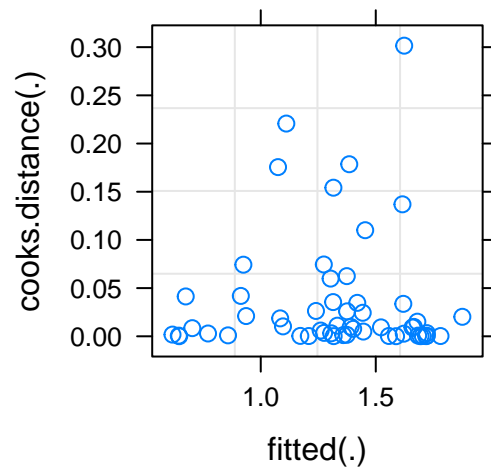
```
plot(shannon_lmer_no_low, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
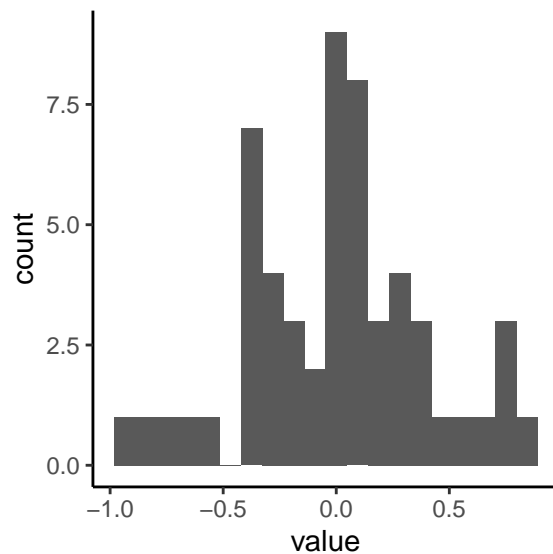


```
plot(shannon_lmer_no_low, hatvalues(.) ~ fitted(.))
```

```
plot(shannon_lmer_no_low, cooks.distance(.) ~ fitted(.))
```
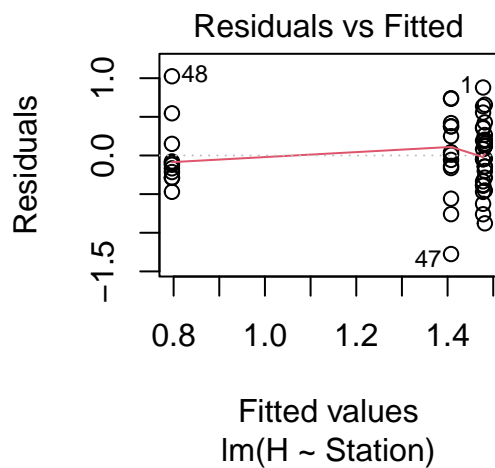


```
as_tibble(resid(shannon_lmer_no_low)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
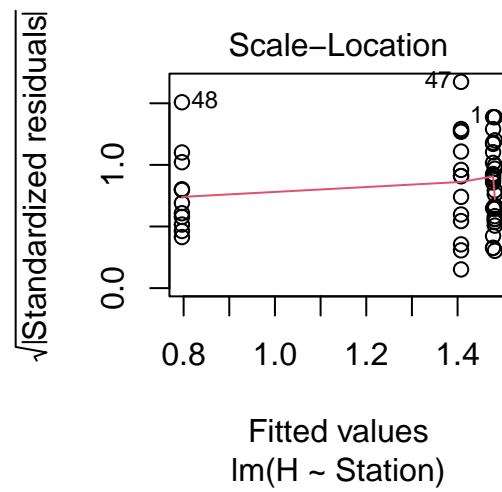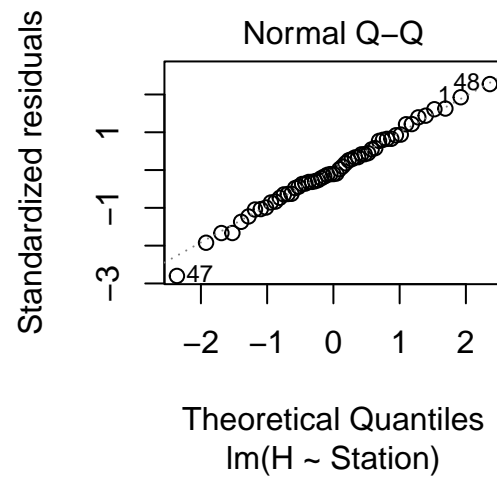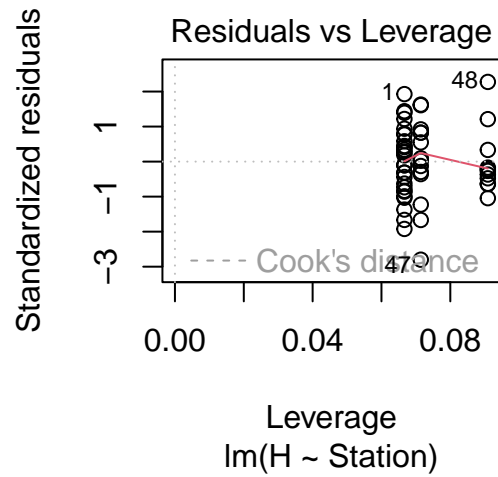
Model diagnostics are not too bad. While we have some points with slightly high leverage compared to other points, the leverage values are only moderate at 0.35 and Cook's distance is < 0.5.

Model Diagnostics (simple Model)

```
plot(shannon_step_no_low)
```

Normal Q–Q

Standardized residuals

Theoretical Quantiles
lm(H ~ Station)

Scale–Location

√|Standardized residuals|

Fitted values
lm(H ~ Station)

Residuals vs Leverage

lm(H ~ Station)

Graphic Summary

```
drop_low %>%
  ggplot(aes(Station, H)) +
  stat_summary(size = 1, color = 'grey50') +
  geom_point(aes(color = Season), size = 2, alpha = .75)
#> No summary function supplied, defaulting to `mean_se()`
```

Data (dropping low salinity samples)
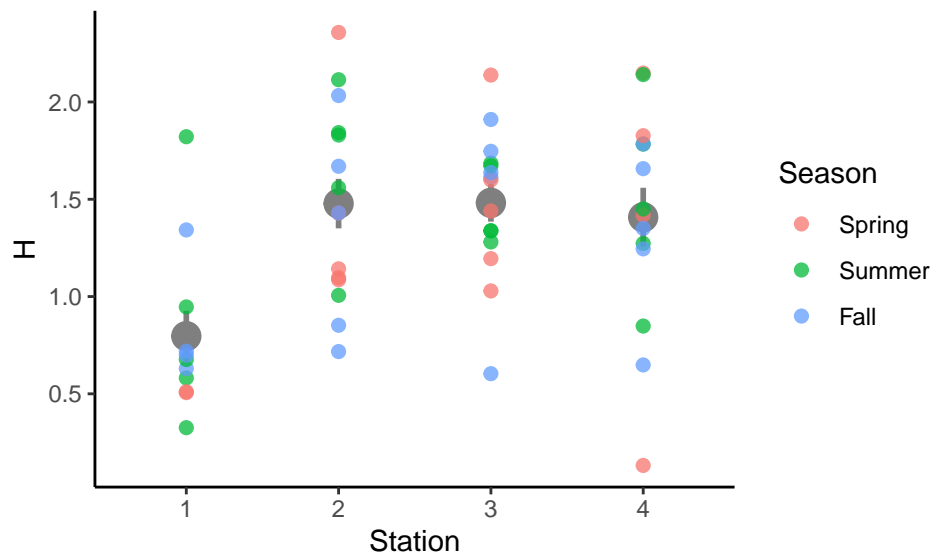
Marginal (Modeled) Means – Full Model   This shows comparisons of "marginal", "least squares" or
"adjusted" means from the full model, where Station was not identified as a significant predictor of diversity.

```
station_emms <- emmeans(shannon_lmer_no_low, ~Station, type = 'response')
plot(station_emms) +
  coord_flip() +
  xlab('Predicted Zooplankton Diversity')
```
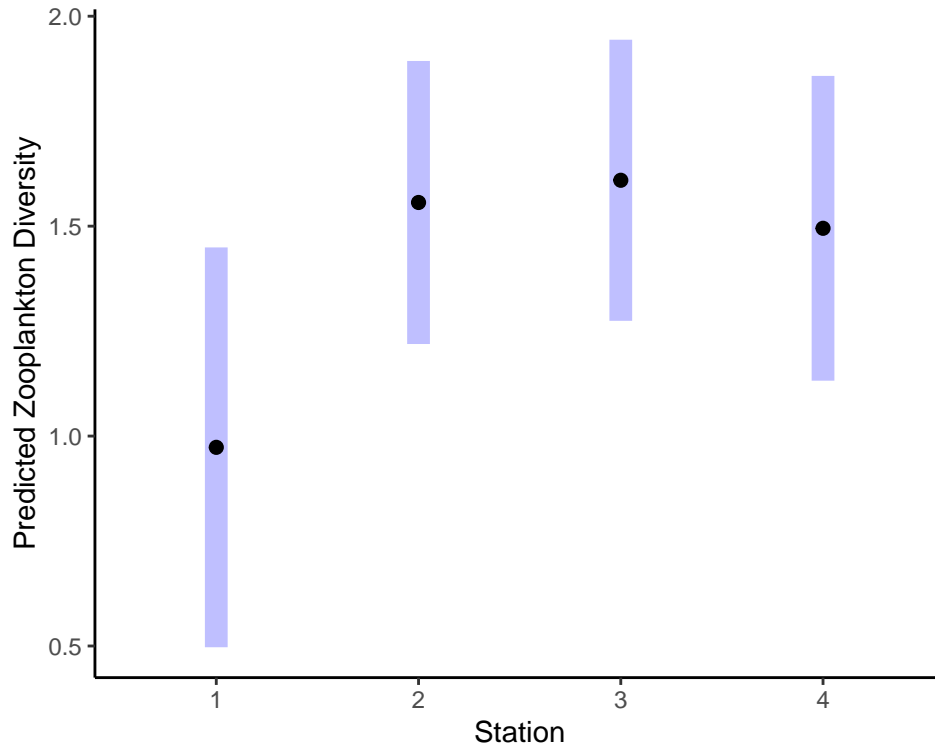


```
pairs(station_emms)
#>  contrast              estimate     SE    df t.ratio p.value
#>  Station1 - Station2   -0.5832 0.285 45.3  -2.049  0.1856
#>  Station1 - Station3   -0.6361 0.265 45.6  -2.398  0.0917
#>  Station1 - Station4   -0.5217 0.298 46.0  -1.751  0.3100
#>  Station2 - Station3   -0.0529 0.176 42.2  -0.300  0.9905
#>  Station2 - Station4    0.0615 0.189 43.3   0.325  0.9880
#>  Station3 - Station4    0.1144 0.172 43.3   0.664  0.9100
#>
#> Degrees-of-freedom method: kenward-roger
#> P value adjustment: tukey method for comparing a family of 4 estimates
```

Plankton diversity is lowest upstream, although the pattern is weak, and does not clearly emerge from the noise in this model.

Marginal (Modeled) Means – Reduced Model

```
station_emms <- emmeans(shannon_step_no_low, ~Station, type = 'response')
plot(station_emms) +
  coord_flip() +
  xlab('Predicted Zooplankton Diversity')
```
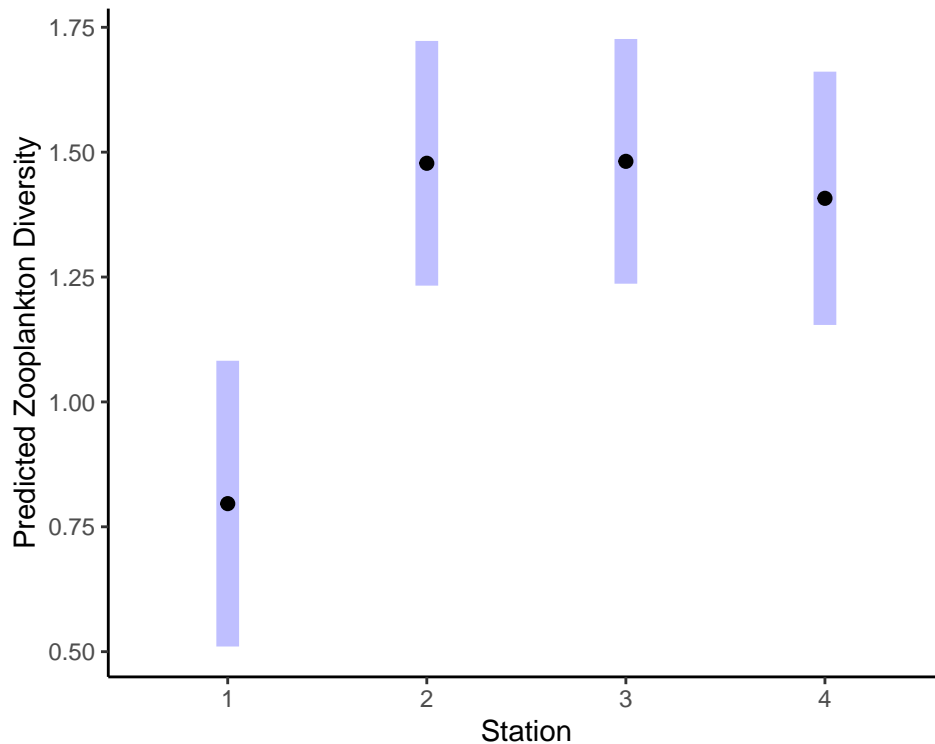
```
pairs(station_emms)
#>  contrast            estimate    SE df t.ratio p.value
#>  Station1 - Station2 -0.68138 0.188 51  -3.632  0.0035
#>  Station1 - Station3 -0.68526 0.188 51  -3.653  0.0033
#>  Station1 - Station4 -0.61126 0.190 51  -3.210  0.0119
#>  Station2 - Station3 -0.00388 0.173 51  -0.022  1.0000
#>  Station2 - Station4  0.07013 0.176 51   0.399  0.9782
#>  Station3 - Station4  0.07401 0.176 51   0.421  0.9746
#>
#> P value adjustment: tukey method for comparing a family of 4 estimates
```

The simpler model has narrower error bars, but shows the same pattern as the data or the full model. Station 1 shows lower zooplankton diversity than the other three sites.

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 $(\log(0) = -\text{Inf}; 1/0 = \text{Inf}, 1 / 0{*}0 = \text{Inf})$. The easiest solution is to add some finite small quantity to the density data, and predict that. Here we predict $\log(\text{Density} + 1)$ using Gaussian models. It's not ideal, especially for species where there are many zeros, but it's a good start. A two stage model would be a bit more robust, but probably no more informative.

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.

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

Next, I create a function to run the `lmer()` analysis. The function takes a data frame or tibble as an argument. The tibble must have data columns with the correct names. Unfortunately, I can't run `step()` on the full model within a nested tibble, as step() is unable to find the data used to construct each model used as a starting point for the stepwise model simplification.

```
my_lmer <- function(.dat) {
  mod_1 <-  lmer(log1p(Density) ~
                   Station +
                   Temp +
                   Sal +
                   log(Turb) +
                   log(Chl) +
                   log1p(Fish) +
                   (1 | Yearf),
  data = .dat, na.action = na.omit)
  return(mod_1)
}
```

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

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

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_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>              Sum Sq Mean Sq NumDF  DenDF F value     Pr(>F)
#> Station      3.7819  1.2606     3 45.593  1.1664 0.3329047
#> Temp        17.0592 17.0592     1 47.665 15.7844 0.0002391 ***
#> Sal         19.0879 19.0879     1 45.973 17.6616 0.0001203 ***
#> log(Turb)    6.1834  6.1834     1 45.802  5.7214 0.0209217 *
#> log(Chl)     2.0405  2.0405     1 48.086  1.8881 0.1757921
#> log1p(Fish)  2.2684  2.2684     1 45.444  2.0989 0.1542670
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 185.9
#>
#> Scaled residuals:
#>     Min      1Q  Median      3Q     Max
#> -3.1843 -0.4444 -0.0919  0.6702  1.6364
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.9221   0.9603
#>  Residual             1.0808   1.0396
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>              Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept) -0.72290    1.26458 46.48062  -0.572 0.570309
#> Station2    -0.95848    0.66343 46.13957  -1.445 0.155288
#> Station3    -0.26349    0.60764 46.24915  -0.434 0.666573
#> Station4    -0.47670    0.67849 46.70685  -0.703 0.485802
#> Temp         0.22724    0.05720 47.66492   3.973 0.000239 ***
#> Sal          0.14314    0.03406 45.97343   4.203 0.000120 ***
#> log(Turb)    0.84003    0.35119 45.80201   2.392 0.020922 *
#> log(Chl)     0.50556    0.36793 48.08555   1.374 0.175792
#> log1p(Fish) -0.11726    0.08094 45.44414  -1.449 0.154267
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>             (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2    -0.235
#> Station3    -0.386  0.795
```

```
#> Station4     -0.425   0.789   0.808
#> Temp         -0.567   0.631   0.619   0.663
#> Sal          -0.161  -0.684  -0.600  -0.568  -0.355
#> log(Turb)    -0.674  -0.036   0.170   0.246   0.219   0.198
#> log(Chl)      0.197  -0.264  -0.237  -0.350  -0.515  -0.123  -0.122
#> log1p(Fish)  -0.289   0.196   0.217   0.214   0.114  -0.162  -0.078  -0.001
```
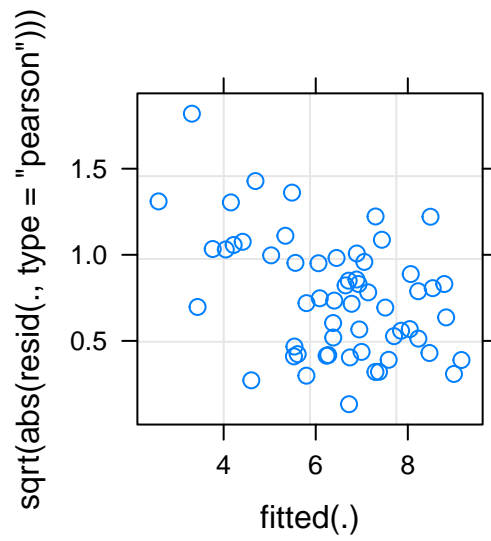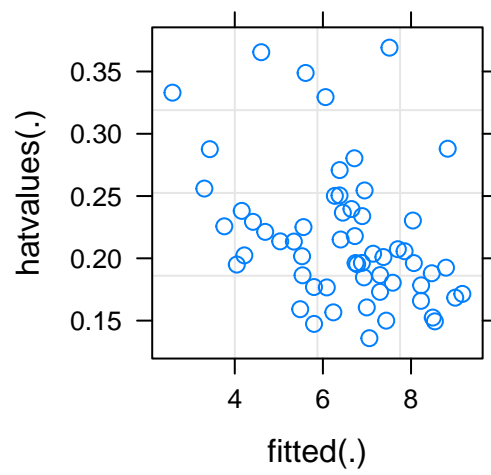
Acartia is nmor abundant

Model Diagnostics
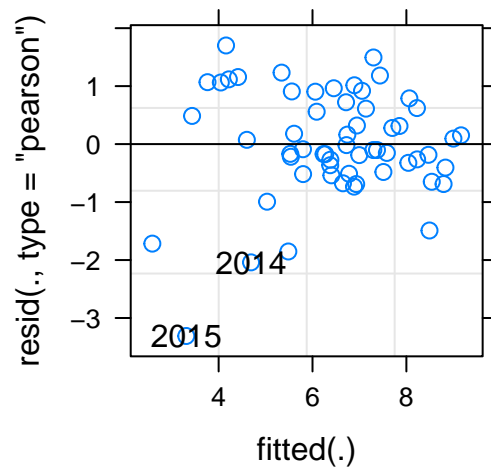
```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
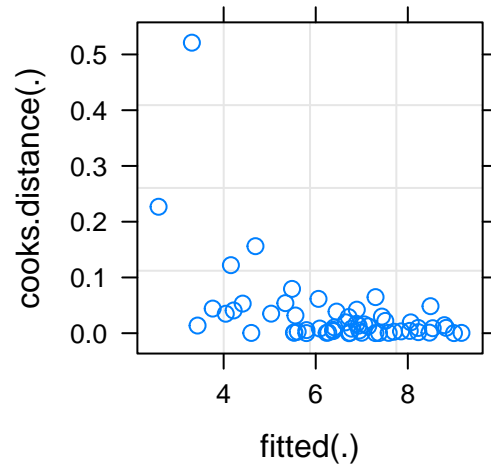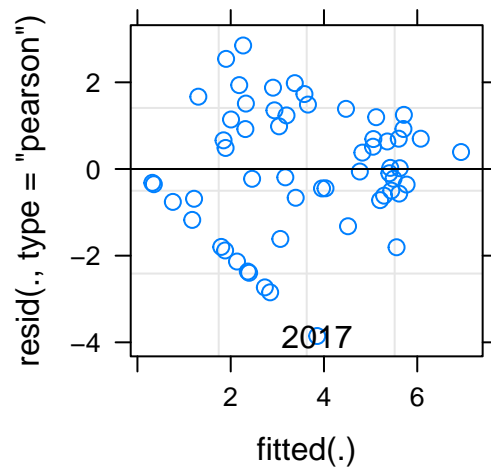


```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```

```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```

```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
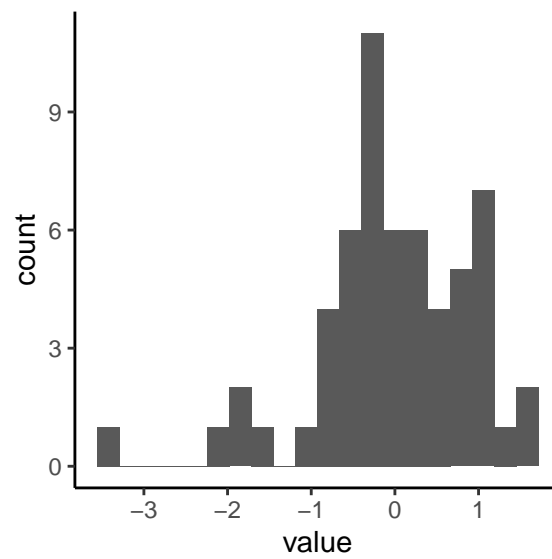
Balanus

```
spp = 'Balanus'
mod_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>            Sum Sq Mean Sq NumDF  DenDF F value     Pr(>F)
#> Station     6.307   2.102     3 45.969  0.8186 0.4902442
#> Temp        6.108   6.108     1 48.565  2.3787 0.1294944
#> Sal         0.381   0.381     1 46.566  0.1482 0.7019890
#> log(Turb)   0.799   0.799     1 46.299  0.3110 0.5797263
#> log(Chl)   42.609  42.609     1 48.881 16.5924 0.0001693 ***
#> log1p(Fish) 7.885   7.885     1 45.798  3.0704 0.0864242 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 226.5
#>
#> Scaled residuals:
#>      Min       1Q   Median       3Q      Max
#> -2.40507 -0.42519 -0.01249  0.68806  1.77812
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 1.303    1.142
#>  Residual             2.568    1.602
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>             Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept)  1.97717    1.89426 48.37954   1.044 0.301780
#> Station2     0.46413    1.01881 46.77245   0.456 0.650815
#> Station3     1.18283    0.93276 46.92301   1.268 0.211017
#> Station4     0.93524    1.03973 47.52116   0.900 0.372920
#> Temp        -0.13462    0.08729 48.56469  -1.542 0.129494
#> Sal         -0.02015    0.05234 46.56625  -0.385 0.701989
#> log(Turb)   -0.30114    0.53995 46.29948  -0.558 0.579726
#> log(Chl)     2.28189    0.56020 48.88112   4.073 0.000169 ***
#> log1p(Fish)  0.21831    0.12459 45.79755   1.752 0.086424 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.236
#> Station3   -0.391  0.793
```
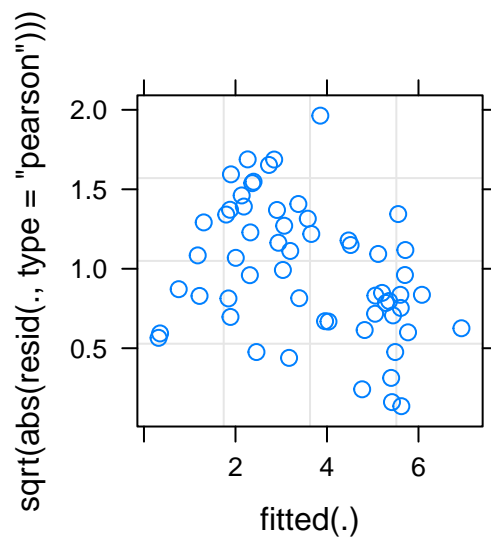
```
#> Station4     -0.430  0.787  0.807
#> Temp         -0.576  0.629  0.617  0.659
#> Sal          -0.167 -0.685 -0.600 -0.569 -0.359
#> log(Turb)    -0.689 -0.041  0.166  0.241  0.213  0.201
#> log(Chl)      0.191 -0.259 -0.231 -0.344 -0.506 -0.122 -0.116
#> log1p(Fish)  -0.295  0.193  0.214  0.212  0.112 -0.159 -0.080  0.001
```

Model Diagnostics

```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```



```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```

```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```
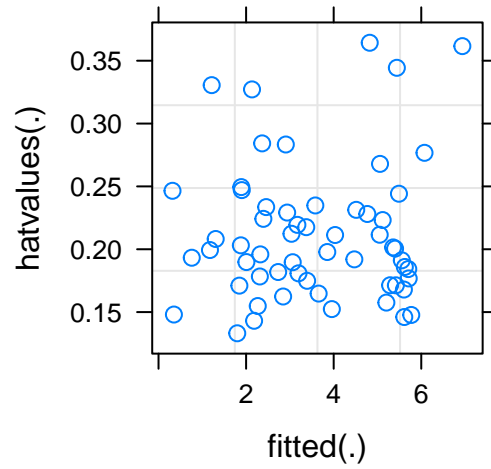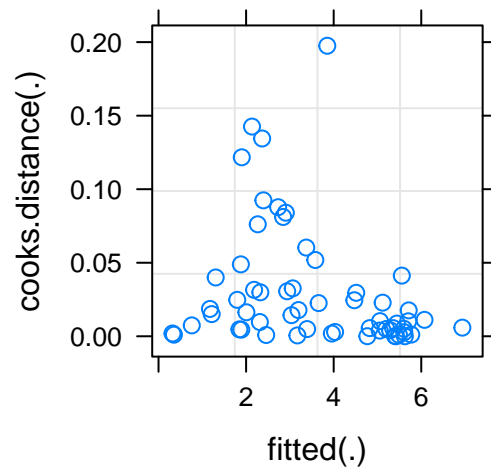


```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
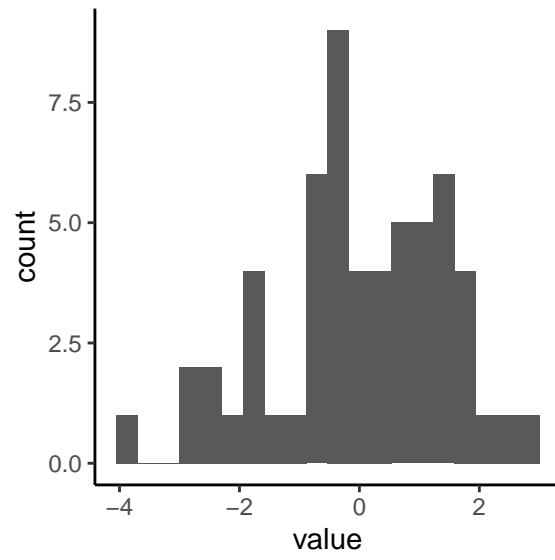
Eurytemora

```
spp = 'Eurytemora'
mod_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>             Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
#> Station     6.1690  2.0563     3 46.126  1.9088 0.141337
#> Temp        0.1218  0.1218     1 42.417  0.1131 0.738349
#> Sal        11.9765 11.9765     1 48.811 11.1173 0.001639 **
#> log(Turb)   3.8564  3.8564     1 48.042  3.5797 0.064525 .
#> log(Chl)    0.2023  0.2023     1 36.716  0.1878 0.667302
#> log1p(Fish) 3.1639  3.1639     1 46.886  2.9369 0.093180 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 178.8
#>
#> Scaled residuals:
#>     Min      1Q  Median      3Q     Max
#> -3.1848 -0.4579 -0.0160  0.4427  2.5973
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.07261  0.2695
#>  Residual             1.07729  1.0379
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>             Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept)  3.44228    1.14938 48.29533   2.995  0.00432 **
#> Station2    -0.64544    0.64314 48.86429  -1.004  0.32053
#> Station3    -1.00675    0.58716 48.97828  -1.715  0.09274 .
#> Station4    -1.36153    0.64677 48.42791  -2.105  0.04049 *
#> Temp        -0.01776    0.05283 42.41684  -0.336  0.73835
#> Sal          0.11040    0.03311 48.81081   3.334  0.00164 **
#> log(Turb)    0.65017    0.34364 48.04171   1.892  0.06453 .
#> log(Chl)    -0.14453    0.33353 36.71618  -0.433  0.66730
#> log1p(Fish)  0.13688    0.07987 46.88559   1.714  0.09318 .
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.207
#> Station3   -0.376  0.782
```
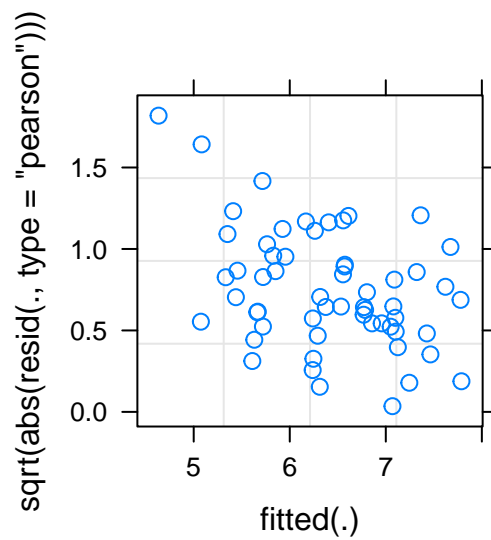
```
#> Station4     -0.411  0.775  0.794
#> Temp         -0.560  0.614  0.599  0.635
#> Sal          -0.189 -0.690 -0.600 -0.573 -0.384
#> log(Turb)    -0.706 -0.076  0.137  0.210  0.173  0.222
#> log(Chl)      0.119 -0.226 -0.193 -0.303 -0.435 -0.114 -0.071
#> log1p(Fish)  -0.300  0.175  0.197  0.193  0.095 -0.144 -0.090  0.015
```

Model Diagnostics
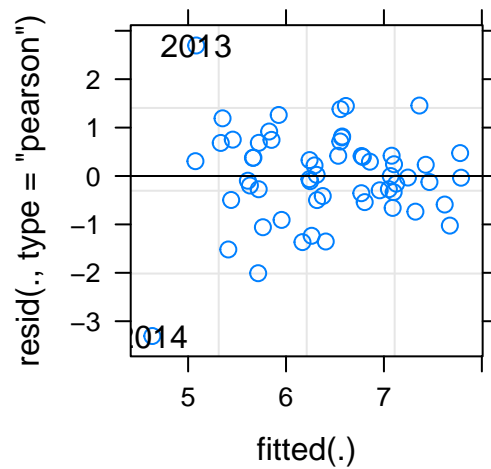
```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```



```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```

```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```
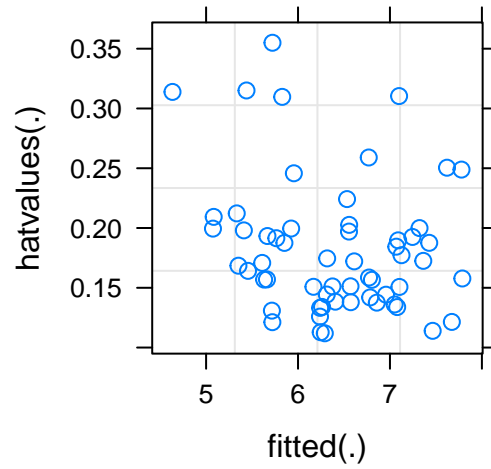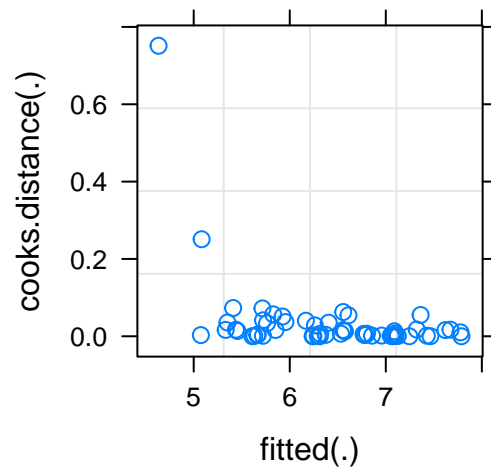


```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
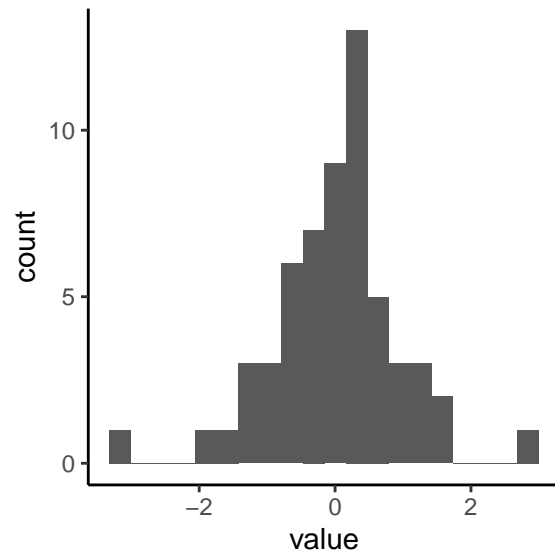
Polychaete

```
spp = "Polychaete"

mod_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>             Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
#> Station     12.593   4.198     3 46.274  1.1746  0.329650
#> Temp        31.714  31.714     1 49.000  8.8745  0.004489 **
#> Sal          7.116   7.116     1 47.205  1.9913  0.164763
#> log(Turb)   24.647  24.647     1 46.802  6.8969  0.011632 *
#> log(Chl)     7.019   7.019     1 48.767  1.9640  0.167416
#> log1p(Fish)  2.864   2.864     1 46.128  0.8015  0.375291
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 241
#>
#> Scaled residuals:
#>     Min      1Q  Median      3Q     Max
#> -1.9678 -0.5202  0.0667  0.6710  1.6547
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 1.075    1.037
#>  Residual             3.574    1.890
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>             Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept)  3.35828    2.18669 48.80632   1.536  0.13105
#> Station2     1.35096    1.19557 47.42947   1.130  0.26417
#> Station3     1.92047    1.09398 47.62141   1.755  0.08561 .
#> Station4     1.93109    1.21651 48.31034   1.587  0.11895
#> Temp        -0.30249    0.10154 48.99999  -2.979  0.00449 **
#> Sal         -0.08672    0.06146 47.20497  -1.411  0.16476
#> log(Turb)    1.66677    0.63467 46.80158   2.626  0.01163 *
#> log(Chl)     0.91035    0.64958 48.76744   1.401  0.16742
#> log1p(Fish)  0.13132    0.14668 46.12755   0.895  0.37529
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>             (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2    -0.232
```
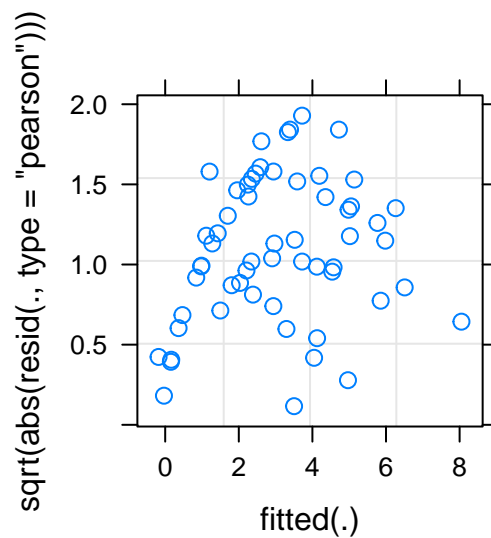
```
#> Station3    -0.391  0.791
#> Station4    -0.430  0.785  0.804
#> Temp        -0.577  0.626  0.613  0.655
#> Sal         -0.173 -0.686 -0.600 -0.570 -0.364
#> log(Turb)   -0.698 -0.048  0.160  0.235  0.205  0.205
#> log(Chl)     0.179 -0.252 -0.223 -0.335 -0.492 -0.121 -0.106
#> log1p(Fish) -0.298  0.189  0.211  0.208  0.109 -0.157 -0.082  0.004
```

Model Diagnostics
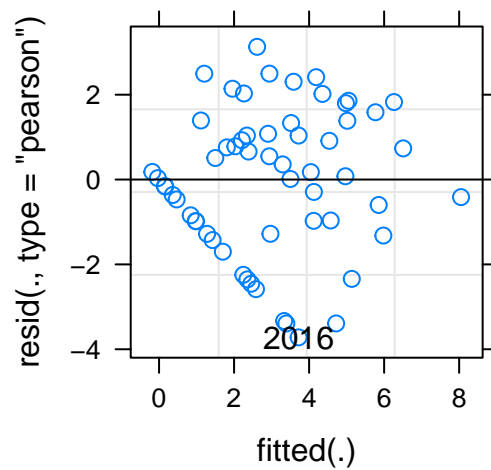
```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```



```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
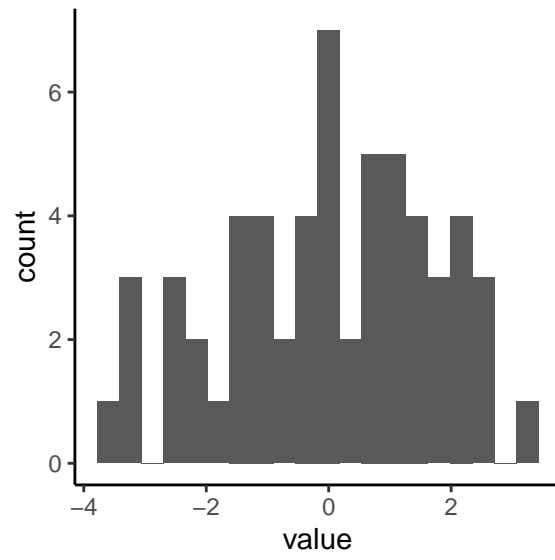
```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```
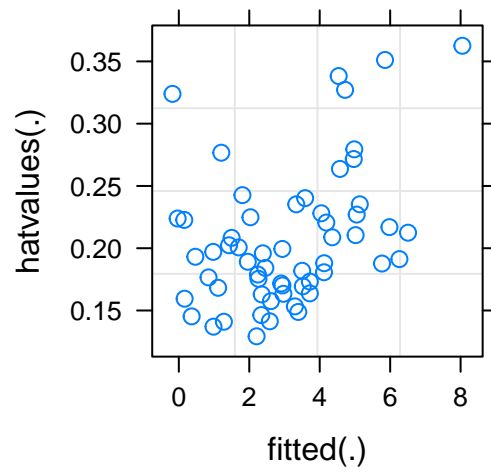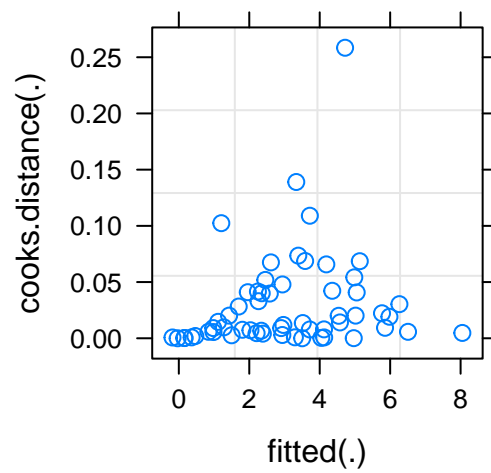


```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
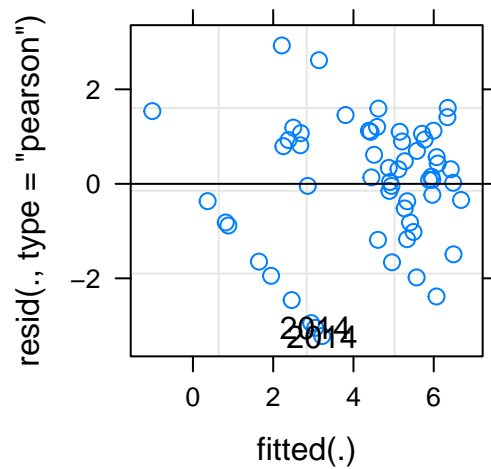
Pseudocal

```
spp = "Pseudocal"
mod_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>              Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
#> Station      49.665 16.5551     3 45.759  7.2227 0.0004559 ***
#> Temp          0.500  0.5000     1 48.858  0.2181 0.6425431
#> Sal           0.032  0.0316     1 46.597  0.0138 0.9070676
#> log(Turb)     4.507  4.5069     1 46.226  1.9663 0.1675222
#> log(Chl)      0.158  0.1585     1 48.998  0.0691 0.7936971
#> log1p(Fish)  11.257 11.2570     1 45.570  4.9112 0.0317173 *
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 220.1
#>
#> Scaled residuals:
#>      Min       1Q   Median       3Q      Max
#> -2.12734 -0.54041  0.09501  0.67954  1.93374
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.8975   0.9474
#>  Residual             2.2921   1.5140
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>              Estimate Std. Error        df t value Pr(>|t|)
#> (Intercept)  0.983512   1.769326 48.635236   0.556 0.580849
#> Station2     3.522204   0.960207 46.844146   3.668 0.000623 ***
#> Station3     3.928660   0.878881 47.038029   4.470 4.91e-05 ***
#> Station4     2.894577   0.978596 47.779159   2.958 0.004804 **
#> Temp         0.038268   0.081935 48.858252   0.467 0.642543
#> Sal          0.005791   0.049342 46.596830   0.117 0.907068
#> log(Turb)   -0.714130   0.509277 46.226243  -1.402 0.167522
#> log(Chl)     0.138066   0.525081 48.998366   0.263 0.793697
#> log1p(Fish)  0.260615   0.117599 45.570194   2.216 0.031717 *
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.234
#> Station3   -0.392  0.792
```
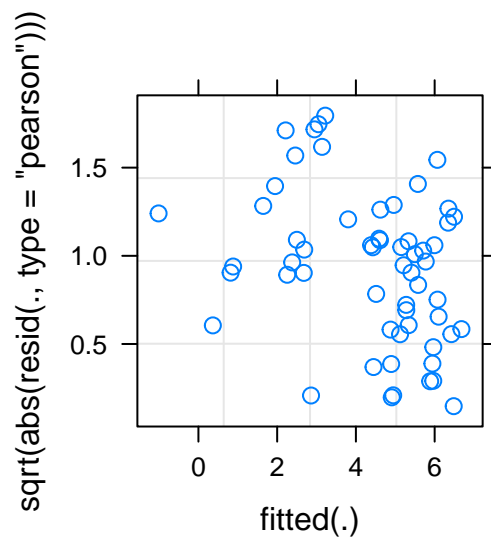
```
#> Station4     -0.431  0.786  0.805
#> Temp         -0.577  0.627  0.615  0.657
#> Sal          -0.170 -0.686 -0.600 -0.570 -0.362
#> log(Turb)    -0.694 -0.045  0.163  0.238  0.209  0.202
#> log(Chl)      0.186 -0.256 -0.228 -0.340 -0.499 -0.122 -0.111
#> log1p(Fish)  -0.297  0.191  0.213  0.210  0.110 -0.158 -0.081  0.003
```

Model Diagnostics

```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
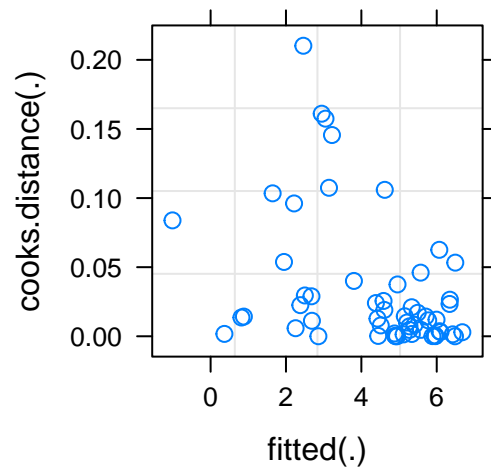


```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```
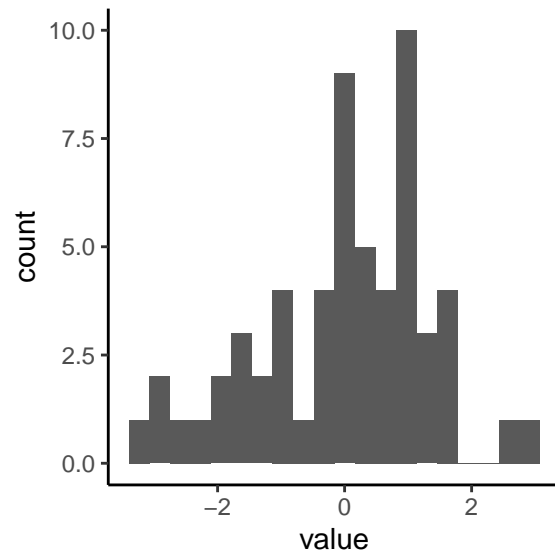
```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```
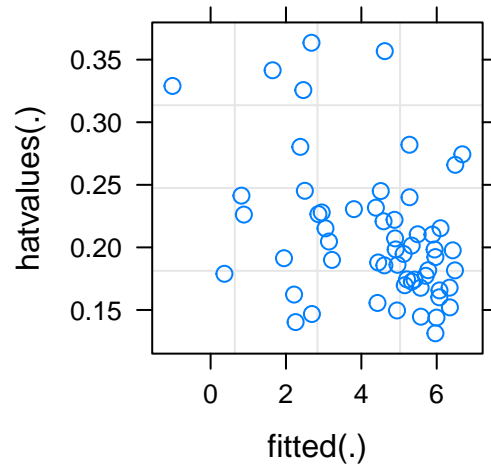


```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```
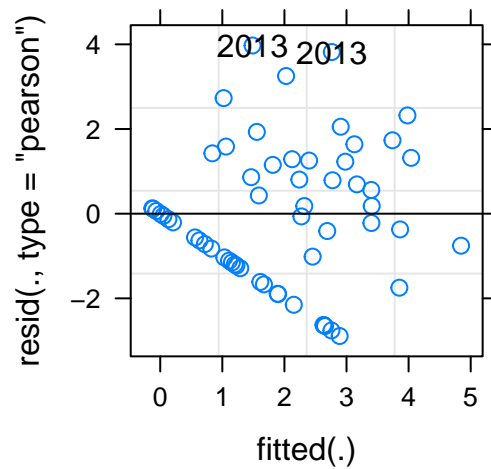
Temora

```
spp = "Temora"
mod_1 <- spp_analysis$lmers[spp_analysis$Species == spp][[1]]
anova(mod_1)
#> Type III Analysis of Variance Table with Satterthwaite's method
#>             Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
#> Station     29.1493  9.7164     3 46.846  3.0139 0.039245 *
#> Temp         0.4965  0.4965     1 45.253  0.1540 0.696570
#> Sal          1.2963  1.2963     1 48.719  0.4021 0.528985
#> log(Turb)    1.6617  1.6617     1 48.120  0.5154 0.476268
#> log(Chl)    30.3438 30.3438     1 41.307  9.4121 0.003796 **
#> log1p(Fish)  0.2030  0.2030     1 47.248  0.0630 0.802935
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_1)
#> Linear mixed model fit by REML. t-tests use Satterthwaite's method [
#> lmerModLmerTest]
#> Formula: log1p(Density) ~ Station + Temp + Sal + log(Turb) + log(Chl) +
#>     log1p(Fish) + (1 | Yearf)
#>    Data: .dat
#>
#> REML criterion at convergence: 232.9
#>
#> Scaled residuals:
#>      Min       1Q   Median       3Q      Max
#> -1.60924 -0.63652 -0.03224  0.67365  2.21248
#>
#> Random effects:
#>  Groups   Name        Variance Std.Dev.
#>  Yearf    (Intercept) 0.2738   0.5233
#>  Residual             3.2239   1.7955
#> Number of obs: 58, groups:  Yearf, 5
#>
#> Fixed effects:
#>             Estimate Std. Error       df t value Pr(>|t|)
#> (Intercept) -0.72137    1.99906 48.60459  -0.361   0.7198
#> Station2     0.52198    1.11617 48.79028   0.468   0.6421
#> Station3     2.25331    1.01939 48.91819   2.210   0.0318 *
#> Station4     1.32368    1.12457 48.81869   1.177   0.2449
#> Temp        -0.03617    0.09216 45.25256  -0.392   0.6966
#> Sal         -0.03643    0.05746 48.71894  -0.634   0.5290
#> log(Turb)    0.42773    0.59578 48.12030   0.718   0.4763
#> log(Chl)     1.78893    0.58311 41.30741   3.068   0.0038 **
#> log1p(Fish)  0.03472    0.13836 47.24800   0.251   0.8029
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Correlation of Fixed Effects:
#>            (Intr) Statn2 Statn3 Statn4 Temp   Sal    lg(Tr) lg(Ch)
#> Station2   -0.211
#> Station3   -0.379  0.783
```
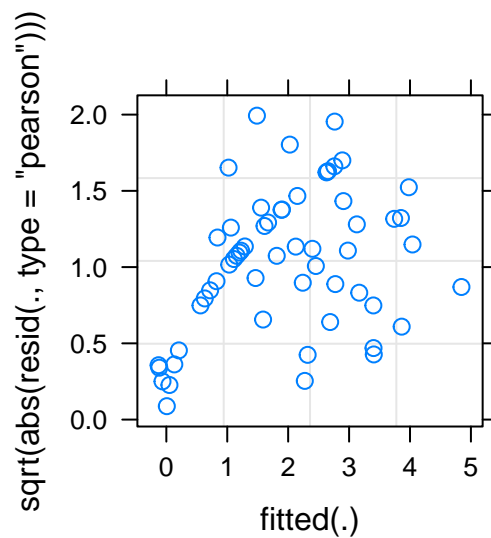
```
#> Station4    -0.415  0.776  0.796
#> Temp        -0.563  0.616  0.601  0.639
#> Sal         -0.187 -0.690 -0.601 -0.573 -0.381
#> log(Turb)   -0.706 -0.072  0.140  0.214  0.178  0.219
#> log(Chl)     0.129 -0.230 -0.197 -0.308 -0.445 -0.115 -0.077
#> log1p(Fish) -0.300  0.177  0.199  0.196  0.098 -0.146 -0.089  0.013
```
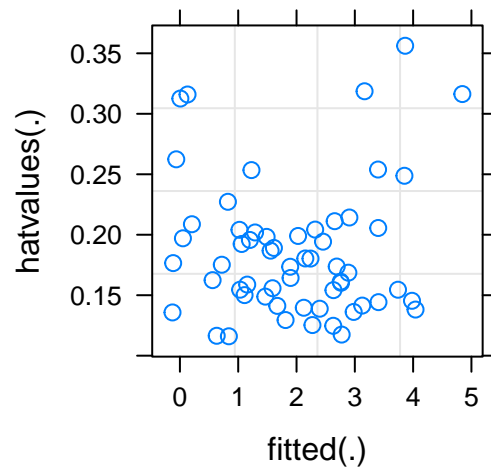
Model Diagnostics

```
plot(mod_1, resid(., type = "pearson") ~ fitted(.), abline = 0,
     id = 0.05)
```
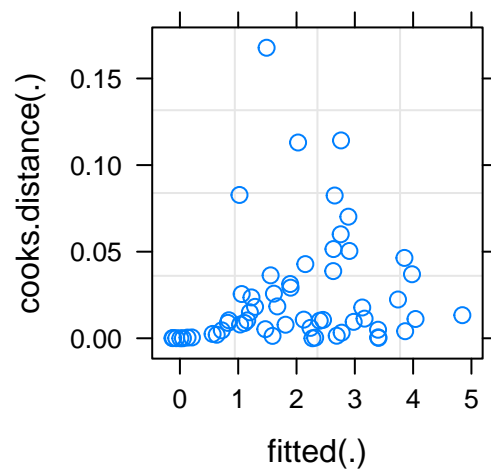


```
plot(mod_1, sqrt(abs(resid(., type = "pearson"))) ~ fitted(.))
```

```
plot(mod_1, hatvalues(.) ~ fitted(.))
```



```
plot(mod_1, cooks.distance(.) ~ fitted(.))
```



```
as_tibble(resid(mod_1)) %>% ggplot(aes(value)) + geom_histogram(bins = 20)
```