Black/Asian carp model selection

Eddie Wu

2024-04-04

Introduction

This .Rmd file is to show the progress on black carp and asian carp temperature and condition analyses. Since sub-sampling from spatial autocorrelation does not give significantly different results from normal analysis for black carp, we present the results without sub-sampling here. For other Asian carp species, we still sub-sample at a distance of 250 km.

SECTION 1 - Black Carp:

1. We first check how condition affects the relationship between black carp age at maturity and temperature by looking at three candidate models, for all six temperatures (annual, cold, warm, gdd0, water annual, water cold):

- Simple linear model (same slope, same intercept)
- Linear additive model (same slope, different intercept)
- Interaction model (different slope, different intercept)

2. Since we found that none of the interaction terms is significant, we will use only the simple linear model for prediction.

3. Since there is no significant relationship between warm temperature and age at maturity, we remove in from the following analyses.

SECTION 2 - Asian carp:

1. We conducted the same analyses on other asian carp species.

```
library(ggplot2)
library(ggfortify)
library(dplyr)
library(knitr)
library(tidyverse)
library(AICcmodavg) # for AICc and akaike weights
library(pwr)

## Import data
asian.carp <- read.csv("asian carp final.csv")
asian.carp$Condition <- as.factor(asian.carp$Condition)

Black <- read.csv("eddie_carp_new.csv")
Black$condition <- as.factor(Black$condition)

## Separate by species
Grass <- asian.carp[asian.carp$Species=="Grass",]
Bighead <- asian.carp[asian.carp$Species=="Bighead",]
Silver <- asian.carp[asian.carp$Species=="Silver",]
```

```
Big.sil <- rbind(Bighead, Silver) # combine the two groups


## Define two functions for AICs
compute_akaike_weights <- function(aic_scores) {
  # Find the AIC of the best model
  aic_min <- min(aic_scores)

  # Calculate delta AIC values
  d_aic <- aic_scores - aic_min

  # Compute Akaike weights
  akaike_weights <- exp(-0.5 * d_aic) / sum(exp(-0.5 * d_aic))

  return(akaike_weights)
}
compare_aic_scores <- function(aic_scores) {
  # Find the AIC of the best model
  aic_min <- min(aic_scores)

  # Determining if the smallest value is 2 units smaller than the others
  is_smaller_by_two <- all(aic_min + 2 <= aic_scores[aic_scores != aic_min])

  # Return the index if
  if (is_smaller_by_two) {
    min_index <- which(aic_scores == aic_min)
    return(min_index)
  } else {
    return(-999)
  }
}
```

SECTION 1: Black carp

For black carp data, we do not subsample at any distances. But we removed the South Ukraine data point for all the following analyses.

Data cleaning

```
# Clean data
Black <- Black %>% filter(!row_number() == 5) %>% filter(sex != "male")

# Remove the South Ukraine data point
black.clean <- Black %>% filter(!row_number() == 20)
```

Check slopes for three models (all six temperatures)

```
# Build the models
black.simple <- lm(log(AAM)~AnnualTemp, data = black.clean)
black.int <- lm(log(AAM)~AnnualTemp*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Annual Temperature

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42489 -0.12464  0.00059  0.09959  0.30683
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.984762   0.074361  26.691  < 2e-16 ***
## AnnualTemp  -0.017186   0.005344  -3.216  0.00433 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1754 on 20 degrees of freedom
## Multiple R-squared:  0.3409, Adjusted R-squared:  0.3079
## F-statistic: 10.34 on 1 and 20 DF,  p-value: 0.004333
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43816 -0.06466 -0.00710  0.12129  0.24825
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.921742   0.104737  18.348 4.24e-13 ***
## AnnualTemp                    -0.009633   0.007761  -1.241    0.230
## conditionnatural               0.117098   0.148321   0.789    0.440
## AnnualTemp:conditionnatural -0.013941   0.010676  -1.306    0.208
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1747 on 18 degrees of freedom
## Multiple R-squared:  0.4115, Adjusted R-squared:  0.3134
## F-statistic: 4.195 on 3 and 18 DF,  p-value: 0.02043
```

- There is no significant interaction term or additive term in black carp using **annual temperature**, thus the simple regression model is the best.

```r
# Build the models
black.simple <- lm(log(AAM)~ColdTemp, data = black.clean)
black.int <- lm(log(AAM)~ColdTemp*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Cold temperature

```
##
## Call:
## lm(formula = log(AAM) ~ ColdTemp, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39468 -0.12079 -0.00699  0.08961  0.29562
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.767262   0.035603  49.638   <2e-16 ***
## ColdTemp    -0.011423   0.003084  -3.704   0.0014 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1664 on 20 degrees of freedom
## Multiple R-squared:  0.4069, Adjusted R-squared:  0.3772
## F-statistic: 13.72 on 1 and 20 DF,  p-value: 0.001405
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ ColdTemp * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41954 -0.07945  0.00692  0.11033  0.24744
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 1.797309   0.050646  35.488   <2e-16 ***
## ColdTemp                   -0.007107   0.004471  -1.590    0.129
## conditionnatural           -0.053456   0.071224  -0.751    0.463
## ColdTemp:conditionnatural -0.007989   0.006176  -1.294    0.212
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1661 on 18 degrees of freedom
## Multiple R-squared:  0.4685, Adjusted R-squared:  0.3799
## F-statistic: 5.288 on 3 and 18 DF,  p-value: 0.00861
```

- There is no significant interaction term or additive term in black carp using cold temperature, thus the simple regression model is the best.

```r
# Build the models
black.simple <- lm(log(AAM)~WarmTemp, data = black.clean)
black.int <- lm(log(AAM)~WarmTemp*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Warm temperature

```
##
## Call:
## lm(formula = log(AAM) ~ WarmTemp, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44602 -0.15359  0.03875  0.13845  0.30624
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.13752    0.26482   8.072 1.02e-07 ***
## WarmTemp    -0.01511    0.01098  -1.377    0.184
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2065 on 20 degrees of freedom
## Multiple R-squared:  0.08655,    Adjusted R-squared:  0.04087
## F-statistic: 1.895 on 1 and 20 DF,  p-value: 0.1839
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WarmTemp * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42989 -0.09849  0.03779  0.13637  0.30886
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                1.854098   0.440016   4.214 0.000522 ***
## WarmTemp                  -0.001877   0.018282  -0.103 0.919358
## conditionnatural           0.440537   0.556557   0.792 0.438936
## WarmTemp:conditionnatural -0.021179   0.023082  -0.918 0.370982
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.21 on 18 degrees of freedom
## Multiple R-squared:   0.15,  Adjusted R-squared:  0.008371
## F-statistic: 1.059 on 3 and 18 DF,  p-value: 0.391
```

- Warm temperature is not a significant predictor of black carp age at maturity (same conclusion as Madison found on other Asian carps), thus we remove it in the following analyses.

```r
# Build the models
black.simple <- lm(log(AAM)~average_gdd_0, data = black.clean)
black.int <- lm(log(AAM)~average_gdd_0*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Base 0 annual growing degree day

```
##
## Call:
## lm(formula = log(AAM) ~ average_gdd_0, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46712 -0.09978  0.00423  0.08019  0.32039
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.142e+00  1.276e-01  16.787 2.97e-13 ***
## average_gdd_0 -6.895e-05  2.307e-05  -2.988  0.00727 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1797 on 20 degrees of freedom
## Multiple R-squared:  0.3086, Adjusted R-squared:  0.2741
## F-statistic: 8.929 on 1 and 20 DF,  p-value: 0.007268
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ average_gdd_0 * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.45437 -0.07235 -0.00356  0.12141  0.24464
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     1.968e+00  1.731e-01  11.374 1.19e-09 ***
## average_gdd_0                  -3.055e-05  3.169e-05  -0.964    0.348
## conditionnatural                3.431e-01  2.472e-01   1.388    0.182
## average_gdd_0:conditionnatural -7.515e-05  4.472e-05  -1.681    0.110
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.174 on 18 degrees of freedom
## Multiple R-squared:  0.4168, Adjusted R-squared:  0.3196
## F-statistic: 4.288 on 3 and 18 DF,  p-value: 0.01893
```

- There is no significant interaction term or additive term in black carp ggd0, thus the simple regression model is the best.

```r
# Build the models
black.simple <- lm(log(AAM)~WaterTemp, data = black.clean)
black.int <- lm(log(AAM)~WaterTemp*condition, data = black.clean)
```

```
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Annual water temperature

```
##
## Call:
## lm(formula = log(AAM) ~ WaterTemp, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43891 -0.14624  0.01017  0.12283  0.35815
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.077127   0.137615  15.094 2.15e-12 ***
## WaterTemp   -0.020957   0.009204  -2.277   0.0339 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1926 on 20 degrees of freedom
## Multiple R-squared:  0.2059, Adjusted R-squared:  0.1661
## F-statistic: 5.184 on 1 and 20 DF,  p-value: 0.03393
```

```
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterTemp * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44069 -0.09479  0.00059  0.13315  0.24735
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.915864   0.178963  10.705  3.1e-09 ***
## WaterTemp                   -0.007394   0.011813  -0.626   0.5392
## conditionnatural             0.375292   0.264899   1.417   0.1736
## WaterTemp:conditionnatural -0.031109   0.017745  -1.753   0.0966 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1843 on 18 degrees of freedom
## Multiple R-squared:  0.345,  Adjusted R-squared:  0.2359
## F-statistic: 3.161 on 3 and 18 DF,  p-value: 0.04996
```

- There is no significant interaction term or additive term in black carp annual water temperature, thus the simple regression model is the best.

```
# Build the models
black.simple <- lm(log(AAM)~WaterCold, data = black.clean)
black.int <- lm(log(AAM)~WaterCold*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Cold water temperature

```
##
## Call:
## lm(formula = log(AAM) ~ WaterCold, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49139 -0.09544 -0.02325  0.11838  0.29075
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.923428   0.059900  32.111  < 2e-16 ***
## WaterCold   -0.023142   0.007403  -3.126  0.00532 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1771 on 20 degrees of freedom
## Multiple R-squared:  0.3283, Adjusted R-squared:  0.2947
## F-statistic: 9.773 on 1 and 20 DF,  p-value: 0.005318
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterCold * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.45824 -0.04944 -0.00493  0.10864  0.21744
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  1.862927   0.073076  25.493 1.41e-15 ***
## WaterCold                   -0.009306   0.009299  -1.001   0.3302
## conditionnatural             0.153727   0.113381   1.356   0.1919
## WaterCold:conditionnatural -0.030329   0.013967  -2.171   0.0435 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1651 on 18 degrees of freedom
## Multiple R-squared:  0.4746, Adjusted R-squared:  0.387
## F-statistic: 5.419 on 3 and 18 DF,  p-value: 0.0078
```

- There is a significant interaction term in black carp using cold quarter water temperature, thus we need to further decide based on R2 and AICc values.

```r
# Build the models
black.simple <- lm(log(AAM)~below5, data = black.clean)
black.int <- lm(log(AAM)~below5*condition, data = black.clean)
```

```r
## Look at the summary (especially the slope for each model)
summary(black.simple)
```

Winter duration

```
##
## Call:
## lm(formula = log(AAM) ~ below5, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34614 -0.13083  0.01528  0.08633  0.32716
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.6449045  0.0474995  34.630  < 2e-16 ***
## below5      0.0018623  0.0004611   4.039 0.000643 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1604 on 20 degrees of freedom
## Multiple R-squared:  0.4492, Adjusted R-squared:  0.4217
## F-statistic: 16.31 on 1 and 20 DF,  p-value: 0.0006427
```

```r
summary(black.int)
```

```
##
## Call:
## lm(formula = log(AAM) ~ below5 * condition, data = black.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38625 -0.09196  0.02259  0.12375  0.27285
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              1.7128262  0.0699762  24.477 2.87e-15 ***
## below5                   0.0012706  0.0006640   1.914   0.0717 .
## conditionnatural        -0.1259758  0.0955090  -1.319   0.2037
## below5:conditionnatural  0.0011157  0.0009266   1.204   0.2441
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1608 on 18 degrees of freedom
## Multiple R-squared:  0.5018, Adjusted R-squared:  0.4187
## F-statistic: 6.043 on 3 and 18 DF,  p-value: 0.004946
```

- There is no significant interaction term or additive term in black carp winter duration, thus the simple regression model is the best.

Model selection for cold water temperature (sig interaction)

```r
# Build the models
black.simple <- lm(log(AAM)~WaterCold, data = black.clean)
black.linear <- lm(log(AAM)~WaterCold+condition, data = black.clean)
```

```r
black.int <- lm(log(AAM)~WaterCold*condition, data = black.clean)

## Get a table of corrected AICs and their Akaike weights
models <- list(black.simple, black.linear, black.int)
mod.names <- c('simple linear', 'linear additive', 'interaction')
aictab(cand.set = models, modnames = mod.names, sort = FALSE)
```

```
##
## Model selection based on AICc:
##
##                 K  AICc Delta_AICc AICcWt    LL
## simple linear   3 -8.49       0.00   0.54  7.91
## linear additive 4 -5.76       2.73   0.14  8.06
## interaction     5 -7.48       1.01   0.32 10.62
```

```r
## R^2 value for three models
r_2 <- data.frame(
  Model = c("Simple linear", "Linear additive", "Interaction"),
  R2 = c(summary(black.simple)$adj.r.squared,
         summary(black.linear)$adj.r.squared,
         summary(black.int)$adj.r.squared)
)
kable(r_2)
```

Model	R2
Simple linear	0.2946727
Linear additive	0.2671446
Interaction	0.3870111

- There was no preference between the two models, as the AICc values are within two units of each other.

Temperature predictions and power analyses

```r
## Build the models with three temperature metrics (simple model here)
black.annual <- lm(log(AAM)~AnnualTemp, data = black.clean)
black.cold <- lm(log(AAM)~ColdTemp, data = black.clean)
black.gdd <- lm(log(AAM)~average_gdd_0, data = black.clean)
black.water <- lm(log(AAM)~WaterTemp, data = black.clean)
black.waterC <- lm(log(AAM)~WaterCold, data = black.clean)


## Power analyses - annual
# calculate the coefficient of determination
coe.annual <- summary(black.annual)$adj.r.squared
pwr.f2.test(u = 1, v = 22 - 1 - 1, f2 = coe.annual/(1 -coe.annual),
            sig.level = 0.05)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20
##              f2 = 0.4449434
##       sig.level = 0.05
```

```
##           power = 0.8450604
```

```r
## Power analyses - cold
# calculate the coefficient of determination
coe.cold <- summary(black.cold)$adj.r.squared
pwr.f2.test(u = 1, v = 22 - 1 - 1, f2 = coe.cold/(1 -coe.cold),
            sig.level = 0.05)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20
##              f2 = 0.6056428
##       sig.level = 0.05
##           power = 0.9344838
```

```r
## Power analyses - gdd0
# calculate the coefficient of determination
coe.gdd <- summary(black.gdd)$adj.r.squared
pwr.f2.test(u = 1, v = 22 - 1 - 1, f2 = coe.gdd/(1 -coe.gdd),
            sig.level = 0.05)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20
##              f2 = 0.37756
##       sig.level = 0.05
##           power = 0.7827331
```

```r
pwr.f2.test(u = 1, f2 = coe.gdd/(1 -coe.gdd),
            sig.level = 0.05, power = 0.8)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20.87807
##              f2 = 0.37756
##       sig.level = 0.05
##           power = 0.8
```

```r
## Power analyses - water annual
# calculate the coefficient of determination
coe.water <- summary(black.water)$adj.r.squared
pwr.f2.test(u = 1, v = 22 - 1 - 1, f2 = coe.water/(1 -coe.water),
            sig.level = 0.05)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20
##              f2 = 0.1992484
##       sig.level = 0.05
```

```
##           power = 0.5131047
```

```r
pwr.f2.test(u = 1, f2 = coe.water/(1 -coe.water),
            sig.level = 0.05, power = 0.8)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 39.40371
##              f2 = 0.1992484
##       sig.level = 0.05
##           power = 0.8
```

```r
## Power analyses - water cold
# calculate the coefficient of determination
coe.waterC <- summary(black.waterC)$adj.r.squared
pwr.f2.test(u = 1, v = 22 - 1 - 1, f2 = coe.waterC/(1 -coe.waterC),
            sig.level = 0.05)
```

```
##
##      Multiple regression power calculation
##
##               u = 1
##               v = 20
##              f2 = 0.4177814
##       sig.level = 0.05
##           power = 0.8221123
```

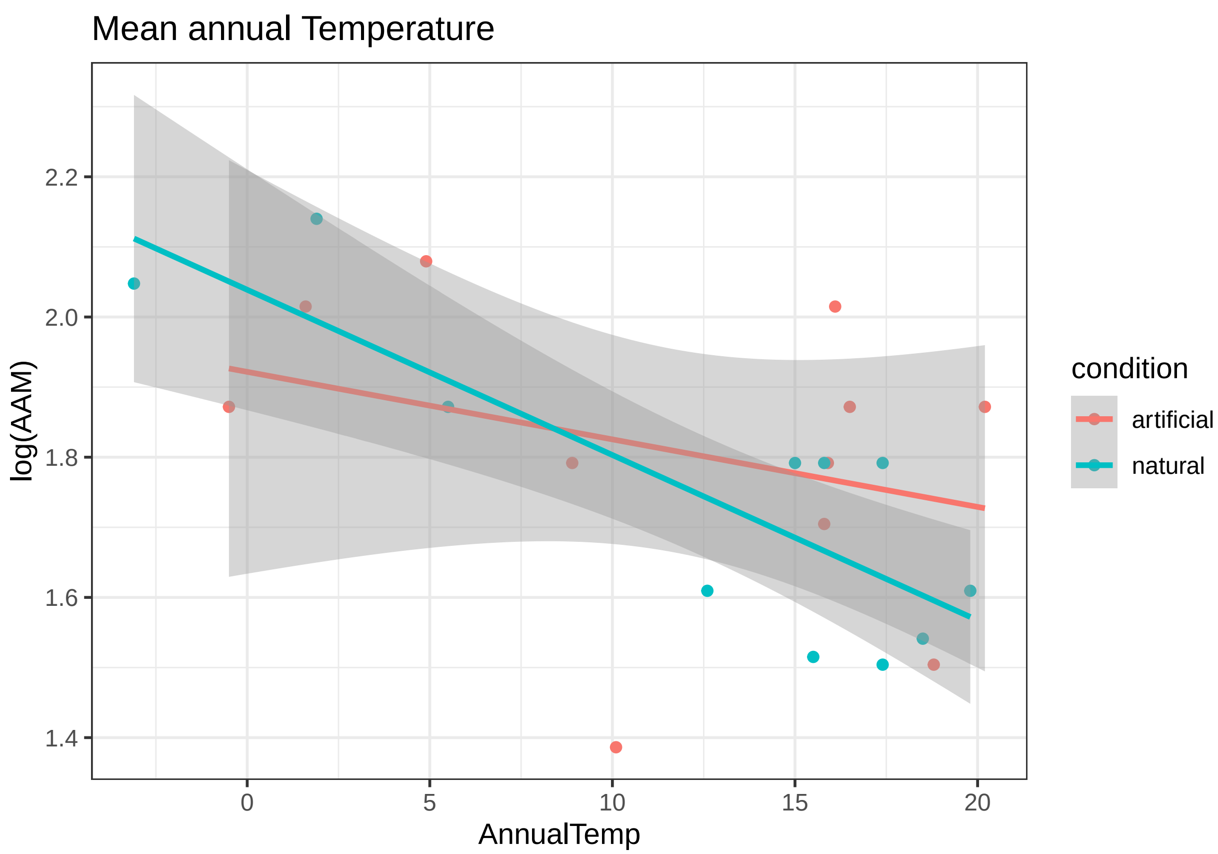
- Power analyses suggested that our current sample size is sufficient enough to produce a strong statistical power for annual temperauture, cold temperature, gdd0 (?), and cold water temperature.
- Annual water?

Black carp graphs with two conditions separated

We separated the black carp dataset into two based on conditions. Since there was no preference over the four models, we used the simple linear model on each set of the data.

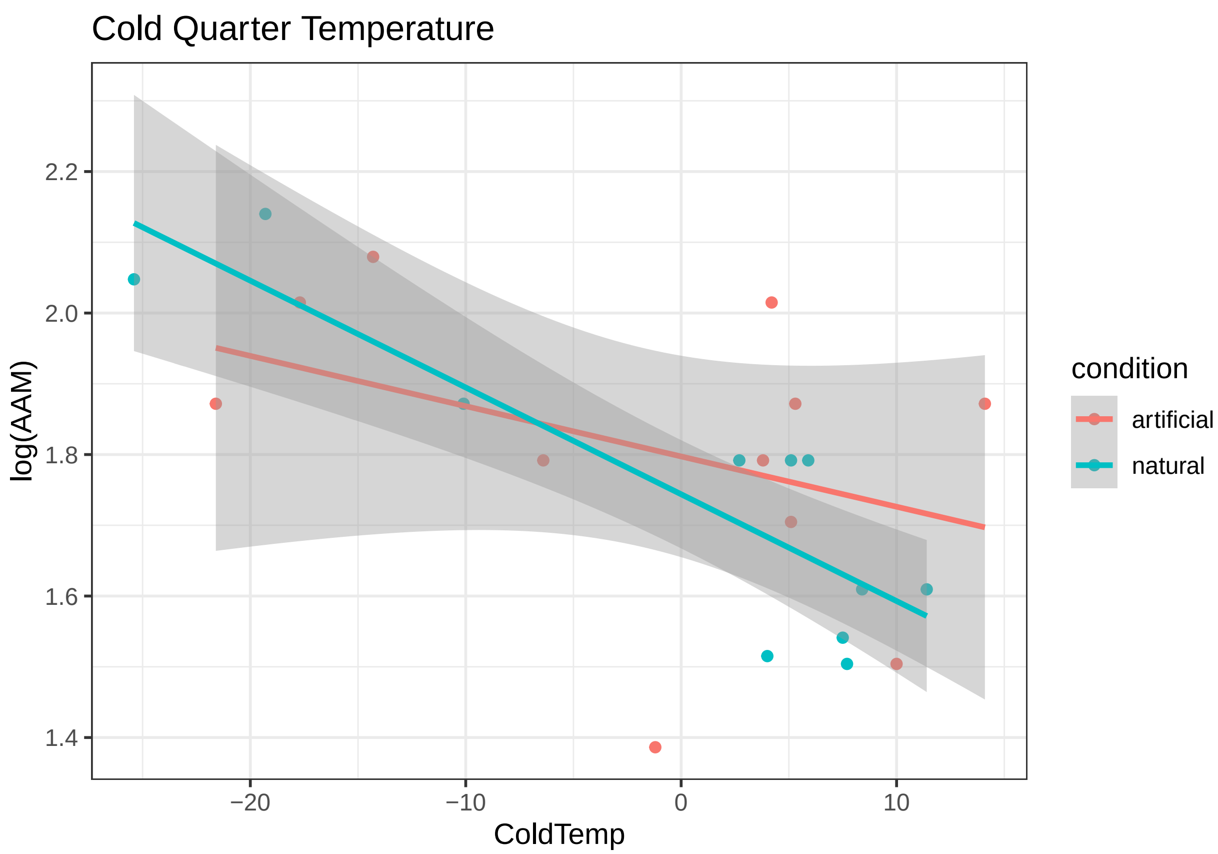
```r
## Annual temperature
ggplot(black.clean, aes(x = AnnualTemp, y = log(AAM), color = condition))+
  geom_point()+
  geom_smooth(method = "lm")+
  theme_bw()+
  labs(title = "Mean annual Temperature")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



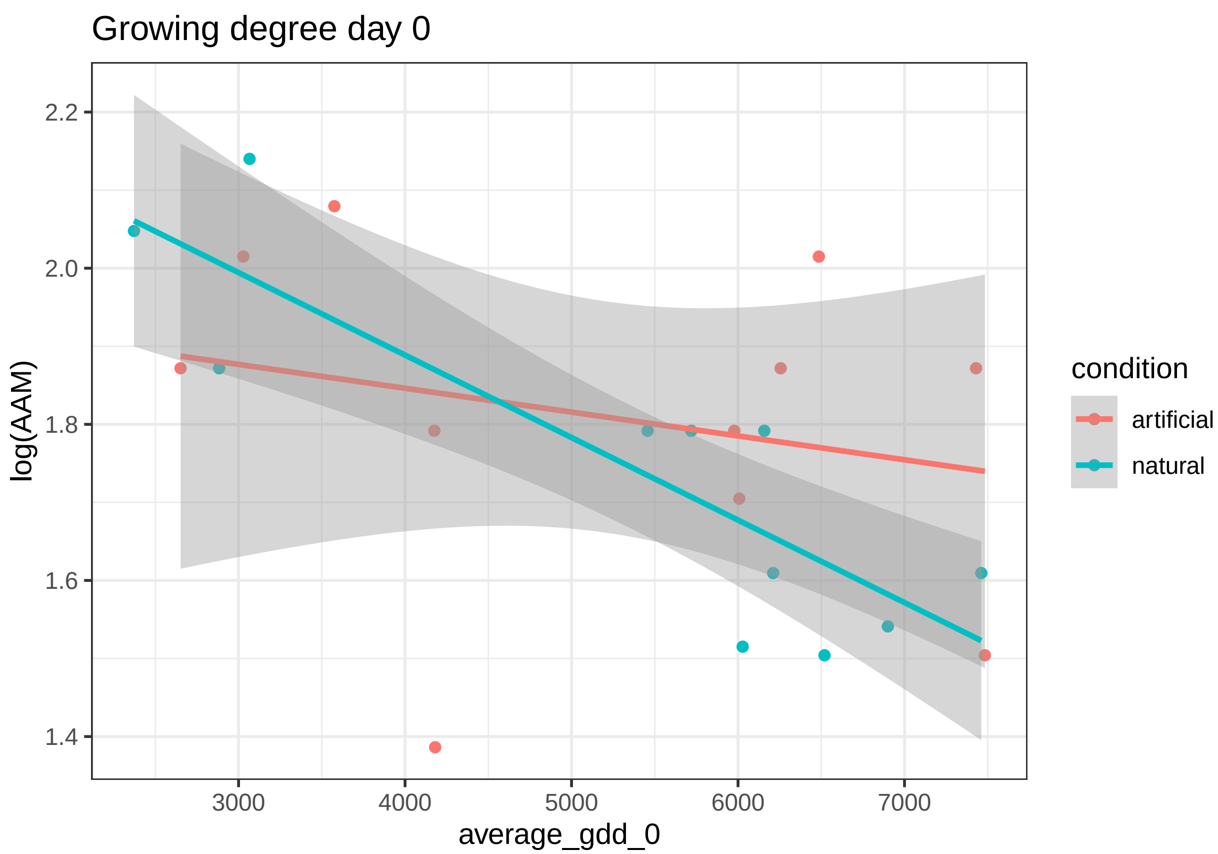

```r
## Cold temperature
ggplot(black.clean, aes(x = ColdTemp, y = log(AAM), color = condition))+
  geom_point()+
  geom_smooth(method = "lm")+
  theme_bw()+
  labs(title = "Cold Quarter Temperature")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



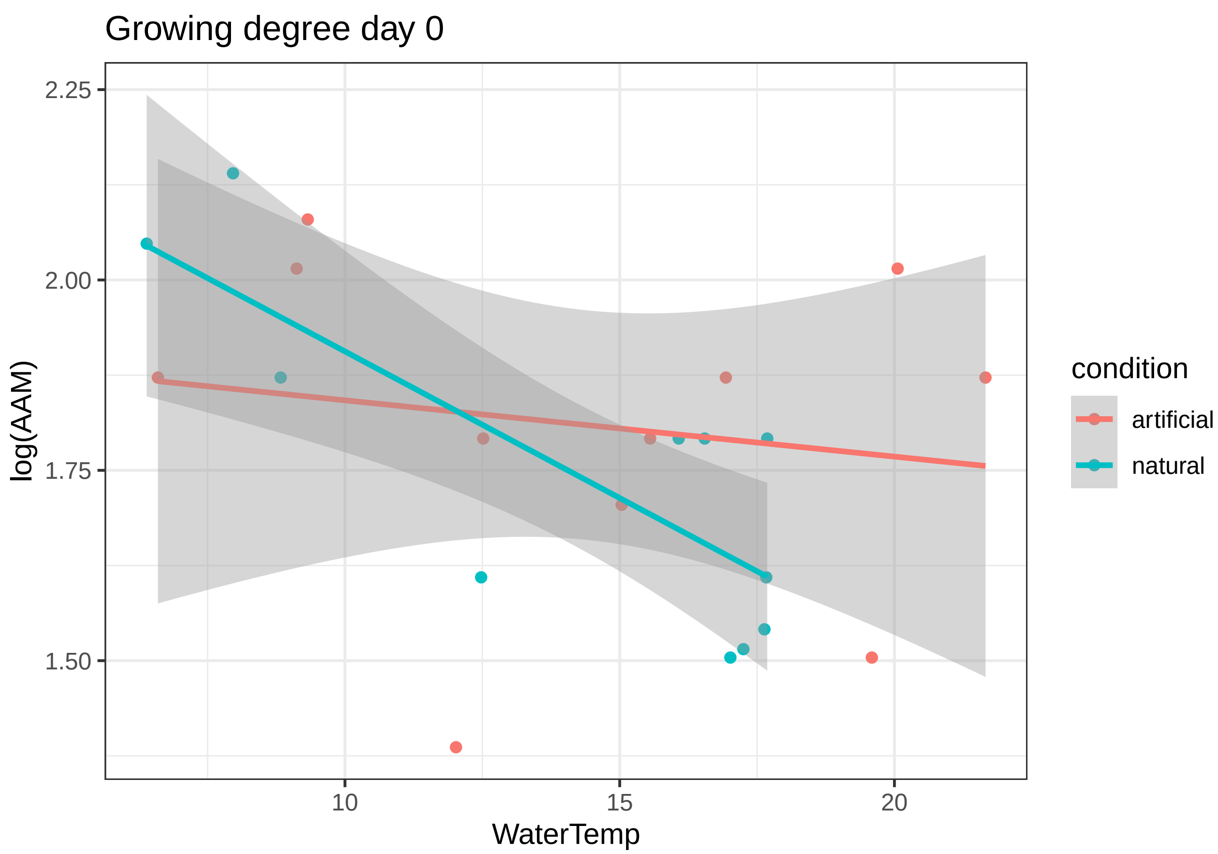

```r
## GDD0
ggplot(black.clean, aes(x = average_gdd_0, y = log(AAM), color = condition))+
  geom_point()+
  geom_smooth(method = "lm")+
  theme_bw()+
  labs(title = "Growing degree day 0")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



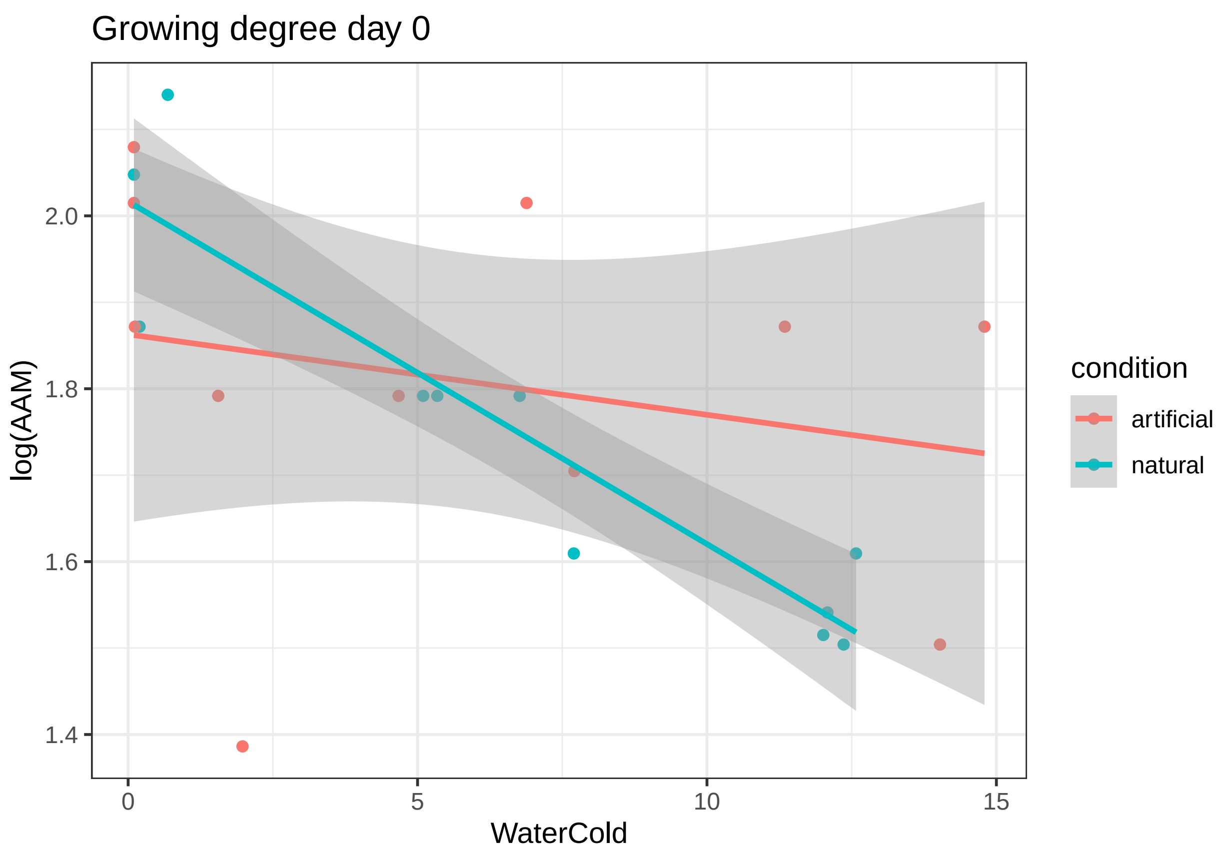

```r
## Annual water temperature
ggplot(black.clean, aes(x = WaterTemp, y = log(AAM), color = condition))+
  geom_point()+
  geom_smooth(method = "lm")+
  theme_bw()+
  labs(title = "Growing degree day 0")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
## Cold water temperature
ggplot(black.clean, aes(x = WaterCold, y = log(AAM), color = condition))+
  geom_point()+
  geom_smooth(method = "lm")+
  theme_bw()+
  labs(title = "Growing degree day 0")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Now that we have seen that the artificial condition data seems to have a larger spread, we would like to run a regression with only the natural conditions to see how much the R2 value can improve? Would it be similar to other Asian carp species (around 0.6)?

```
## Separate into two data sets
black.natural <- black.clean[black.clean$condition == "natural",]
black.artificial <- black.clean[black.clean$condition == "artificial",]


## Run the models
black.annual.n <- lm(log(AAM)~AnnualTemp, data = black.natural)
black.cold.n <- lm(log(AAM)~ColdTemp, data = black.natural)
black.gdd.n <- lm(log(AAM)~average_gdd_0, data = black.natural)
black.water.n <- lm(log(AAM)~WaterTemp, data = black.natural)
black.waterC.n <- lm(log(AAM)~WaterCold, data = black.natural)


black.annual.a <- lm(log(AAM)~AnnualTemp, data = black.artificial)
black.cold.a <- lm(log(AAM)~ColdTemp, data = black.artificial)
black.gdd.a <- lm(log(AAM)~average_gdd_0, data = black.artificial)
black.water.a <- lm(log(AAM)~WaterTemp, data = black.artificial)
black.waterC.a <- lm(log(AAM)~WaterCold, data = black.artificial)


## Compare the model parameters
summary(black.annual.n)
```

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp, data = black.natural)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15831 -0.09440 -0.03738  0.11596  0.16311
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.038839   0.075970  26.838  6.7e-10 ***
## AnnualTemp  -0.023574   0.005304  -4.445  0.00161 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1264 on 9 degrees of freedom
## Multiple R-squared:  0.687,  Adjusted R-squared:  0.6523
## F-statistic: 19.76 on 1 and 9 DF,  p-value: 0.001612
```

```r
summary(black.annual.a)
```

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp, data = black.artificial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43816 -0.05978  0.02318  0.12682  0.24825
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.921742   0.127277  15.099 1.07e-07 ***
## AnnualTemp  -0.009633   0.009431  -1.021    0.334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2124 on 9 degrees of freedom
## Multiple R-squared:  0.1039, Adjusted R-squared:  0.004303
## F-statistic: 1.043 on 1 and 9 DF,  p-value: 0.3337
```

```r
summary(black.cold.n)
```

```
##
## Call:
## lm(formula = log(AAM) ~ ColdTemp, data = black.natural)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.168342 -0.084535 -0.007609  0.096764  0.136973
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.743853   0.033825  51.555 1.95e-12 ***
## ColdTemp    -0.015096   0.002878  -5.246 0.000531 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1122 on 9 degrees of freedom
## Multiple R-squared:  0.7536, Adjusted R-squared:  0.7262
## F-statistic: 27.52 on 1 and 9 DF,  p-value: 0.0005305
```

```r
summary(black.cold.a)
```

```
##
## Call:
## lm(formula = log(AAM) ~ ColdTemp, data = black.artificial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41954 -0.06766  0.02146  0.14343  0.24744
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.797309   0.062927  28.562 3.85e-10 ***
## ColdTemp    -0.007107   0.005555  -1.279    0.233
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2063 on 9 degrees of freedom
## Multiple R-squared:  0.1539, Adjusted R-squared:  0.05987
## F-statistic: 1.637 on 1 and 9 DF,  p-value: 0.2328
```

```r
summary(black.gdd.n)
```

```
##
## Call:
## lm(formula = log(AAM) ~ average_gdd_0, data = black.natural)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15923 -0.08195 -0.01309  0.08566  0.15266
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.311e+00  1.172e-01  19.729 1.02e-08 ***
## average_gdd_0 -1.057e-04  2.094e-05  -5.048 0.000692 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1154 on 9 degrees of freedom
## Multiple R-squared:  0.739,  Adjusted R-squared:   0.71
## F-statistic: 25.48 on 1 and 9 DF,  p-value: 0.000692
```

```r
summary(black.gdd.a)
```

```
##
## Call:
## lm(formula = log(AAM) ~ average_gdd_0, data = black.artificial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.45437 -0.06458  0.00598  0.13471  0.24464
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.968e+00  2.161e-01   9.108 7.75e-06 ***
## average_gdd_0 -3.055e-05  3.957e-05  -0.772     0.46
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2172 on 9 degrees of freedom
## Multiple R-squared:  0.0621, Adjusted R-squared:  -0.04212
## F-statistic: 0.5959 on 1 and 9 DF,  p-value: 0.4599
```

```r
summary(black.water.n)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterTemp, data = black.natural)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20126 -0.09558 -0.00148  0.12860  0.18153
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.291156   0.146522  15.637 7.86e-08 ***
## WaterTemp   -0.038503   0.009934  -3.876  0.00375 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1383 on 9 degrees of freedom
## Multiple R-squared:  0.6254, Adjusted R-squared:  0.5837
## F-statistic: 15.02 on 1 and 9 DF,  p-value: 0.003755
```

```r
summary(black.water.a)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterTemp, data = black.artificial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44069 -0.06575  0.00470  0.14127  0.24735
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.915864   0.214544   8.930  9.1e-06 ***
## WaterTemp   -0.007394   0.014162  -0.522    0.614
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.221 on 9 degrees of freedom
## Multiple R-squared:  0.0294, Adjusted R-squared:  -0.07845
## F-statistic: 0.2726 on 1 and 9 DF,  p-value: 0.6142
```

```r
summary(black.waterC.n)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterCold, data = black.natural)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13705 -0.02420 -0.01319  0.03911  0.15057
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.01665    0.04467  45.145 6.42e-12 ***
## WaterCold   -0.03963    0.00537  -7.381 4.19e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08508 on 9 degrees of freedom
## Multiple R-squared:  0.8582, Adjusted R-squared:  0.8425
## F-statistic: 54.48 on 1 and 9 DF,  p-value: 4.188e-05
```

```r
summary(black.waterC.a)
```

```
##
## Call:
## lm(formula = log(AAM) ~ WaterCold, data = black.artificial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.45824 -0.07156  0.00996  0.14973  0.21744
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.862927   0.096240   19.36 1.21e-08 ***
## WaterCold   -0.009306   0.012247   -0.76    0.467
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2175 on 9 degrees of freedom
## Multiple R-squared:  0.06029,    Adjusted R-squared:  -0.04412
## F-statistic: 0.5775 on 1 and 9 DF,  p-value: 0.4667
```

```r
## Comparing R^2
r_2 <- data.frame(
  Temperature = c("Annual", "Cold", "GGDO", "Water", "Water cold"),
  Black.combined = c(0.31, 0.38, 0.27, 0.17, 0.29),
  Black.natural = c(summary(black.annual.n)$adj.r.squared,
                    summary(black.cold.n)$adj.r.squared,
                    summary(black.gdd.n)$adj.r.squared,
                    summary(black.water.n)$adj.r.squared,
                    summary(black.waterC.n)$adj.r.squared),
  Black.artificial = c(summary(black.annual.a)$adj.r.squared,
                       summary(black.cold.a)$adj.r.squared,
                       summary(black.gdd.a)$adj.r.squared,
                       summary(black.water.a)$adj.r.squared,
```

```r
                       summary(black.waterC.a)$adj.r.squared),
  Asian.total = c(0.6, 0.57, 0.62, NA, NA)
)
kable(r_2)
```

Temperature	Black.combined	Black.natural	Black.artificial	Asian.total
Annual	0.31	0.6522554	0.0043026	0.60
Cold	0.38	0.7261770	0.0598742	0.57
GGD0	0.27	0.7100129	-0.0421154	0.62
Water	0.17	0.5837250	-0.0784478	NA
Water cold	0.29	0.8424617	-0.0441176	NA

- It turned out that removing the artificial conditions greatly increase the predictive power of the model for all five temperatures. While the artificial model alone did not even have a significant relationship.
- The resulting R2 (for the natural environments) is similar to that of the Asian carp.

SECTION 2: Asian carp

1. We followed the same work flow for other Asian carp species. However, for Asian carp, we subsample at a distance of 250 km to avoid spatial autocorrelation.

Data cleaning and matrices for results

```r
## Look at the spatial codes for the current asian carp data
asian.carp.clean <- asian.carp %>%
  filter(Condition %in% c("natural", "artificial"))

table(asian.carp.clean$Code_Str)
```

```
##
##  A AA AB AC AD AE AF AG AH AI AJ AK AL  B  C  D  E  F  G  H  I  J  K  L  M  N
##  4  3  3  2  1  1  1  3  1  3  1  3  2  1  1  1  1  2  1  1  3  4  1  2  2  1
##  O  P  Q  R  S  T  U  V  W  X  Y  Z
##  1  1  1  3  1  1  3  3  1  1  1  1
```

```r
# Stratified sub-sampling gives 21 artificial and 17 natural conditions.

# Simple linear model - slope, intercept, p, r2, AICc, F stats
linear.results <- matrix(NA,1000,18)
colnames(linear.results) <- c("slope.a",
                                    "intercept.a",
                                    "p for slope.a",
                                    "r2.a",
                                    "AICc.a",
                                    "F stats.a",
                                "slope.c",
                                    "intercept.c",
                                    "p for slope.c",
                                    "r2.c",
                                    "AICc.c",
                                    "F stats.c",
                                "slope.g",
                                    "intercept.g",
```

```
                                    "p for slope.g",
                                    "r2.g",
                                    "AICc.g",
                                    "F stats.g")

# Interaction model - slope, intercept, p(slope), p(interaction), r2, AICc
int.results <- matrix(NA,1000,24)
colnames(int.results) <- c("slope.a",
                                    "intercept.a",
                                    "p for slope.a",
                                    "r2.a",
                                    "AICc.a",
                                    "F stats.a",
                                    "p additive term.a",
                                    "p interaction term.a",
                           "slope.c",
                                    "intercept.c",
                                    "p for slope.c",
                                    "r2.c",
                                    "AICc.c",
                                    "F stats.c",
                                    "p additive term.c",
                                    "p interaction term.c",
                           "slope.g",
                                    "intercept.g",
                                    "p for slope.g",
                                    "r2.g",
                                    "AICc.g",
                                    "F stats.g",
                                    "p additive term.g",
                                    "p interaction term.g")
```

- Stratified sub-sampling gives 21 artificial and 17 natural conditions.

Define the models

```
# For 1000 iterations
for(i in 1:1000){
  sub <- asian.carp.clean %>% group_by(Code_Str) %>% sample_n(size=1)

  ## annual
  reg.linear.annual <- lm(log(AAM)~AnnualTemp, data = sub)
  reg.int.annual <- lm(log(AAM)~AnnualTemp*Condition, data = sub)

  # simple linear model
  linear.results[i,1]<-summary(reg.linear.annual)$coef[2,1] #slope
  linear.results[i,2]<-summary(reg.linear.annual)$coef[1,1] #intercept
  linear.results[i,3]<-summary(reg.linear.annual)$coef[2,4] #p-value
  linear.results[i,4]<-summary(reg.linear.annual)$adj.r.squared #r2
  linear.results[i,5]<-as.numeric(AICc(reg.linear.annual)) #AICc
  linear.results[i,6]<-summary(reg.linear.annual)$fstatistic[1] #F stats

  # interaction model
  int.results[i,1]<-summary(reg.int.annual)$coef[2,1] #slope
```

```r
int.results[i,2]<-summary(reg.int.annual)$coef[1,1] #intercept
int.results[i,3]<-summary(reg.int.annual)$coef[2,4] #p(slope)
int.results[i,4]<-summary(reg.int.annual)$adj.r.squared #r2
int.results[i,5]<-as.numeric(AICc(reg.int.annual)) #AICc
int.results[i,6]<-summary(reg.int.annual)$fstatistic[1] #F stats
int.results[i,7]<-summary(reg.int.annual)$coef[3,4] #p(additive term)
int.results[i,8]<-summary(reg.int.annual)$coef[4,4] #p(interaction term)


## cold
reg.linear.cold <- lm(log(AAM)~ColdTemp, data = sub)
reg.int.cold <- lm(log(AAM)~ColdTemp*Condition, data = sub)

# simple linear model
linear.results[i,7]<-summary(reg.linear.cold)$coef[2,1] #slope
linear.results[i,8]<-summary(reg.linear.cold)$coef[1,1] #intercept
linear.results[i,9]<-summary(reg.linear.cold)$coef[2,4] #p-value
linear.results[i,10]<-summary(reg.linear.cold)$adj.r.squared #r2
linear.results[i,11]<-as.numeric(AICc(reg.linear.cold)) #AICc
linear.results[i,12]<-summary(reg.linear.cold)$fstatistic[1] #F stats

# interaction model
int.results[i,9]<-summary(reg.int.cold)$coef[2,1] #slope
int.results[i,10]<-summary(reg.int.cold)$coef[1,1] #intercept
int.results[i,11]<-summary(reg.int.cold)$coef[2,4] #p(slope)
int.results[i,12]<-summary(reg.int.cold)$adj.r.squared #r2
int.results[i,13]<-as.numeric(AICc(reg.int.cold)) #AICc
int.results[i,14]<-summary(reg.int.cold)$fstatistic[1] #F stats
int.results[i,15]<-summary(reg.int.cold)$coef[3,4] #p(additive term)
int.results[i,16]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)


## gdd0
reg.linear.gdd <- lm(log(AAM)~average_gdd_0, data = sub)
reg.int.gdd <- lm(log(AAM)~average_gdd_0*Condition, data = sub)

# simple linear model
linear.results[i,13]<-summary(reg.linear.gdd)$coef[2,1] #slope
linear.results[i,14]<-summary(reg.linear.gdd)$coef[1,1] #intercept
linear.results[i,15]<-summary(reg.linear.gdd)$coef[2,4] #p-value
linear.results[i,16]<-summary(reg.linear.gdd)$adj.r.squared #r2
linear.results[i,17]<-as.numeric(AICc(reg.linear.gdd)) #AICc
linear.results[i,18]<-summary(reg.linear.gdd)$fstatistic[1] #F stats

# interaction model
int.results[i,17]<-summary(reg.int.gdd)$coef[2,1] #slope
int.results[i,18]<-summary(reg.int.gdd)$coef[1,1] #intercept
int.results[i,19]<-summary(reg.int.gdd)$coef[2,4] #p(slope)
int.results[i,20]<-summary(reg.int.gdd)$adj.r.squared #r2
int.results[i,21]<-as.numeric(AICc(reg.int.gdd)) #AICc
int.results[i,22]<-summary(reg.int.gdd)$fstatistic[1] #F stats
int.results[i,23]<-summary(reg.int.gdd)$coef[3,4] #p(additive term)
int.results[i,24]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)
```

```
}
```

Check the slopes and additive/interaction terms

Now we need to:

1. Check if the additive or interaction term is significant.
2. Check if the relationship is significant between age at maturity and temperature (significant slope).

```r
## Additive/interaction term
# annual
mean(unique(int.results[,"p additive term.a"]))
```

```
## [1] 0.7581767
```

```r
table(int.results[,"p additive term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.a"]))
```

```
## [1] 0.6639399
```

```r
table(int.results[,"p interaction term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# cold
mean(unique(int.results[,"p additive term.c"]))
```

```
## [1] 0.60517
```

```r
table(int.results[,"p additive term.c"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.c"]))
```

```
## [1] 0.7935565
```

```r
table(int.results[,"p interaction term.c"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# gdd
mean(unique(int.results[,"p additive term.g"]))
```

```
## [1] 0.3696879
```

```r
table(int.results[,"p additive term.g"] < 0.05)
```

```
##
## FALSE
```

```
##  1000
```

```r
mean(unique(int.results[,"p interaction term.g"]))
```

```
## [1] 0.7935565
```

```r
table(int.results[,"p interaction term.g"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# We found no siginficant additive or interaction term


### Now let's look at the simple regression model:

## p-value for the slope (also is the p-value for the entire model)
# annual
mean(unique(linear.results[,"p for slope.a"]))
```

```
## [1] 1.574099e-07
```

```r
table(linear.results[,"p for slope.a"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# cold
mean(unique(linear.results[,"p for slope.c"]))
```

```
## [1] 4.078877e-07
```

```r
table(linear.results[,"p for slope.c"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# gdd
mean(unique(linear.results[,"p for slope.g"]))
```

```
## [1] 6.414672e-08
```

```r
table(linear.results[,"p for slope.g"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
## F statistics
mean(unique(linear.results[,"F stats.a"]))
```

```
## [1] 44.51514
```

```r
mean(unique(linear.results[,"F stats.c"]))
```

```
## [1] 39.98203
```

```r
mean(unique(linear.results[,"F stats.g"]))
```

```
## [1] 47.70175
```

```r
## Slope
mean(unique(linear.results[,"slope.a"]))
```

```
## [1] -0.04167286
```

```r
mean(unique(linear.results[,"slope.c"]))
```

```
## [1] -0.02534065
```

```r
mean(unique(linear.results[,"slope.g"]))
```

```
## [1] -0.0001514909
```

```r
## Intercept
mean(unique(linear.results[,"intercept.a"]))
```

```
## [1] 2.006672
```

```r
mean(unique(linear.results[,"intercept.c"]))
```

```
## [1] 1.504725
```

```r
mean(unique(linear.results[,"intercept.g"]))
```

```
## [1] 2.306622
```

- No significant additive or interaction term in the multiple linear regression model. As a result, we should use the simple regression model.
- In a simple regression model, significant slope using all three temperatures.

Compare the R2

```r
# annual
r2annual <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,4])),
         mean(unique(int.results[,4])))
)
kable(r2annual)
```

Model	R2
Simple linear	0.5389231
Interaction	0.5164976

```r
# cold
r2cold <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,10])),
         mean(unique(int.results[,12])))
)
kable(r2cold)
```

Model	R2
Simple linear	0.5117198

Model	R2
Interaction	0.4893225

```
# gdd
r2gdd <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,16])),
         mean(unique(int.results[,20])))
)
kable(r2gdd)
```

Model	R2
Simple linear	0.5569595
Interaction	0.5465196

Compare among the temperatures (SIMPLE MODEL ONLY)

```
## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  R2 = c(mean(unique(linear.results[,"r2.a"])),
         mean(unique(linear.results[,"r2.c"])),
         mean(unique(linear.results[,"r2.g"])))
)
kable(r2)
```

Temperature	R2
Annual	0.5389231
Cold	0.5117198
GDD0	0.5569595

```
## Check the AICc scores and akaike weights in ONLY LINEAR MODEL
weight.matrix <- matrix(NA, 1000, 3)
count <- numeric(0)

for (i in 1:1000) {
  # Create a list of the aic values of the current iteration
  aic_value <- c(linear.results[i,5], linear.results[i,11],
                 linear.results[i,17])

  ## check the akaike weights
  weight <- compute_akaike_weights(aic_value)
  weight.matrix[i,c(1,2,3)] <- round(weight[c(1,2,3)],3)

  ## check the AICc scores
  indexing <- compare_aic_scores(aic_value)
  if (indexing != -999) {
    count <- c(count, indexing)
  }
}
```

```
summary(weight.matrix)
```

```
##        V1               V2               V3
##  Min.   :0.1120   Min.   :0.0320   Min.   :0.3310
##  1st Qu.:0.2290   1st Qu.:0.0720   1st Qu.:0.5310
##  Median :0.2880   Median :0.0935   Median :0.6130
##  Mean   :0.2942   Mean   :0.1016   Mean   :0.6042
##  3rd Qu.:0.3520   3rd Qu.:0.1230   3rd Qu.:0.6853
##  Max.   :0.5480   Max.   :0.2970   Max.   :0.8200
```

```
table(count)
```

```
## count
##   3
## 311
```

- With the simple linear model, around 30% of the time when using GDD is preferred over using ColdTemp.

Two conditions separated

```
## Separate the two conditions
asian.natural <- asian.carp.clean[asian.carp.clean$Condition == "natural",]
asian.artificial <- asian.carp.clean[asian.carp.clean$Condition == "artificial",]

# Check sp
table(asian.natural$Code_Str)
```

```
##
## A B C D E F G H I J K L M N O P Q
## 4 1 1 1 1 2 1 1 3 4 1 2 2 1 1 1 1
```

```
table(asian.artificial$Code_Str)
```

```
##
## AA AB AC AD AE AF AG AH AI AJ AK AL  R  S  T  U  V  W  X  Y  Z
##  3  3  2  1  1  1  3  1  3  1  3  2  3  1  1  3  3  1  1  1  1
```

```
# store the results
r2.natural <- matrix(NA, 1000, 3)
r2.artificial <- matrix(NA, 1000, 3)

p.natural <- matrix(NA, 1000, 3)
p.artificial <- matrix(NA, 1000, 3)

F.natural <- matrix(NA, 1000, 3)
F.artificial <- matrix(NA, 1000, 3)


## Separating environment for 1000 iterations
for(i in 1:1000){

  ## natural
  sub.n <- asian.natural %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
```

```
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.n)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.n)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.n)

  # r2
  r2.natural[i,1]<-summary(reg.annual)$adj.r.squared
  r2.natural[i,2]<-summary(reg.cold)$adj.r.squared
  r2.natural[i,3]<-summary(reg.gdd)$adj.r.squared

  # p
  p.natural[i,1]<-summary(reg.annual)$coef[2,4]
  p.natural[i,2]<-summary(reg.cold)$coef[2,4]
  p.natural[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
  F.natural[i,1] <- summary(reg.annual)$fstatistic[1]
  F.natural[i,2] <- summary(reg.cold)$fstatistic[1]
  F.natural[i,3] <- summary(reg.gdd)$fstatistic[1]


  ## artificial
  sub.a <- asian.artificial %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.a)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.a)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.a)

  # r2
  r2.artificial[i,1]<-summary(reg.annual)$adj.r.squared
  r2.artificial[i,2]<-summary(reg.cold)$adj.r.squared
  r2.artificial[i,3]<-summary(reg.gdd)$adj.r.squared

  # p
  p.artificial[i,1]<-summary(reg.annual)$coef[2,4]
  p.artificial[i,2]<-summary(reg.cold)$coef[2,4]
  p.artificial[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
  F.artificial[i,1] <- summary(reg.annual)$fstatistic[1]
  F.artificial[i,2] <- summary(reg.cold)$fstatistic[1]
  F.artificial[i,3] <- summary(reg.gdd)$fstatistic[1]
}


## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.R2 = c(mean(unique(r2.natural[,1])),
                 mean(unique(r2.natural[,2])),
                 mean(unique(r2.natural[,3]))),
  Artificial.R2 = c(mean(unique(r2.artificial[,1])),
                    mean(unique(r2.artificial[,2])),
```

```r
                    mean(unique(r2.artificial[,3])))
)
kable(r2)
```

Temperature	Natural.R2	Artificial.R2
Annual	0.4791394	0.5263301
Cold	0.4405415	0.5092780
GDD0	0.4401723	0.6223827

```r
## Compare the F stats
Fstats <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.F = c(mean(unique(F.natural[,1])),
                mean(unique(F.natural[,2])),
                mean(unique(F.natural[,3]))),
  Artificial.F = c(mean(unique(F.artificial[,1])),
                   mean(unique(F.artificial[,2])),
                   mean(unique(F.artificial[,3])))
)
kable(Fstats)
```

Temperature	Natural.F	Artificial.F
Annual	16.16041	23.39163
Cold	13.87143	22.00896
GDD0	13.91589	34.19097

```r
## Check the p values
table(p.natural[,1] < 0.05) # natural, annual
```

```
##
## TRUE
## 1000
```

```r
table(p.natural[,2] < 0.05) # natural, cold
```

```
##
## TRUE
## 1000
```

```r
table(p.natural[,3] < 0.05) # natural, gdd
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,1] < 0.05) # artificial, annual
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,2] < 0.05) # artificial, cold
```

```
##
## TRUE
## 1000
```

```
table(p.artificial[,3] < 0.05) # artificial, gdd
```

```
##
## TRUE
## 1000
```

- Natural and Artificial have similar R2 values when separated.

SECTION 3: Grass carp

Data cleaning and matrices for results

```
Grass.clean <- Grass %>%
  filter(Condition %in% c("natural", "artificial"))

table(Grass.clean$Code_Str)
```

```
##
##  A AA AB AC AD AF AG AI AJ AK AL  B  E  F  G  I  J  L  M  N  O  P  R  S  T  U
##  2  1  2  1  1  1  1  1  1  1  1  1  1  1  1  2  2  2  1  1  1  1  1  1  1  1
##  V  W  Y  Z
##  1  1  1  1
```

```
# Simple linear model - slope, intercept, p, r2, AICc, F stats
linear.results <- matrix(NA,1000,18)
colnames(linear.results) <- c("slope.a",
                                  "intercept.a",
                                  "p for slope.a",
                                  "r2.a",
                                  "AICc.a",
                                  "F stats.a",
                              "slope.c",
                                  "intercept.c",
                                  "p for slope.c",
                                  "r2.c",
                                  "AICc.c",
                                  "F stats.c",
                              "slope.g",
                                  "intercept.g",
                                  "p for slope.g",
                                  "r2.g",
                                  "AICc.g",
                                  "F stats.g")

# Interaction model - slope, intercept, p(slope), p(interaction), r2, AICc
int.results <- matrix(NA,1000,24)
colnames(int.results) <- c("slope.a",
                                  "intercept.a",
                                  "p for slope.a",
                                  "r2.a",
                                  "AICc.a",
                                  "F stats.a",
```

```
                                    "p additive term.a",
                                    "p interaction term.a",
                            "slope.c",
                                    "intercept.c",
                                    "p for slope.c",
                                    "r2.c",
                                    "AICc.c",
                                    "F stats.c",
                                    "p additive term.c",
                                    "p interaction term.c",
                            "slope.g",
                                    "intercept.g",
                                    "p for slope.g",
                                    "r2.g",
                                    "AICc.g",
                                    "F stats.g",
                                    "p additive term.g",
                                    "p interaction term.g")
```

Define the models

```
# For 1000 iterations
for(i in 1:1000){
  sub <- Grass.clean %>% group_by(Code_Str) %>% sample_n(size=1)

  ## annual
  reg.linear.annual <- lm(log(AAM)~AnnualTemp, data = sub)
  reg.int.annual <- lm(log(AAM)~AnnualTemp*Condition, data = sub)

  # simple linear model
  linear.results[i,1]<-summary(reg.linear.annual)$coef[2,1] #slope
  linear.results[i,2]<-summary(reg.linear.annual)$coef[1,1] #intercept
  linear.results[i,3]<-summary(reg.linear.annual)$coef[2,4] #p-value
  linear.results[i,4]<-summary(reg.linear.annual)$adj.r.squared #r2
  linear.results[i,5]<-as.numeric(AICc(reg.linear.annual)) #AICc
  linear.results[i,6]<-summary(reg.linear.annual)$fstatistic[1] #F stats

  # interaction model
  int.results[i,1]<-summary(reg.int.annual)$coef[2,1] #slope
  int.results[i,2]<-summary(reg.int.annual)$coef[1,1] #intercept
  int.results[i,3]<-summary(reg.int.annual)$coef[2,4] #p(slope)
  int.results[i,4]<-summary(reg.int.annual)$adj.r.squared #r2
  int.results[i,5]<-as.numeric(AICc(reg.int.annual)) #AICc
  int.results[i,6]<-summary(reg.int.annual)$fstatistic[1] #F stats
  int.results[i,7]<-summary(reg.int.annual)$coef[3,4] #p(additive term)
  int.results[i,8]<-summary(reg.int.annual)$coef[4,4] #p(interaction term)


  ## cold
  reg.linear.cold <- lm(log(AAM)~ColdTemp, data = sub)
  reg.int.cold <- lm(log(AAM)~ColdTemp*Condition, data = sub)

  # simple linear model
```

```
  linear.results[i,7]<-summary(reg.linear.cold)$coef[2,1] #slope
  linear.results[i,8]<-summary(reg.linear.cold)$coef[1,1] #intercept
  linear.results[i,9]<-summary(reg.linear.cold)$coef[2,4] #p-value
  linear.results[i,10]<-summary(reg.linear.cold)$adj.r.squared #r2
  linear.results[i,11]<-as.numeric(AICc(reg.linear.cold)) #AICc
  linear.results[i,12]<-summary(reg.linear.cold)$fstatistic[1] #F stats

  # interaction model
  int.results[i,9]<-summary(reg.int.cold)$coef[2,1] #slope
  int.results[i,10]<-summary(reg.int.cold)$coef[1,1] #intercept
  int.results[i,11]<-summary(reg.int.cold)$coef[2,4] #p(slope)
  int.results[i,12]<-summary(reg.int.cold)$adj.r.squared #r2
  int.results[i,13]<-as.numeric(AICc(reg.int.cold)) #AICc
  int.results[i,14]<-summary(reg.int.cold)$fstatistic[1] #F stats
  int.results[i,15]<-summary(reg.int.cold)$coef[3,4] #p(additive term)
  int.results[i,16]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)


  ## gdd0
  reg.linear.gdd <- lm(log(AAM)~average_gdd_0, data = sub)
  reg.int.gdd <- lm(log(AAM)~average_gdd_0*Condition, data = sub)

  # simple linear model
  linear.results[i,13]<-summary(reg.linear.gdd)$coef[2,1] #slope
  linear.results[i,14]<-summary(reg.linear.gdd)$coef[1,1] #intercept
  linear.results[i,15]<-summary(reg.linear.gdd)$coef[2,4] #p-value
  linear.results[i,16]<-summary(reg.linear.gdd)$adj.r.squared #r2
  linear.results[i,17]<-as.numeric(AICc(reg.linear.gdd)) #AICc
  linear.results[i,18]<-summary(reg.linear.gdd)$fstatistic[1] #F stats

  # interaction model
  int.results[i,17]<-summary(reg.int.gdd)$coef[2,1] #slope
  int.results[i,18]<-summary(reg.int.gdd)$coef[1,1] #intercept
  int.results[i,19]<-summary(reg.int.gdd)$coef[2,4] #p(slope)
  int.results[i,20]<-summary(reg.int.gdd)$adj.r.squared #r2
  int.results[i,21]<-as.numeric(AICc(reg.int.gdd)) #AICc
  int.results[i,22]<-summary(reg.int.gdd)$fstatistic[1] #F stats
  int.results[i,23]<-summary(reg.int.gdd)$coef[3,4] #p(additive term)
  int.results[i,24]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)
}
```

Check the slopes and additive/interaction terms

Now we need to:

1. Check if the additive or interaction term is significant.
2. Check if the relationship is significant between age at maturity and temperature (significant slope).

```
## Additive/interaction term
# annual
mean(unique(int.results[,"p additive term.a"]))
```

```
## [1] 0.6277282
```

```r
table(int.results[,"p additive term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.a"]))
```

```
## [1] 0.8550577
```

```r
table(int.results[,"p interaction term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# cold
mean(unique(int.results[,"p additive term.c"]))
```

```
## [1] 0.7575554
```

```r
table(int.results[,"p additive term.c"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.c"]))
```

```
## [1] 0.8573869
```

```r
table(int.results[,"p interaction term.c"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# gdd
mean(unique(int.results[,"p additive term.g"]))
```

```
## [1] 0.4489923
```

```r
table(int.results[,"p additive term.g"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.g"]))
```

```
## [1] 0.8573869
```

```r
table(int.results[,"p interaction term.g"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
### Now let's look at the simple regression model:

## p-value for the slope (also is the p-value for the entire model)
```

```r
# annual
mean(unique(linear.results[,"p for slope.a"]))
```

```
## [1] 1.123214e-07
```

```r
table(linear.results[,"p for slope.a"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# cold
mean(unique(linear.results[,"p for slope.c"]))
```

```
## [1] 8.397821e-08
```

```r
table(linear.results[,"p for slope.c"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# gdd
mean(unique(linear.results[,"p for slope.g"]))
```

```
## [1] 3.814636e-08
```

```r
table(linear.results[,"p for slope.g"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
## F statistics
mean(unique(linear.results[,"F stats.a"]))
```

```
## [1] 51.62361
```

```r
mean(unique(linear.results[,"F stats.c"]))
```

```
## [1] 52.67616
```

```r
mean(unique(linear.results[,"F stats.g"]))
```

```
## [1] 57.21974
```

- No significant additive or interaction term in the multiple linear model.
- Significant slope.

Compare the R2

```r
# annual
r2annual <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,5])),
         mean(unique(int.results[,5])))
)
kable(r2annual)
```

Model	R2
Simple linear	9.814516
Interaction	14.483718

```
# cold
r2cold <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,11])),
         mean(unique(int.results[,11])))
)
kable(r2cold)
```

Model	R2
Simple linear	9.4038941
Interaction	0.0000338

```
# gdd
r2gdd <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,17])),
         mean(unique(int.results[,17])))
)
kable(r2gdd)
```

Model	R2
Simple linear	7.7587726
Interaction	-0.0001413

Compare among three temperatures (SIMPLE MODEL ONLY)

```
## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  R2 = c(mean(unique(linear.results[,"r2.a"])),
         mean(unique(linear.results[,"r2.c"])),
         mean(unique(linear.results[,"r2.g"])))
)
kable(r2)
```

Temperature	R2
Annual	0.6345688
Cold	0.6397366
GDD0	0.6589762

```
## Check the AICc scores and akaike weights in ONLY LINEAR MODEL
weight.matrix <- matrix(NA, 1000, 3)
count <- numeric(0)
```

```
for (i in 1:1000) {
  # Create a list of the aic values of the current iteration
  aic_value <- c(linear.results[i,5], linear.results[i,11],
                 linear.results[i,17])

  ## check the akaike weights
  weight <- compute_akaike_weights(aic_value)
  weight.matrix[i,c(1,2,3)] <- round(weight[c(1,2,3)],3)

  ## check the AICc scores
  indexing <- compare_aic_scores(aic_value)
  if (indexing != -999) {
    count <- c(count, indexing)
  }
}

summary(weight.matrix)
```

```
##        V1               V2               V3
##  Min.   :0.1430   Min.   :0.1410   Min.   :0.4120
##  1st Qu.:0.1570   1st Qu.:0.2130   1st Qu.:0.5120
##  Median :0.1810   Median :0.2560   Median :0.5550
##  Mean   :0.1994   Mean   :0.2499   Mean   :0.5505
##  3rd Qu.:0.2330   3rd Qu.:0.3150   3rd Qu.:0.5950
##  Max.   :0.2660   Max.   :0.3750   Max.   :0.6810
```

```
table(count)
```

```
## count
##   3
## 248
```

- Cold temperature did not show any preference over Annual temperature. No significant differences.

Two conditions separated

```
## Separate the two conditions
grass.natural <- Grass.clean[Grass.clean$Condition == "natural",]
grass.artificial <- Grass.clean[Grass.clean$Condition == "artificial",]

# Check sp
table(grass.natural$Code_Str)
```

```
##
## A B E F G I J L M N O P
## 2 1 1 1 1 2 2 2 1 1 1 1
```

```
table(grass.artificial$Code_Str)
```

```
##
## AA AB AC AD AF AG AI AJ AK AL  R  S  T  U  V  W  Y  Z
##  1  2  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
# store the results
r2.natural <- matrix(NA, 1000, 3)
r2.artificial <- matrix(NA, 1000, 3)
```

```
p.natural <- matrix(NA, 1000, 3)
p.artificial <- matrix(NA, 1000, 3)

F.natural <- matrix(NA, 1000, 3)
F.artificial <- matrix(NA, 1000, 3)


## Separating environment for 1000 iterations
for(i in 1:1000){

  ## natural
  sub.n <- grass.natural %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.n)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.n)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.n)

  # r2
  r2.natural[i,1]<-summary(reg.annual)$adj.r.squared
  r2.natural[i,2]<-summary(reg.cold)$adj.r.squared
  r2.natural[i,3]<-summary(reg.gdd)$adj.r.squared

  # p
  p.natural[i,1]<-summary(reg.annual)$coef[2,4]
  p.natural[i,2]<-summary(reg.cold)$coef[2,4]
  p.natural[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
  F.natural[i,1] <- summary(reg.annual)$fstatistic[1]
  F.natural[i,2] <- summary(reg.cold)$fstatistic[1]
  F.natural[i,3] <- summary(reg.gdd)$fstatistic[1]


  ## artificial
  sub.a <- grass.artificial %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.a)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.a)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.a)

  # r2
  r2.artificial[i,1]<-summary(reg.annual)$adj.r.squared
  r2.artificial[i,2]<-summary(reg.cold)$adj.r.squared
  r2.artificial[i,3]<-summary(reg.gdd)$adj.r.squared

  # p
  p.artificial[i,1]<-summary(reg.annual)$coef[2,4]
  p.artificial[i,2]<-summary(reg.cold)$coef[2,4]
  p.artificial[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
```

```r
  F.artificial[i,1] <- summary(reg.annual)$fstatistic[1]
  F.artificial[i,2] <- summary(reg.cold)$fstatistic[1]
  F.artificial[i,3] <- summary(reg.gdd)$fstatistic[1]
}


## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.R2 = c(mean(unique(r2.natural[,1])),
                 mean(unique(r2.natural[,2])),
                 mean(unique(r2.natural[,3]))),
  Artificial.R2 = c(mean(unique(r2.artificial[,1])),
                    mean(unique(r2.artificial[,2])),
                    mean(unique(r2.artificial[,3])))
)
kable(r2)
```

Temperature	Natural.R2	Artificial.R2
Annual	0.6444860	0.5415039
Cold	0.6381564	0.5389077
GDD0	0.5792947	0.6225748

```r
sub.n <- grass.artificial %>% group_by(Code_Str) %>% sample_n(size=1)
reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.n)
summary(reg.annual)
```

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp, data = sub.n)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54506 -0.15588  0.03183  0.15277  0.51283
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.987439   0.143808  13.820 2.59e-10 ***
## AnnualTemp  -0.040344   0.008897  -4.535 0.000338 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3039 on 16 degrees of freedom
## Multiple R-squared:  0.5624, Adjusted R-squared:  0.535
## F-statistic: 20.56 on 1 and 16 DF,  p-value: 0.0003384
```

```r
## Compare the F stats
Fstats <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.F = c(mean(unique(F.natural[,1])),
                mean(unique(F.natural[,2])),
                mean(unique(F.natural[,3]))),
  Artificial.F = c(mean(unique(F.artificial[,1])),
```

```r
                    mean(unique(F.artificial[,2])),
                    mean(unique(F.artificial[,3])))
)
kable(Fstats)
```

Temperature	Natural.F	Artificial.F
Annual	22.79757	21.08509
Cold	21.58674	20.89783
GDD0	17.00048	29.05468

```r
## Check the p values
table(p.natural[,1] < 0.05) # natural, annual
```

```
##
## TRUE
## 1000
```

```r
table(p.natural[,2] < 0.05) # natural, cold
```

```
##
## TRUE
## 1000
```

```r
table(p.natural[,3] < 0.05) # natural, gdd
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,1] < 0.05) # artificial, annual
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,2] < 0.05) # artificial, cold
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,3] < 0.05) # artificial, gdd
```

```
##
## TRUE
## 1000
```

SECTION 4: Bighead and silver carp

Data cleaning and matrices for results

```r
Big.sil.clean <- Big.sil %>%
  filter(Condition %in% c("natural", "artificial"))

table(Big.sil.clean$Code)
```

```
##
##  A AC AD AF AG AI AL AP  B  C  D  E  H  J  M  N  O  S  Z
##  4  1  1  2  2  1  1  1  3  1  2  2  1  1  2  1  2  2  2
```

```
# Simple linear model - slope, intercept, p, r2, AICc, F stats
linear.results <- matrix(NA,1000,18)
colnames(linear.results) <- c("slope.a",
                                  "intercept.a",
                                  "p for slope.a",
                                  "r2.a",
                                  "AICc.a",
                                  "F stats.a",
                              "slope.c",
                                  "intercept.c",
                                  "p for slope.c",
                                  "r2.c",
                                  "AICc.c",
                                  "F stats.c",
                              "slope.g",
                                  "intercept.g",
                                  "p for slope.g",
                                  "r2.g",
                                  "AICc.g",
                                  "F stats.g")

# Interaction model - slope, intercept, p(slope), p(interaction), r2, AICc
int.results <- matrix(NA,1000,24)
colnames(int.results) <- c("slope.a",
                                  "intercept.a",
                                  "p for slope.a",
                                  "r2.a",
                                  "AICc.a",
                                  "F stats.a",
                                  "p additive term.a",
                                  "p interaction term.a",
                          "slope.c",
                                  "intercept.c",
                                  "p for slope.c",
                                  "r2.c",
                                  "AICc.c",
                                  "F stats.c",
                                  "p additive term.c",
                                  "p interaction term.c",
                          "slope.g",
                                  "intercept.g",
                                  "p for slope.g",
                                  "r2.g",
                                  "AICc.g",
                                  "F stats.g",
                                  "p additive term.g",
                                  "p interaction term.g")
```

- Stratified sub-sampling gives 13 artificial points and 10 natural points.

Define the models

```
# For 1000 iterations
for(i in 1:1000){
  sub <- Big.sil.clean %>% group_by(Code_Str) %>% sample_n(size=1)

  ## annual
  reg.linear.annual <- lm(log(AAM)~AnnualTemp, data = sub)
  reg.int.annual <- lm(log(AAM)~AnnualTemp*Condition, data = sub)

  # simple linear model
  linear.results[i,1]<-summary(reg.linear.annual)$coef[2,1] #slope
  linear.results[i,2]<-summary(reg.linear.annual)$coef[1,1] #intercept
  linear.results[i,3]<-summary(reg.linear.annual)$coef[2,4] #p-value
  linear.results[i,4]<-summary(reg.linear.annual)$adj.r.squared #r2
  linear.results[i,5]<-as.numeric(AICc(reg.linear.annual)) #AICc
  linear.results[i,6]<-summary(reg.linear.annual)$fstatistic[1] #F stats

  # interaction model
  int.results[i,1]<-summary(reg.int.annual)$coef[2,1] #slope
  int.results[i,2]<-summary(reg.int.annual)$coef[1,1] #intercept
  int.results[i,3]<-summary(reg.int.annual)$coef[2,4] #p(slope)
  int.results[i,4]<-summary(reg.int.annual)$adj.r.squared #r2
  int.results[i,5]<-as.numeric(AICc(reg.int.annual)) #AICc
  int.results[i,6]<-summary(reg.int.annual)$fstatistic[1] #F stats
  int.results[i,7]<-summary(reg.int.annual)$coef[3,4] #p(additive term)
  int.results[i,8]<-summary(reg.int.annual)$coef[4,4] #p(interaction term)


  ## cold
  reg.linear.cold <- lm(log(AAM)~ColdTemp, data = sub)
  reg.int.cold <- lm(log(AAM)~ColdTemp*Condition, data = sub)

  # simple linear model
  linear.results[i,7]<-summary(reg.linear.cold)$coef[2,1] #slope
  linear.results[i,8]<-summary(reg.linear.cold)$coef[1,1] #intercept
  linear.results[i,9]<-summary(reg.linear.cold)$coef[2,4] #p-value
  linear.results[i,10]<-summary(reg.linear.cold)$adj.r.squared #r2
  linear.results[i,11]<-as.numeric(AICc(reg.linear.cold)) #AICc
  linear.results[i,12]<-summary(reg.linear.cold)$fstatistic[1] #F stats

  # interaction model
  int.results[i,9]<-summary(reg.int.cold)$coef[2,1] #slope
  int.results[i,10]<-summary(reg.int.cold)$coef[1,1] #intercept
  int.results[i,11]<-summary(reg.int.cold)$coef[2,4] #p(slope)
  int.results[i,12]<-summary(reg.int.cold)$adj.r.squared #r2
  int.results[i,13]<-as.numeric(AICc(reg.int.cold)) #AICc
  int.results[i,14]<-summary(reg.int.cold)$fstatistic[1] #F stats
  int.results[i,15]<-summary(reg.int.cold)$coef[3,4] #p(additive term)
  int.results[i,16]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)


  ## gdd0
  reg.linear.gdd <- lm(log(AAM)~average_gdd_0, data = sub)
```

```r
  reg.int.gdd <- lm(log(AAM)~average_gdd_0*Condition, data = sub)

  # simple linear model
  linear.results[i,13]<-summary(reg.linear.gdd)$coef[2,1] #slope
  linear.results[i,14]<-summary(reg.linear.gdd)$coef[1,1] #intercept
  linear.results[i,15]<-summary(reg.linear.gdd)$coef[2,4] #p-value
  linear.results[i,16]<-summary(reg.linear.gdd)$adj.r.squared #r2
  linear.results[i,17]<-as.numeric(AICc(reg.linear.gdd)) #AICc
  linear.results[i,18]<-summary(reg.linear.gdd)$fstatistic[1] #F stats

  # interaction model
  int.results[i,17]<-summary(reg.int.gdd)$coef[2,1] #slope
  int.results[i,18]<-summary(reg.int.gdd)$coef[1,1] #intercept
  int.results[i,19]<-summary(reg.int.gdd)$coef[2,4] #p(slope)
  int.results[i,20]<-summary(reg.int.gdd)$adj.r.squared #r2
  int.results[i,21]<-as.numeric(AICc(reg.int.gdd)) #AICc
  int.results[i,22]<-summary(reg.int.gdd)$fstatistic[1] #F stats
  int.results[i,23]<-summary(reg.int.gdd)$coef[3,4] #p(additive term)
  int.results[i,24]<-summary(reg.int.cold)$coef[4,4] #p(interaction term)
}
```

Check the slopes and additive/interaction terms

Now we need to:

1. Check if the additive or interaction term is significant.
2. Check if the relationship is significant between age at maturity and temperature (significant slope).

```r
## Additive/interaction term
# annual
mean(unique(int.results[,"p additive term.a"]))
```

```
## [1] 0.4336953
```

```r
table(int.results[,"p additive term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.a"]))
```

```
## [1] 0.9000645
```

```r
table(int.results[,"p interaction term.a"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# cold
mean(unique(int.results[,"p additive term.c"]))
```

```
## [1] 0.1756641
```

```r
table(int.results[,"p additive term.c"] < 0.05)
```

```
##
```

```
## FALSE  TRUE
##   993     7
```

```r
mean(unique(int.results[,"p interaction term.c"]))
```

```
## [1] 0.8675939
```

```r
table(int.results[,"p interaction term.c"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
# gdd
mean(unique(int.results[,"p additive term.g"]))
```

```
## [1] 0.8533478
```

```r
table(int.results[,"p additive term.g"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
mean(unique(int.results[,"p interaction term.g"]))
```

```
## [1] 0.8675939
```

```r
table(int.results[,"p interaction term.g"] < 0.05)
```

```
##
## FALSE
##  1000
```

```r
### Now let's look at the simple regression model:

## p-value for the slope (also is the p-value for the entire model)
# annual
mean(unique(linear.results[,"p for slope.a"]))
```

```
## [1] 0.001188979
```

```r
table(linear.results[,"p for slope.a"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# cold
mean(unique(linear.results[,"p for slope.c"]))
```

```
## [1] 0.008824907
```

```r
table(linear.results[,"p for slope.c"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
# gdd
mean(unique(linear.results[,"p for slope.g"]))
```

```
## [1] 0.0004937825
```

```r
table(linear.results[,"p for slope.g"] < 0.05)
```

```
##
## TRUE
## 1000
```

```r
## F statistics
mean(unique(linear.results[,"F stats.a"]))
```

```
## [1] 14.52885
```

```r
mean(unique(linear.results[,"F stats.c"]))
```

```
## [1] 8.615378
```

```r
mean(unique(linear.results[,"F stats.g"]))
```

```
## [1] 17.39038
```

- No significant additive or interaction term.
- Significant slope.

Compare the R2

```r
# annual
r2annual <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,5])),
         mean(unique(int.results[,5])))
)
kable(r2annual)
```

Model	R2
Simple linear	16.20863
Interaction	19.56529

```r
# cold
r2cold <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,11])),
         mean(unique(int.results[,11])))
)
kable(r2cold)
```

Model	R2
Simple linear	20.3864723
Interaction	0.0280717

```r
# gdd
r2gdd <- data.frame(
  Model = c("Simple linear", "Interaction"),
  R2 = c(mean(unique(linear.results[,17])),
```

```
         mean(unique(int.results[,17]))
)
kable(r2gdd)
```

Model	R2
Simple linear	14.4233171
Interaction	-0.0001327

Compare between annual and cold (SIMPLE MODEL ONLY)

```
## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  R2 = c(mean(unique(linear.results[,"r2.a"])),
         mean(unique(linear.results[,"r2.c"])),
         mean(unique(linear.results[,"r2.g"])))
)
kable(r2)
```

Temperature	R2
Annual	0.3793263
Cold	0.2559958
GDD0	0.4257729

```
## Check the AICc scores and akaike weights in ONLY LINEAR MODEL
weight.matrix <- matrix(NA, 1000, 3)
count <- numeric(0)

for (i in 1:1000) {
  # Create a list of the aic values of the current iteration
  aic_value <- c(linear.results[i,5], linear.results[i,11],
                 linear.results[i,17])

  ## check the akaike weights
  weight <- compute_akaike_weights(aic_value)
  weight.matrix[i,c(1,2,3)] <- round(weight[c(1,2,3)],3)

  ## check the AICc scores
  indexing <- compare_aic_scores(aic_value)
  if (indexing != -999) {
    count <- c(count, indexing)
  }
}

summary(weight.matrix)
```

```
##        V1               V2                 V3
##  Min.   :0.2070   Min.   :0.02300   Min.   :0.590
##  1st Qu.:0.2560   1st Qu.:0.03100   1st Qu.:0.655
##  Median :0.2810   Median :0.03500   Median :0.685
##  Mean   :0.2826   Mean   :0.03536   Mean   :0.682
```

```
##   3rd Qu.:0.3070   3rd Qu.:0.04000   3rd Qu.:0.709
##   Max.   :0.3740   Max.   :0.05000   Max.   :0.760
```

```r
table(count)
```

```
## count
##   3
## 274
```

- For bs, using annual temperature is always better than using the cold temperature (difference in AIC > 2) for all models. This was also explained by lower R^2 values for the cold temperature models.

Two conditions separated

```r
## Separate the two conditions
bs.natural <- Big.sil.clean[Big.sil.clean$Condition == "natural",]
bs.artificial <- Big.sil.clean[Big.sil.clean$Condition == "artificial",]

# Check sp
table(bs.natural$Code_Str)
```

```
##
## A C D F H I J K M Q
## 2 1 1 1 1 1 2 1 1 1
```

```r
table(bs.artificial$Code_Str)
```

```
##
## AA AB AC AE AG AH AI AK AL  R  U  V  X
##  2  1  1  1  2  1  2  2  1  2  2  2  1
```

```r
# store the results
r2.natural <- matrix(NA, 1000, 3)
r2.artificial <- matrix(NA, 1000, 3)

p.natural <- matrix(NA, 1000, 3)
p.artificial <- matrix(NA, 1000, 3)

F.natural <- matrix(NA, 1000, 3)
F.artificial <- matrix(NA, 1000, 3)


## Separating environment for 1000 iterations
for(i in 1:1000){

  ## natural
  sub.n <- bs.natural %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.n)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.n)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.n)

  # r2
  r2.natural[i,1]<-summary(reg.annual)$adj.r.squared
  r2.natural[i,2]<-summary(reg.cold)$adj.r.squared
  r2.natural[i,3]<-summary(reg.gdd)$adj.r.squared
```

```
  # p
  p.natural[i,1]<-summary(reg.annual)$coef[2,4]
  p.natural[i,2]<-summary(reg.cold)$coef[2,4]
  p.natural[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
  F.natural[i,1] <- summary(reg.annual)$fstatistic[1]
  F.natural[i,2] <- summary(reg.cold)$fstatistic[1]
  F.natural[i,3] <- summary(reg.gdd)$fstatistic[1]


  ## artificial
  sub.a <- bs.artificial %>% group_by(Code_Str) %>% sample_n(size=1)

  # models
  reg.annual <- lm(log(AAM)~AnnualTemp, data = sub.a)
  reg.cold <- lm(log(AAM)~ColdTemp, data = sub.a)
  reg.gdd <- lm(log(AAM)~average_gdd_0, data = sub.a)

  # r2
  r2.artificial[i,1]<-summary(reg.annual)$adj.r.squared
  r2.artificial[i,2]<-summary(reg.cold)$adj.r.squared
  r2.artificial[i,3]<-summary(reg.gdd)$adj.r.squared

  # p
  p.artificial[i,1]<-summary(reg.annual)$coef[2,4]
  p.artificial[i,2]<-summary(reg.cold)$coef[2,4]
  p.artificial[i,3]<-summary(reg.gdd)$coef[2,4]

  # F
  F.artificial[i,1] <- summary(reg.annual)$fstatistic[1]
  F.artificial[i,2] <- summary(reg.cold)$fstatistic[1]
  F.artificial[i,3] <- summary(reg.gdd)$fstatistic[1]
}


## Compare the R2 for three temperatures
r2 <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.R2 = c(mean(unique(r2.natural[,1])),
                 mean(unique(r2.natural[,2])),
                 mean(unique(r2.natural[,3]))),
  Artificial.R2 = c(mean(unique(r2.artificial[,1])),
                    mean(unique(r2.artificial[,2])),
                    mean(unique(r2.artificial[,3])))
)
kable(r2)
```

Temperature	Natural.R2	Artificial.R2
Annual	0.2400272	0.5282247
Cold	0.1392577	0.3646856
GDD0	0.2814281	0.5911505

```r
## Compare the F stats
Fstats <- data.frame(
  Temperature = c("Annual", "Cold", "GDD0"),
  Natural.F = c(mean(unique(F.natural[,1])),
                mean(unique(F.natural[,2])),
                mean(unique(F.natural[,3]))),
  Artificial.F = c(mean(unique(F.artificial[,1])),
                   mean(unique(F.artificial[,2])),
                   mean(unique(F.artificial[,3])))
)
kable(Fstats)
```

Temperature	Natural.F	Artificial.F
Annual	3.845641	15.003771
Cold	2.457653	8.168436
GDD0	4.524979	18.963667

```r
## Check the p values
table(p.natural[,1] < 0.05) # natural, annual
```

```
##
## FALSE
##  1000
```

```r
table(p.natural[,2] < 0.05) # natural, cold
```

```
##
## FALSE
##  1000
```

```r
table(p.natural[,3] < 0.05) # natural, gdd
```

```
##
## FALSE
##  1000
```

```r
table(p.artificial[,1] < 0.05) # artificial, annual
```

```
##
## TRUE
## 1000
```

```r
table(p.artificial[,2] < 0.05) # artificial, cold
```

```
##
## FALSE  TRUE
##    49   951
```

```r
table(p.artificial[,3] < 0.05) # artificial, gdd
```

```
##
## TRUE
## 1000
```

- For bighead and silver carp, there were fewer data points (32 datapoints in total), but more subsample sets (10 sets of subsamples. This gave us 19 data points after subsampling with a much larger variation (due to a larger number of combinations). At extremes, we would have 13 artificial and 6 natural (if all subseting choose artificial); or 10 natural and 9 artificial (if all subsetting choose natural).

- So we use stratified sub-sampling to reduce this effect.

Concluding points

1. Black carp: Using cold temperature have a better fit (higher R2). No preference over the four types of models. So we chose the simple linear model (Akaike weight = 51%).
2. Black carp: When separate the two conditions, we see a large increase in the R2 for the natural condition. The artificial condition alone did not have a significant relationship between log AAM and temperature.
3. Asian carp: The simple linear model is preferred (87% for annual, and 70% for cold). For grass carp, the simple linear model is preferred only for using the cold temperature (100%). For bs carp, there is no preference among the four models.
4. Asian carp: Using annual temperature is preferred (65% of times when annual temperature is the preferred). There is no preferrence for grass carp; but strong preference for bs carp using annual temperature.
5. Stratified sub-sampling reduced the large variation in AIC values in the bighead and silver carp combined dataset.