Urosalpinx growth rate

Andrew Villeneuve

6/14/2021

# The Study - read first

For this part of the study, we measured growth rates of juvenile Urosalpinx snails. We measured snail length using ImageJ within two days of hatching, and grew them in tea strainers separated by population for 24 days in a common garden experiment. Nine replicates per population were distributed in six temperatures, with three groups of three subreplicates in each temperature/population treatment. Snails were given food ad libitum. The end length was recorded using digital calipers after 24 days, and we subtracted the starting length from the beginning length to get the growth rate (over 24 days; see data exploration for reasoning). From this growth data, we can create thermal performance curves (TPCs) for each population to show the growth response of Urosaplinx populations to a range of laboratory temperatures.

The analysis of these growth curves requires two steps:

1. the creation of models that describe the curved TPC shape using the rTPC package, and
2. the extraction of breakpoints (thermal optima = x breakpoint, maximal trait performance = y optima) for each population’s TPC and modeling which environmental factors best describe any patterns in these optima across populations.

These two steps are presented, separately, in the Data Analysis section. The organization of these analyses thusly:

* TPC fitting - I create, analyze, and plot nonlinear regressions using rezende model in rTPC for each population
* Breakpoint analysis - I extract the thermal optima (x brkpt) and maximal trait performance (y brkpt) of each population’s fitted TPC.

# Metadata

* code
  + Unique code for each indiviudal snail, corresponding to population, temperature treatment. First digit = temperature (1=16,2=20,3=24,4=26, 5=28, 6=30), second digit = site (1=Willapa, 2=Humboldt, 3=Great Bay, 4=Woods Hole, 5=Oyster, 6=Beaufort, 7=Folly Beach, 8=Skidaway), third digit = tupperware bin number (1-3), fourth digit = snail replicate (1-3)
* pop
  + Source population of each snail. See data table below for list of site abbreivations with site.
* temp
  + Common garden temperature the snails were raised in for 24 days. Degrees C
* hatch
  + hatch date of each snail from it’s egg case (m/dd/yyyy)
* exp.date
  + Date on which hatchling snails were placed in the common garden experiment. Not more then 2 days from the hatch date. (m/dd/yyyy)
* grow.date
  + End date where growth measurements were taken. 24 days after exp.date, therefore no more then 26 days post hatch (m/dd/yyyy)
* alive
  + Tracks if snails survived the experient. m marks missing, n marks no, y marks yes
* rem.oysters
  + Was there a surplus of food at the end of the experiment? n marks no, y marks yes
* cal.length.start
  + caliper length of hatchlings upon entering the experment. We took photos of snails before entering snails into the experiment, and then used ImageJ to extract snail sizes. Size in mm
* cal.length.end
  + caliper length of hatchling at the end of the experiemnt. We took caliper measurements of the snails, as well as verifying the measurements using a subset of photographs in ImageJ. Size in mm
* wt
  + End weight of snail. Note that no initial starting weight was recorded. Weight in g.
* ran.out
  + Did the snail ever run out of food during the consumption experiment? 1 for yes, 0 for no
* bin
  + Bin number, controls subreplication. The third digit of the code.
* oce
  + Ocean (Atlantic or Pacific)

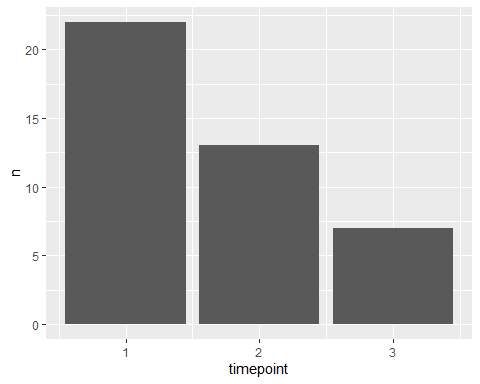
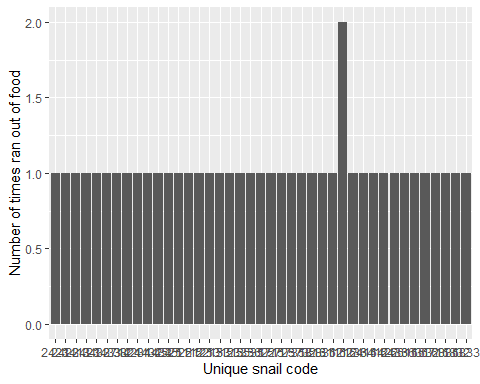
## site.abbreviation site  
## 1 gb Great Bay, NH  
## 2 wh Woods Hole, MA  
## 3 oy Oyster, VA  
## 4 bf Beaufort, NC  
## 5 fb Folly Beach, SC  
## 6 gcsk Skidaway, GA  
## 7 nah1516 Willapa, WA  
## 8 hmi2 Humboldt, CA

# Data Setup

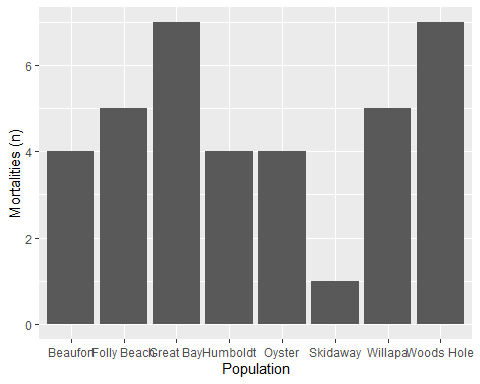
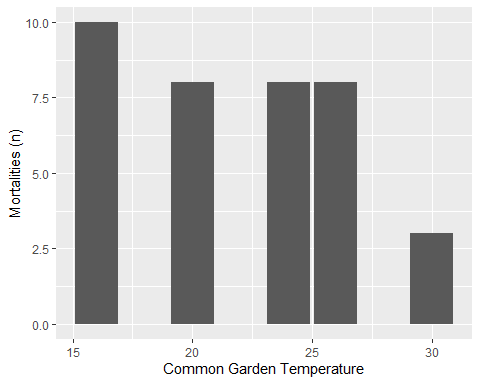
We had to clean parts of the data to prepare it for analysis. The first issue we had to resolve was the extreme weight and caliper length outliers, which was clearly due to a misplaced decimal or missing 0 during data entry (these data points were off by a factor of 10 from “correct” data). In the silenced part of the markdown, this had to be corrected 5/432 times.

The second issue was that some snails ran out of food during the growth experiment, which could jeopardize our assumption of unlimited growth while in the common garden experiment. We checked snail consumption three times over the course of the experiment, both to ensure snails had food but also to record which ones ran out of food. The vast majority (391/432) of snails never ran out of food. 40/432 ran out of food once, while 1/432 ran out of food twice. The plot below shows this breakdown, with snails that never ran out food removed. For the purposes of this experiment, we decided to include snails that missed a single meal during the entirety of the experiment, but removed the single case in which a snail ran out of food twice. When we remove these snails that ran out of food once, we get the same results (in terms of models and significance). Further evidence it’s ok to include these.

Further, the second plot below shows that most snails consumed all their food at timepoint 1, followed by timepoint 2 and finally timepoint 3 (seven snails ran out at t3)



The third issue was that some snails died, or went missing, over the course of the experiment. 37 snails died, and 1 went missing. We removed these snails from consideration, as we could not get a final growth measurement. The two plots below show the distribution of dead snails across site and temperature. Intriguingly, snails tended to died at lower temperatures, especially at 16C. We will see later on this was the temperature of lowest growth as well.



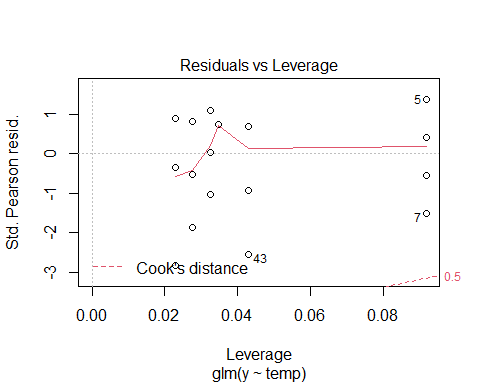
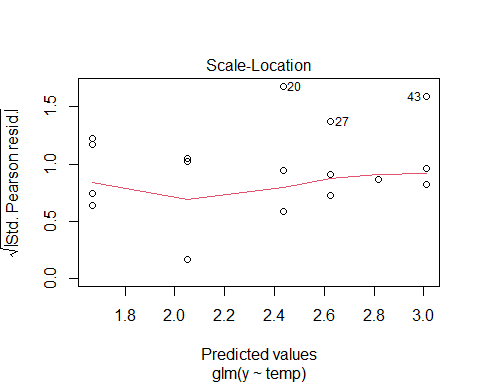
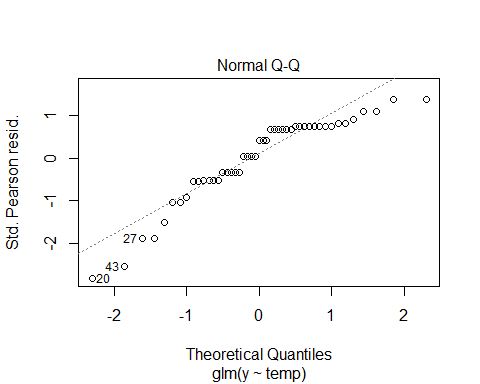
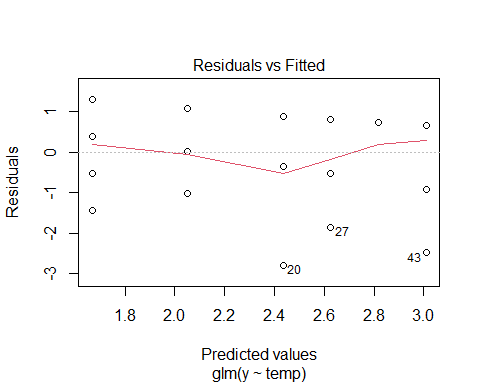
In, between removing snails that ran out of food twice (n=1), died (n=37), or went missing (n=1), there were 393 snails whose growth rates we kept.

So, there looks like there could be some significant patterns in survival between populations and common garden temperatures. Do we see that? Let’s see at just a population and temperature level.

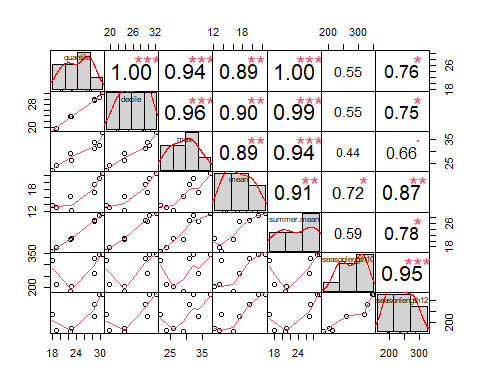
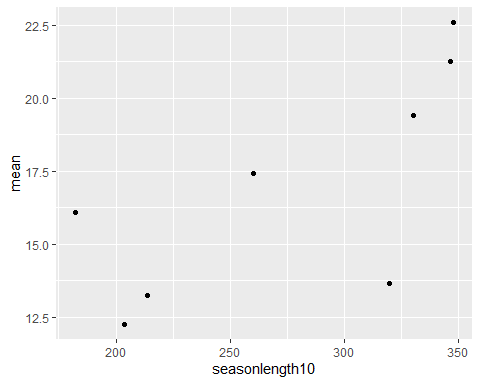
## Response 1 :  
## Df Sum Sq Mean Sq F value Pr(>F)   
## site 7 4.5390 0.6484 1.0379 0.421531   
## temp 1 5.5919 5.5919 8.9503 0.004852 \*\*  
## Residuals 38 23.7414 0.6248   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Response 2 :  
## Df Sum Sq Mean Sq F value Pr(>F)   
## site 7 4.5390 0.6484 1.0379 0.421531   
## temp 1 5.5919 5.5919 8.9503 0.004852 \*\*  
## Residuals 38 23.7414 0.6248   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## 1 observation deleted due to missingness

## 1   
## 0.9530274

##   
## Call:  
## glm(formula = y ~ site + temp, family = binomial, data = survival)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.3503 -0.4526 0.2808 0.9318 1.6268   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.74789 1.27194 1.374 0.16938   
## siteFolly Beach -1.89668 1.12006 -1.693 0.09039 .   
## siteGreat Bay -2.09304 1.09190 -1.917 0.05525 .   
## siteHumboldt -1.70598 1.11729 -1.527 0.12679   
## siteOyster -1.45814 1.13924 -1.280 0.20057   
## siteSkidaway -2.09304 1.09190 -1.917 0.05525 .   
## siteWillapa -1.45814 1.13924 -1.280 0.20057   
## siteWoods Hole -1.45814 1.13924 -1.280 0.20057   
## temp 0.09706 0.03541 2.741 0.00613 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 54.562 on 46 degrees of freedom  
## Residual deviance: 39.524 on 38 degrees of freedom  
## (1 observation deleted due to missingness)  
## AIC: 111.33  
##   
## Number of Fisher Scoring iterations: 5



##   
## Call:  
## lm(formula = seasonlength10 ~ mean, data = q, family = "gaussian")  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -81.489 -15.069 -5.354 17.674 86.874   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 55.864 87.150 0.641 0.545   
## mean 12.923 5.016 2.576 0.042 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 51.17 on 6 degrees of freedom  
## Multiple R-squared: 0.5252, Adjusted R-squared: 0.4461   
## F-statistic: 6.637 on 1 and 6 DF, p-value: 0.04198



Temperature does have an effect, with increasing survivorship with increasing common garden temperature. As for sites, there may be some differences, but we can’t tell this way. We will have to use environmental data as a proxy.

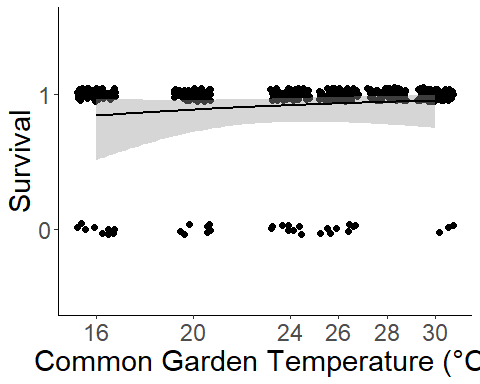
## Registered S3 methods overwritten by 'car':  
## method from  
## influence.merMod lme4  
## cooks.distance.influence.merMod lme4  
## dfbeta.influence.merMod lme4  
## dfbetas.influence.merMod lme4

## Analysis of Deviance Table (Type II tests)  
##   
## Response: y  
## LR Chisq Df Pr(>Chisq)   
## temp 7.6081 1 0.005811 \*\*  
## site 7.5162 7 0.377181   
## temp:site 3.4661 7 0.838805   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Family: binomial ( logit )  
## Formula: y ~ (temp \* site)  
## Data: survival  
##   
## AIC BIC logLik deviance df.resid   
## 121.9 151.5 -44.9 89.9 31   
##   
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 3.970e+00 5.187e+00 0.765 0.444  
## temp 3.206e-06 2.120e-01 0.000 1.000  
## siteFolly Beach -7.054e+00 5.694e+00 -1.239 0.215  
## siteGreat Bay -2.741e+00 5.562e+00 -0.493 0.622  
## siteHumboldt -2.583e+00 5.671e+00 -0.456 0.649  
## siteOyster -3.046e+00 5.739e+00 -0.531 0.596  
## siteSkidaway -5.290e+00 5.532e+00 -0.956 0.339  
## siteWillapa -4.057e+00 5.717e+00 -0.710 0.478  
## siteWoods Hole -3.552e+00 5.725e+00 -0.621 0.535  
## temp:siteFolly Beach 2.399e-01 2.429e-01 0.988 0.323  
## temp:siteGreat Bay 2.842e-02 2.279e-01 0.125 0.901  
## temp:siteHumboldt 3.788e-02 2.329e-01 0.163 0.871  
## temp:siteOyster 6.865e-02 2.370e-01 0.290 0.772  
## temp:siteSkidaway 1.414e-01 2.290e-01 0.617 0.537  
## temp:siteWillapa 1.142e-01 2.379e-01 0.480 0.631  
## temp:siteWoods Hole 9.125e-02 2.373e-01 0.384 0.701

##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## temp 3 107.40 0.00 0.28 0.28 -50.42  
## max 4 109.47 2.07 0.10 0.38 -50.26  
## mean 4 109.52 2.12 0.10 0.47 -50.28  
## seasonlength10 4 109.77 2.37 0.09 0.56 -50.41  
## seasonlength12 4 109.78 2.38 0.08 0.64 -50.41  
## lat 4 109.78 2.38 0.08 0.73 -50.42  
## t.mean 4 109.78 2.38 0.08 0.81 -50.42  
## s.mean 4 109.79 2.39 0.08 0.90 -50.42  
## q.mean 4 109.79 2.39 0.08 0.98 -50.42  
## null 2 112.64 5.24 0.02 1.00 -54.18  
## temp.i 16 139.99 32.59 0.00 1.00 -44.93

## Family: binomial ( logit )  
## Formula: y ~ temp + (1 | site)  
## Data: survival  
##   
## AIC BIC logLik deviance df.resid   
## 106.8 112.4 -50.4 100.8 44   
##   
## Random effects:  
##   
## Conditional model:  
## Groups Name Variance Std.Dev.   
## site (Intercept) 1.672e-09 4.089e-05  
## Number of obs: 47, groups: site, 8  
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.13375 0.79639 0.168 0.86663   
## temp 0.09588 0.03513 2.729 0.00636 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1



There is no difference when we look at survival across site, both purely as a comparison between sites as a factor and when we code sites by environmental data. However, survival of juveniles increases with common garden temperature. A cool result!

## Environmental Data

We extracted temperature data from each site (completed a few chunks above in the code). With this data, we calculated different environmental predictors that might explain patterns in growth.

* “Latitude” \*The latitude of collection sites
* “Quantile” (°C)  
  \*The average SST of the upper 75th percentile of summer months (06/01 - 09/30)
* “Decile” (°C) \*The average SST of the upper 90th percentile of summer months (06/01 - 09/30)
* “max” \*The maximum SST value recorded during summer months (06/01 - 09/30)
* “summer mean” (°C) \*The mean SST of the site, calculated during the summer months (06/01 - 09/30)
* "seasonlength10/12 (days) \*The number of days where the average daily temperature exceeded a threshold.
* “spring” \*The mean temperature during the initial first month of spawning
* “spring2” \*The mean temperature during the entire spawning period.

# Data exploration

This data exploration is for linear models only, describing putative TPCs. Therefore, the section below is just for information purposes and is shown collapsed or is silenced- see R code for details.

## Do sites differ?

Do populations differ in their growth rate across the common garden experiemnt? Here, ANOVA tells us that growth between sites are significantly different. We are justified in pursuing population and temperature level differences.

anovasites<-(aov(cal.length~pop\*temp,data=growth.alive))  
summary(anovasites)

## Df Sum Sq Mean Sq F value Pr(>F)   
## pop 7 23.2 3.3 7.274 3.47e-08 \*\*\*  
## temp 1 352.0 352.0 771.549 < 2e-16 \*\*\*  
## pop:temp 7 10.8 1.5 3.385 0.00162 \*\*   
## Residuals 374 170.6 0.5   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

growth.alive$pop<-as.factor(growth.alive$pop)

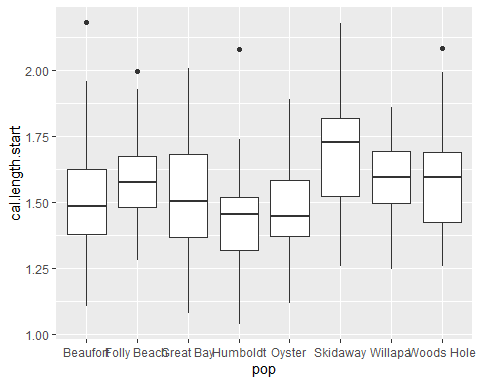
## Shell size by population

Should we use end caliper lengths, or do we need to subtract initial caliper length from end caliper length? In other words, do initial caliper lengths differ, requiring us to standardize our growth rate? Here, we find that populations do differ in initial growth. Tukey post-hoc comparisons (silenced, in code) further support this. Thus, we must standardize growth by creating a growth rate of Final size - initial size. The univariate boxplots below also show that while outliers do appear to be present, they can be attributed to population or temperature level differences.

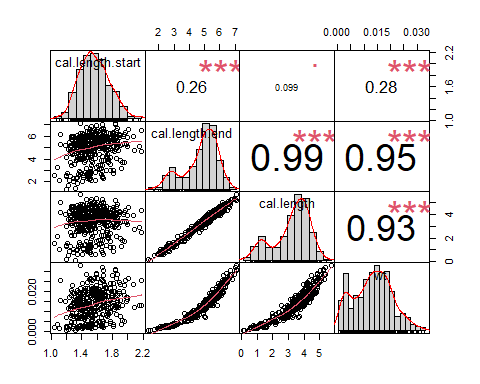
## Df Sum Sq Mean Sq F value Pr(>F)   
## pop 7 2.16 0.30856 8.539 9.92e-10 \*\*\*  
## Residuals 382 13.80 0.03614   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Beaufort Folly Beach Great Bay Humboldt Oyster Skidaway   
## 0.2137522 0.1710044 0.2129816 0.1801442 0.1783026 0.2043057   
## Willapa Woods Hole   
## 0.1574000 0.1940845

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = cal.length.start ~ pop, data = growth.alive)  
##   
## $pop  
## diff lwr upr p adj  
## Folly Beach-Beaufort 0.068810000 -0.048263994 0.185883994 0.6263723  
## Great Bay-Beaufort 0.014400426 -0.103307297 0.132108148 0.9999524  
## Humboldt-Beaufort -0.080368571 -0.196831464 0.036094322 0.4146657  
## Oyster-Beaufort -0.038020000 -0.153893204 0.077853204 0.9742735  
## Skidaway-Beaufort 0.177444615 0.062690986 0.292198245 0.0000926  
## Willapa-Beaufort 0.059205833 -0.057868161 0.176279828 0.7844796  
## Woods Hole-Beaufort 0.057581739 -0.060783647 0.175947125 0.8164141  
## Great Bay-Folly Beach -0.054409574 -0.173299560 0.064480411 0.8593668  
## Humboldt-Folly Beach -0.149178571 -0.266836237 -0.031520906 0.0032591  
## Oyster-Folly Beach -0.106830000 -0.223903994 0.010243994 0.1028212  
## Skidaway-Folly Beach 0.108634615 -0.007331398 0.224600629 0.0849532  
## Willapa-Folly Beach -0.009604167 -0.127866760 0.108658427 0.9999971  
## Woods Hole-Folly Beach -0.011228261 -0.130769406 0.108312884 0.9999922  
## Humboldt-Great Bay -0.094768997 -0.213057263 0.023519269 0.2243842  
## Oyster-Great Bay -0.052420426 -0.170128148 0.065287297 0.8757799  
## Skidaway-Great Bay 0.163044190 0.046438427 0.279649953 0.0006695  
## Willapa-Great Bay 0.044805408 -0.074084578 0.163695393 0.9454432  
## Woods Hole-Great Bay 0.043181314 -0.076980548 0.163343175 0.9575889  
## Oyster-Humboldt 0.042348571 -0.074114322 0.158811464 0.9548268  
## Skidaway-Humboldt 0.257813187 0.142464144 0.373162229 0.0000000  
## Willapa-Humboldt 0.139574405 0.021916739 0.257232070 0.0080966  
## Woods Hole-Humboldt 0.137950311 0.019007591 0.256893031 0.0107233  
## Skidaway-Oyster 0.215464615 0.100710986 0.330218245 0.0000006  
## Willapa-Oyster 0.097225833 -0.019848161 0.214299828 0.1855884  
## Woods Hole-Oyster 0.095601739 -0.022763647 0.213967125 0.2153223  
## Willapa-Skidaway -0.118238782 -0.234204796 -0.002272768 0.0420253  
## Woods Hole-Skidaway -0.119862876 -0.237132483 -0.002593269 0.0410856  
## Woods Hole-Willapa -0.001624094 -0.121165239 0.117917051 1.0000000



## Correlations

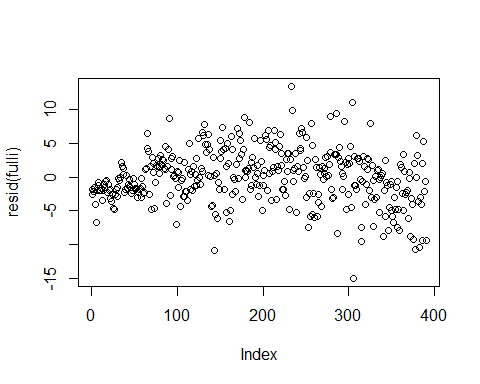


Here, we see that cal length and weight are both highly correlated. Thus, we will not include weight in any of our models.

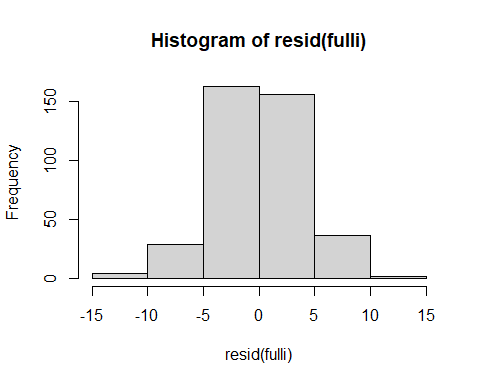
## Residuals of full model

Here, we investigate the residuals of the full model (lm(cal.length~pop \* temp \* bin,growth.alive)). This is to check if we need to transform our data, and if our assumptions of normality are warranted.

growth.alive<-na.omit(growth.alive)  
growth.alive.sqrt<-growth.alive  
growth.alive.sqrt$cal.length<-(growth.alive$cal.length)^2  
fulli<-(lm(cal.length~pop\*temp\*bin,growth.alive.sqrt))  
  
plot(resid(fulli))



hist(resid(fulli))



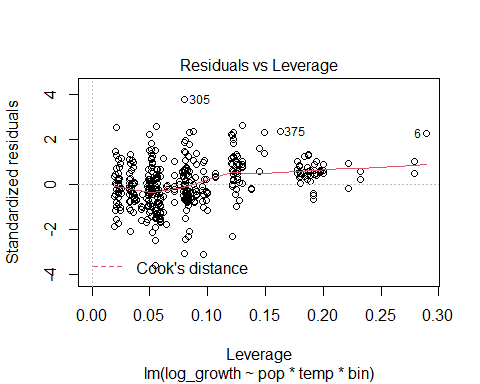
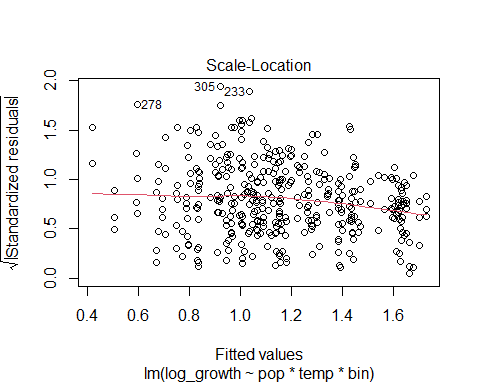
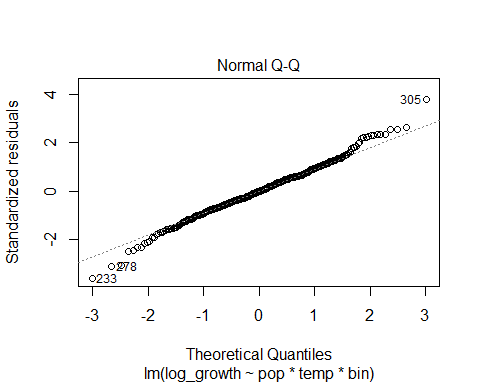
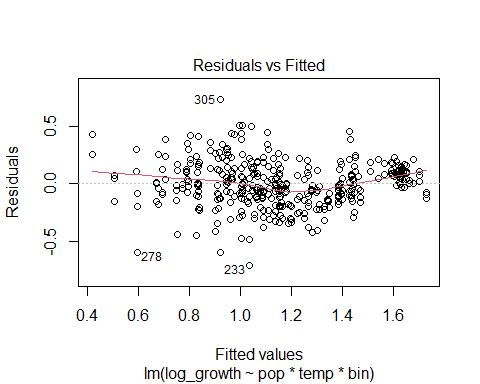
skewness(growth.alive.sqrt$cal.length)

## [1] -0.05442563

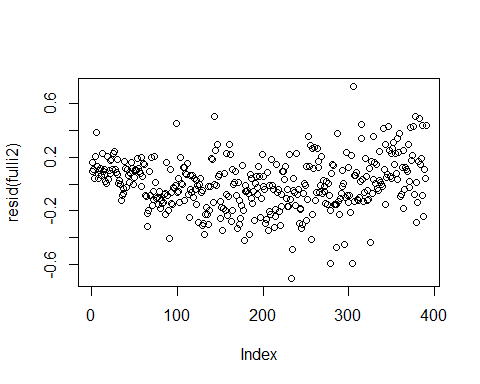
#negative skew  
growth.alive.sqrt<-tidyr::drop\_na(growth.alive.sqrt)  
growth.alive.sqrt<-na.omit(growth.alive.sqrt)

Things look ok, but maybe log-transforming growth will improve our skew and residual plots.

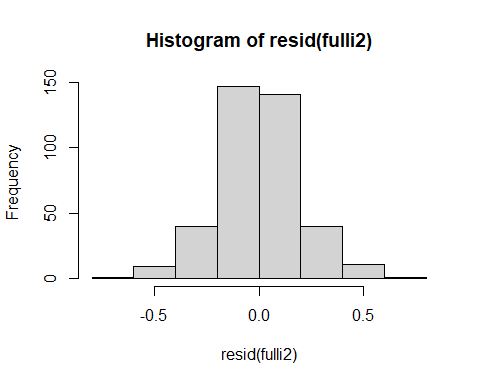
growth.alive$log\_growth<-log(max(growth.alive$cal.length+1)-growth.alive$cal.length)  
fulli2<-(lm(log\_growth~pop\*temp\*bin,growth.alive))  
  
plot(fulli2)



plot(resid(fulli2))



hist(resid(fulli2))



skewness(growth.alive$log\_growth)

## [1] 0.04950179

While the log transformation improves the skewness slightly, we will assume the distribution of residuals are normal and proceed with untransformed growth rate.

# Data Analysis

## Model predictors

We are going to create TPCs for each population by temperature, using both rTPC regression. What predictors should be used in these models?

##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## pop\*temp 17 820.01 0.00 0.3 0.30 -392.18  
## pop\*temp\*oce 17 820.01 0.00 0.3 0.59 -392.18  
## pop\*temp\*out 32 820.81 0.81 0.2 0.79 -375.45  
## pop\*temp\*oce\*out 32 820.81 0.81 0.2 0.98 -375.45  
## pop.temp 10 828.90 8.89 0.0 0.99 -404.16  
## pop.temp.oce 10 828.90 8.89 0.0 0.99 -404.16  
## pop.temp.bin 11 829.65 9.64 0.0 0.99 -403.47  
## pop.temp.oce.bin 11 829.65 9.64 0.0 1.00 -403.47  
## pop.temp.out 11 831.01 11.01 0.0 1.00 -404.16  
## pop.temp.oce.out 11 831.01 11.01 0.0 1.00 -404.16  
## fulla 12 831.77 11.76 0.0 1.00 -403.47  
## temp\*out 5 834.30 14.30 0.0 1.00 -412.07  
## bin\*out\*temp 9 838.27 18.27 0.0 1.00 -409.90  
## oce\*temp\*out 9 841.00 20.99 0.0 1.00 -411.26  
## pop\*temp\*bin 33 844.44 24.44 0.0 1.00 -386.07  
## pop\*temp\*oce\*bin 33 844.44 24.44 0.0 1.00 -386.07  
## temp\*oce\*bin\*out 17 851.09 31.09 0.0 1.00 -407.72  
## temp 3 853.64 33.63 0.0 1.00 -423.79  
## oce 3 853.64 33.63 0.0 1.00 -423.79  
## temp.bin 4 854.64 34.63 0.0 1.00 -423.27  
## temp.out 4 855.56 35.56 0.0 1.00 -423.73  
## temp.oce 4 855.68 35.68 0.0 1.00 -423.79  
## bin.out.temp 5 856.53 36.52 0.0 1.00 -423.18  
## temp\*bin 5 856.53 36.52 0.0 1.00 -423.19  
## oce.temp.bin 5 856.69 36.69 0.0 1.00 -423.27  
## temp\*oce 5 857.47 37.46 0.0 1.00 -423.66  
## oce.temp.out 5 857.61 37.61 0.0 1.00 -423.73  
## temp.oce.bin.out 6 858.59 38.58 0.0 1.00 -423.18  
## fulli 56 860.55 40.54 0.0 1.00 -364.69  
## oce\*temp\*bin 9 862.78 42.77 0.0 1.00 -422.15  
## pop.out 10 1237.12 417.12 0.0 1.00 -608.27  
## out 3 1240.95 420.94 0.0 1.00 -617.44  
## bin.out 4 1242.62 422.61 0.0 1.00 -617.26  
## oce.out 4 1242.79 422.78 0.0 1.00 -617.34  
## bin\*out 5 1244.04 424.03 0.0 1.00 -616.94  
## oce.bin.out 5 1244.46 424.45 0.0 1.00 -617.15  
## oce\*out 5 1244.83 424.82 0.0 1.00 -617.33  
## pop 9 1247.38 427.37 0.0 1.00 -614.45  
## pop.oce 9 1247.38 427.37 0.0 1.00 -614.45  
## pop\*oce 9 1247.38 427.37 0.0 1.00 -614.45  
## pop.bin 10 1248.75 428.74 0.0 1.00 -614.08  
## null 2 1249.56 429.55 0.0 1.00 -622.76  
## oce\*bin\*out 9 1250.54 430.54 0.0 1.00 -616.03  
## pop\*out 17 1250.68 430.68 0.0 1.00 -607.52  
## bin 3 1250.96 430.96 0.0 1.00 -622.45  
## oce.bin 4 1252.95 432.95 0.0 1.00 -622.42  
## oce\*bin 5 1253.80 433.79 0.0 1.00 -621.82  
## pop\*bin 17 1260.64 440.64 0.0 1.00 -612.50

Here, we see that a few models are well supported. We choose the interactive pop\*temp model only because 1) the additive model only tells us if populations are different at each temperature, while the interactive model also tells us the populations slopes with temperature and gives us a TPC 2) oce adds nothing to the models, so is removed. 3) because we will create models for growth in random bins for each population, we will be accounting for bin-level variance and do not need to explicitly model it.

##   
## Call:  
## lm(formula = cal.length ~ pop \* temp, data = growth.alive)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.66991 -0.40576 0.02389 0.45756 1.57275   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.77328 0.49386 -3.591 0.000374 \*\*\*  
## popFolly Beach 0.53984 0.70613 0.765 0.445045   
## popGreat Bay -1.02905 0.71996 -1.429 0.153745   
## popHumboldt 0.39919 0.69843 0.572 0.567972   
## popOyster -0.45481 0.69034 -0.659 0.510417   
## popSkidaway 0.63587 0.68474 0.929 0.353679   
## popWillapa 0.48179 0.72715 0.663 0.508016   
## popWoods Hole 1.45221 0.72167 2.012 0.044905 \*   
## temp 0.21233 0.02002 10.605 < 2e-16 \*\*\*  
## popFolly Beach:temp -0.03559 0.02878 -1.237 0.216949   
## popGreat Bay:temp 0.05824 0.02928 1.989 0.047417 \*   
## popHumboldt:temp -0.02477 0.02840 -0.872 0.383738   
## popOyster:temp 0.02420 0.02802 0.864 0.388399   
## popSkidaway:temp -0.03873 0.02790 -1.388 0.165985   
## popWillapa:temp -0.01411 0.02929 -0.482 0.630423   
## popWoods Hole:temp -0.06146 0.02907 -2.114 0.035146 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6754 on 374 degrees of freedom  
## Multiple R-squared: 0.6935, Adjusted R-squared: 0.6812   
## F-statistic: 56.41 on 15 and 374 DF, p-value: < 2.2e-16

## Family: gaussian ( identity )  
## Formula: cal.length ~ pop \* temp + (1 | cal.length.start)  
## Data: growth.alive  
##   
## AIC BIC logLik deviance df.resid   
## 820.3 891.7 -392.2 784.3 372   
##   
## Random effects:  
##   
## Conditional model:  
## Groups Name Variance Std.Dev.  
## cal.length.start (Intercept) 0.01063 0.1031   
## Residual 0.42688 0.6534   
## Number of obs: 390, groups: cal.length.start, 309  
##   
## Dispersion estimate for gaussian family (sigma^2): 0.427   
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.77336 0.48364 -3.667 0.000246 \*\*\*  
## popFolly Beach 0.53523 0.69148 0.774 0.438907   
## popGreat Bay -1.02770 0.70517 -1.457 0.145016   
## popHumboldt 0.37116 0.69707 0.532 0.594408   
## popOyster -0.46453 0.67808 -0.685 0.493298   
## popSkidaway 0.62943 0.67093 0.938 0.348175   
## popWillapa 0.48865 0.71276 0.686 0.492985   
## popWoods Hole 1.44315 0.70808 2.038 0.041538 \*   
## temp 0.21226 0.01961 10.822 < 2e-16 \*\*\*  
## popFolly Beach:temp -0.03536 0.02819 -1.254 0.209725   
## popGreat Bay:temp 0.05827 0.02867 2.032 0.042128 \*   
## popHumboldt:temp -0.02375 0.02824 -0.841 0.400365   
## popOyster:temp 0.02472 0.02758 0.896 0.370019   
## popSkidaway:temp -0.03843 0.02735 -1.405 0.160037   
## popWillapa:temp -0.01431 0.02870 -0.499 0.618059   
## popWoods Hole:temp -0.06110 0.02852 -2.142 0.032175 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## fixed 17 820.01 0.00 0.75 0.75 -392.18  
## random 18 822.16 2.16 0.25 1.00 -392.16

Based on this supported model structure, we will construct models that follow this formulation: growth~pop\*temp.

## Model fitting using rTPC

It’s hard to compare TPC against one another. One method we’ve settled on is the use of nonlinear regression models using the package rTPC to quantify the shape of the reaction as well as the thermal optima (x) and the maximal trait performance (y). Here, we used the segmented package to create single-optima broken stick regressions that also allow us to extract optimas.

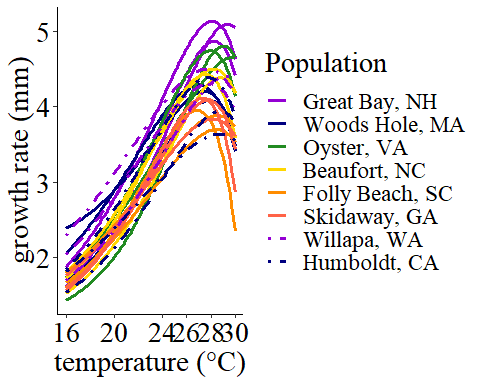
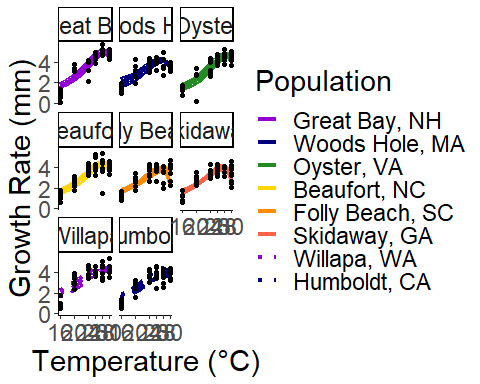
We used the rezende et al. 2019 model, based on a low AIC score, appropriate and few parameters, good CIs, and a model that performs well across many levels of biological organization. We used the following formulation: glm(shell length ~ temperature, population,family=gaussian), such that we modeled the response of shell lengths from a single population to temperature. We created three models for each population, one for each bin replicate. Each population had nine individuals per temperature randomly distributed between three bins. This allows us to extract three data points of MTP and three data points of Topt for every population instead of just one.

## rTPC model and predictions on binned data

### Now that we have created dataframes for binned population data, we can produce thermal performance curves for each pop-bin combination using rTPC. I referred heavily to the vignettes provided with the package.

## Plot the models

### Below are the fitted rezende models using rTPC. These are for visualization within the markdown. To see the plots which were used in the manuscript, as well as the process for creating bootstrapped confidence intervals, see the rmd file “TPC\_bootstrap.Rmd”.



## Breakpoint analysis

To be able to complete statistical analysis of the differences in TPC curves, I extracted the x and y componenets of each curve to give me the thermal optima and maximal trait performance, respectively. This extraction is silenced in code. Once we have extracted the thermal optima and maximal trait performance, we can move on to the relationship between environment and each breakpoint componenet.

### Maximum trait Performance (y axis)

The AIC table below tells us which environmental predictors best describe the relationship of maximal trait performance across populations.

##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## seasonlength10 4 9.76 0.00 0.32 0.32 0.17  
## seasonlength12 4 10.86 1.11 0.18 0.50 -0.38  
## spring 4 11.52 1.76 0.13 0.64 -0.71  
## nullo 3 11.87 2.11 0.11 0.75 -2.33  
## spring2 4 12.52 2.76 0.08 0.83 -1.21  
## lat 4 13.00 3.24 0.06 0.89 -1.45  
## s.mean 4 14.54 4.78 0.03 0.92 -2.22  
## q.mean 4 14.64 4.89 0.03 0.95 -2.27  
## t.mean 4 14.67 4.91 0.03 0.97 -2.28  
## omax 4 14.77 5.01 0.03 1.00 -2.33

Here, it appears that the season length as calculated at 10C is the best predictor.

### Linear model analysis of Maximal trait performance, season length = 10 C

## 2.5 % 97.5 %  
## cond((Int)) 1.718496494 7.6160512813  
## cond(seasonlength10) -0.007181533 -0.0008434195  
## cond(seasonlength12) -0.007046753 -0.0002594164  
## cond(spring) -0.005400111 0.2690080149

##   
## Call:  
## model.avg(object = a, b, c)  
##   
## Component model call:   
## glmmTMB(formula = brkpty ~ <3 unique rhs>, data = brkpts, family =   
## gaussian, ziformula = ~0, dispformula = ~1)  
##   
## Component models:   
## df logLik AICc delta weight  
## 1 4 0.17 9.76 0.00 0.50  
## 2 4 -0.38 10.86 1.11 0.29  
## 3 4 -0.71 11.52 1.76 0.21  
##   
## Term codes:   
## cond(seasonlength10) cond(seasonlength12) cond(spring)   
## 1 2 3   
##   
## Model-averaged coefficients:   
## (full average)   
## Estimate Std. Error Adjusted SE z value Pr(>|z|)   
## cond((Int)) 4.667274 1.486306 1.504506 3.102 0.00192 \*\*  
## cond(seasonlength10) -0.002017 0.002277 0.002311 0.873 0.38277   
## cond(seasonlength12) -0.001056 0.001873 0.001900 0.556 0.57828   
## cond(spring) 0.027453 0.061367 0.062333 0.440 0.65963   
##   
## (conditional average)   
## Estimate Std. Error Adjusted SE z value Pr(>|z|)   
## cond((Int)) 4.667274 1.486306 1.504506 3.102 0.00192 \*\*  
## cond(seasonlength10) -0.004012 0.001519 0.001617 2.482 0.01308 \*   
## cond(seasonlength12) -0.003653 0.001627 0.001731 2.110 0.03488 \*   
## cond(spring) 0.131804 0.065775 0.070003 1.883 0.05972 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## R2m R2c  
## [1,] 0.4207005 0.8246244

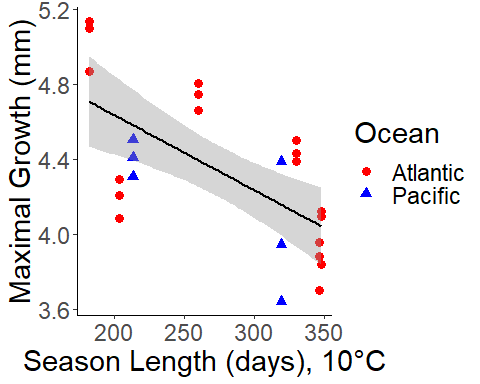
## R2m R2c  
## [1,] 0.3502089 0.8241001

## R2m R2c  
## [1,] 0.3033397 0.8237515

## [1] 0.6999585

## [1] 0.7312628

## `geom\_smooth()` using formula 'y ~ x'



But what about the thermal optima?

### Thermal Optima (Breakpoint X axis)

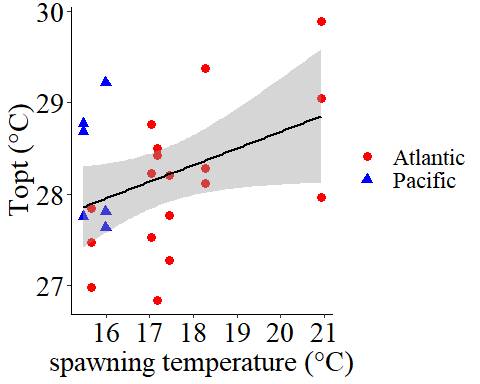
##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## spring 4 59.62 0.00 0.39 0.39 -24.76  
## nullo 3 60.96 1.34 0.20 0.58 -26.88  
## seasonlength12 4 63.40 3.78 0.06 0.64 -26.65  
## lat 4 63.48 3.87 0.06 0.70 -26.69  
## spring2 4 63.55 3.93 0.05 0.75 -26.72  
## seasonlength10 4 63.64 4.02 0.05 0.81 -26.77  
## s.mean 4 63.71 4.09 0.05 0.86 -26.80  
## q.mean 4 63.75 4.13 0.05 0.90 -26.82  
## t.mean 4 63.76 4.14 0.05 0.95 -26.83  
## omax 4 63.86 4.24 0.05 1.00 -26.88

## Family: gaussian ( identity )  
## Formula: brkptx ~ spring + (1 | pop)  
## Data: brkpts  
##   
## AIC BIC logLik deviance df.resid   
## 57.5 62.2 -24.8 49.5 20   
##   
## Random effects:  
##   
## Conditional model:  
## Groups Name Variance Std.Dev.   
## pop (Intercept) 5.110e-10 2.261e-05  
## Residual 4.608e-01 6.788e-01  
## Number of obs: 24, groups: pop, 8  
##   
## Dispersion estimate for gaussian family (sigma^2): 0.461   
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 25.03696 1.44812 17.289 <2e-16 \*\*\*  
## spring 0.18221 0.08356 2.181 0.0292 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## R2m R2c  
## [1,] 0.1713106 0.1713106

## 2.5 % 97.5 % Estimate  
## cond.(Intercept) 22.19870602 27.8752227 2.503696e+01  
## cond.spring 0.01843007 0.3459864 1.822083e-01  
## pop.cond.Std.Dev.(Intercept) 0.00000000 Inf 2.260548e-05

## `geom\_smooth()` using formula 'y ~ x'



##   
## Call:  
## lm(formula = brkptx ~ seasonlength10 + seasonlength12 + lat +   
## spring + spring2 + max + t.mean, data = brkpts)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.0827 -0.4714 0.0455 0.4653 1.0000   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 29.075452 17.834398 1.630 0.1226   
## seasonlength10 0.005561 0.016934 0.328 0.7469   
## seasonlength12 -0.011803 0.026281 -0.449 0.6594   
## lat -0.072862 0.272710 -0.267 0.7927   
## spring 0.424037 0.191335 2.216 0.0415 \*  
## spring2 -0.158863 0.212579 -0.747 0.4657   
## max 0.017214 0.142330 0.121 0.9052   
## t.mean -0.049802 0.214575 -0.232 0.8194   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7192 on 16 degrees of freedom  
## Multiple R-squared: 0.3754, Adjusted R-squared: 0.1021   
## F-statistic: 1.373 on 7 and 16 DF, p-value: 0.2818

## seasonlength10 seasonlength12 lat spring spring2   
## 55.017601 132.690859 82.701165 4.670334 14.658812   
## max t.mean   
## 22.089889 49.160981

## Model :  
## brkptx ~ seasonlength10 + seasonlength12 + lat + spring + spring2 +   
## max + t.mean

## [1] 1.417836

Here, mean temperature during initial spawning period is the best predictor.

#### Sensitivity analysis

One complication with the use of invasive species is that the invasive populations may be subject to founders effects. While we have good reason to believe that our Pacific populations are locally adapted (see manuscript), we should run a sensitivity analysis and see if our analysis holds up with Atlantic populations only.

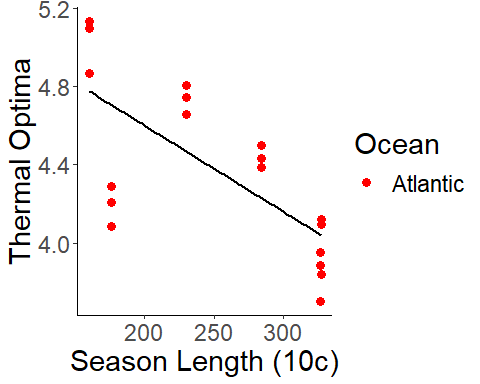
##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## seasonlength12 4 2.03 0.00 0.21 0.21 4.53  
## lat 4 2.44 0.41 0.17 0.38 4.32  
## seasonlength10 4 2.61 0.59 0.15 0.53 4.23  
## nullo 3 3.08 1.06 0.12 0.65 2.32  
## s.mean 4 4.03 2.01 0.08 0.73 3.52  
## spring 4 4.14 2.12 0.07 0.80 3.47  
## q.mean 4 4.37 2.35 0.06 0.86 3.35  
## t.mean 4 4.61 2.58 0.06 0.92 3.23  
## spring2 4 4.80 2.77 0.05 0.97 3.14  
## max 4 5.98 3.96 0.03 1.00 2.55

## Family: gaussian ( identity )  
## Formula: brkpty ~ seasonlength12 + (1 | pop)  
## Data: brkpts\_wh  
##   
## AIC BIC logLik deviance df.resid   
## -1.1 2.5 4.5 -9.1 14   
##   
## Random effects:  
##   
## Conditional model:  
## Groups Name Variance Std.Dev.  
## pop (Intercept) 0.07601 0.2757   
## Residual 0.01356 0.1164   
## Number of obs: 18, groups: pop, 6  
##   
## Dispersion estimate for gaussian family (sigma^2): 0.0136   
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 5.487521 0.449246 12.215 <2e-16 \*\*\*  
## seasonlength12 -0.004425 0.001731 -2.556 0.0106 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## [1] 0.8512634

## R2m R2c  
## [1,] 0.5089356 0.9256733

## `geom\_smooth()` using formula 'y ~ x'



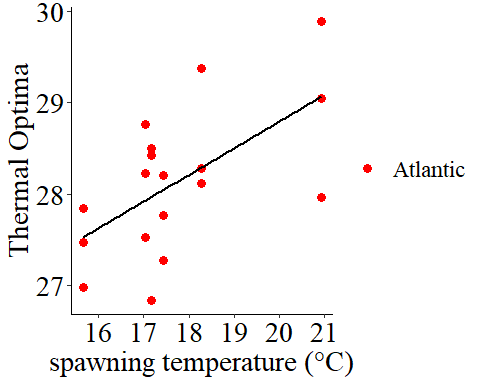
##   
## Model selection based on AICc:  
##   
## K AICc Delta\_AICc AICcWt Cum.Wt LL  
## spring 4 44.92 0.00 0.76 0.76 -16.92  
## nullo 3 49.38 4.46 0.08 0.84 -20.83  
## spring2 4 50.80 5.88 0.04 0.88 -19.86  
## seasonlength12 4 52.38 7.46 0.02 0.90 -20.65  
## seasonlength10 4 52.45 7.52 0.02 0.92 -20.68  
## lat 4 52.50 7.58 0.02 0.94 -20.71  
## omax 4 52.54 7.62 0.02 0.95 -20.73  
## t.mean 4 52.73 7.81 0.02 0.97 -20.83  
## q.mean 4 52.74 7.81 0.02 0.98 -20.83  
## s.mean 4 52.74 7.82 0.02 1.00 -20.83

## Family: gaussian ( identity )  
## Formula: brkptx ~ spring + (1 | pop)  
## Data: brkpts\_wh  
##   
## AIC BIC logLik deviance df.resid   
## 41.8 45.4 -16.9 33.8 14   
##   
## Random effects:  
##   
## Conditional model:  
## Groups Name Variance Std.Dev.   
## pop (Intercept) 1.014e-10 1.007e-05  
## Residual 3.838e-01 6.196e-01  
## Number of obs: 18, groups: pop, 6  
##   
## Dispersion estimate for gaussian family (sigma^2): 0.384   
##   
## Conditional model:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 22.96595 1.60707 14.291 < 2e-16 \*\*\*  
## spring 0.29120 0.09014 3.231 0.00124 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## [1] 2.515942e-10

## R2m R2c  
## [1,] 0.3804007 0.3804007

## `geom\_smooth()` using formula 'y ~ x'



With Pacific sites dropped, we see that the same environmental predictors are maintained, and that signficane remains! Therefore, we can present our results from the Pacific and the Atlantic after finishing this sensitivity analysis.