Selection pressures in *Syngnathus fuscus*

Coley Tosto

2024-01-05

```
#This is a cohesive list of all the libraries used in this document
library(ggplot2)
library(fBasics)
library(pwr)
```

```
#MomIDs and embryo counts for each section of the male's brood pouch
em_dat <- read.csv("data/EmbryoParentage_FU.csv")

#Metadata for males and females from the mesocosm experiments
fem_meso <- read.csv("data/all_fem_meso_fuscus.csv")
mal_meso <- read.csv("data/all_mal_meso_fuscus.csv")
```

Calculating the degree of sexual dimorphism

I noticed that there was a certain degree of sexual dimorphism in terms of an ornamentation that was present only in the females, but I also want to explore any size sexual dimorphism that may be present in this species.

I will be looking at total standard length (mm, measured from the tip of the snout to the tip of the caudal fin), snout-vent length (mm, measured from the tip of the snout to the urogenital opening), torso depth (mm), and snout length (mm). To look at these differnces I will be performing t-tests between males and females. First, I need to see if assumptions are met, i.e. variances are equal and data is normally distributed.

```
#Testing to see if the variances are equal
var.test(fem_meso$length, mal_meso$length) #equal
var.test(fem_meso$depth, mal_meso$depth) #not equal
var.test(fem_meso$svl, mal_meso$svl) #equal
var.test(fem_meso$snout_len, mal_meso$snout_len) #equal

#Testing for normal distribution
normalTest(fem_meso$length, method = "da") #normal
normalTest(fem_meso$depth, method = "da") #not normal
normalTest(fem_meso$svl, method = "da") #normal
normalTest(fem_meso$snout_len, method = "da") #normal
normalTest(mal_meso$length, method = "da") #normal
normalTest(mal_meso$depth, method = "da") #normal
normalTest(mal_meso$svl, method = "da") #normal
normalTest(mal_meso$snout_len, method = "da") #not normal
```

Variances are equal for standard length, snout-vent length, and snout length but not for torso depth. Data are normally distributed for all variables **except** for torso depth in females and snout length in males.

Based on this assumption testing a normal t-test will be used for standard length and snout-vent length and a Wilcoxon rank sum test will be used for snout length and torso depth.

```
t.test(fem_meso$length, mal_meso$length,
       var.equal = TRUE) #Sig. difference
t.test(fem_meso$svl, mal_meso$svl,
       var.equal = FALSE) #Sig. difference

wilcox.test(fem_meso$depth, mal_meso$depth) #Sig. difference
wilcox.test(fem_meso$snout_len,
            mal_meso$snout_len) #Sig. difference
```

There is a significant difference in all of the morphometrics between males and females, with females being larger in all of the categories.

It may be that females are just larger in general so that is inflating all of the other metrics, particularly depth. Are females actually deeper than males or are they just longer so the depth follows along suit? To try and address this I am going to adjust the depth by the total length of the fish and then re-run the t-test.

```
#Adjusting the depth by the length
fem_depth_adj <- fem_meso$depth/fem_meso$length
mal_depth_adj <- mal_meso$depth/mal_meso$length

#Re-checking assumptions
var.test(fem_depth_adj, mal_depth_adj) #not equal
normalTest(fem_depth_adj, method = "da") #not normal
normalTest(mal_depth_adj, method = "da") #normal

#Running the pairwise test
wilcox.test(fem_depth_adj, mal_depth_adj)

#Checking the power of the test to ensure we are detecting
#a true difference
d_mean <- abs(mean(fem_depth_adj, na.rm = TRUE) -
                mean(mal_depth_adj))
pool_sd <- sqrt((var(fem_depth_adj, na.rm = TRUE) +
                   var(mal_depth_adj))/ 2)
d <- d_mean/pool_sd

pwr.t.test(n = length(mal_depth_adj),
           d = d,
           sig.level = 0.05,
           type = 'one.sample',
           alternative = 'two.sided')
```

Even when we account for the length, depth is still significantly different between the sexes, with deeper individuals being females. This was checked with a power test to make sure that we are detecting a true difference and it returned a high power.

Calculating mating and reproductive success for individuals who mated

Syngnathus fuscus (Northern pipefish) were sampled from one cohesive seagrass beds in Chesapeake Bay in Cape Charles, Virgina. Sexually mature females (standard length $\geq$ 120mm) and pregnant males were collected and brought back to the University of Tampa for mesocosm experiments. In these mesocosms, 6 males and 6 females were housed together in a 140L tank for a period of 6-weeks and allowed to mate freely. Parentage analysis was done with all of the pregnant males from the trials to figure out how many times each male and female mated, and the number of eggs that were transferred. The results of that are here.

First I had to calculate the mating and reproductive success for each male and female who mated based on the assigned mom for each genotyped embryo.

```
#Row-by-Row analysis of parentage data by male brood pouch section

#Read in the data
#em_dat <- read.csv("~/EmbryoParentage.csv")

#For each row in the dataset(each section of the pouch) apply this function
mom_counts <- do.call(rbind,apply(em_dat, 1, function(one_section){

  #Save all of the momIDs into an object
  mom_ids<-c(one_section[grep("momID",names(one_section))])

  #Calculate the number of eggs that belongs to each potential mom based on
  #the proportions and total number of developed and undeveloped embryos
  mom_props<-c(as.numeric(one_section[grep("prop",names(one_section))]))
  mom_counts_dev<-mom_props*as.numeric(one_section["num_embryos_dev"])
  mom_counts_und<-mom_props*as.numeric(one_section["num_embryos_non_dev"])

  #Create a dataframe that contains the maleID, pouch section number and the
  #number of eggs that belongs to each momID
  this_section<-data.frame(
    maleID=one_section["maleID"],
    section_num=one_section["section_num"],
    mom_ids[which((mom_counts_dev + mom_counts_und) > 0)],
    mom_counts_dev[which((mom_counts_dev + mom_counts_und)>0)],
    mom_counts_und[which((mom_counts_dev + mom_counts_und)>0)]
  )

  #Rename the columns
  colnames(this_section)[3:5]<-c("momID","num_dev","num_und")

  return(this_section)

}))

#Calculate female fitness
fem_fitness<-do.call(rbind,by(mom_counts, mom_counts$momID,function(dat){

  mom_fitness<-data.frame(
    momID=unique(dat$momID),
    MatingSuccess=length(unique(dat$maleID)),
    NumDeveloped=round(sum(dat$num_dev)),
```

```
    NumUndeveloped=round(sum(dat$num_und))
  )
  return(mom_fitness)
}))

fem_fitness$totalEggs <- fem_fitness$NumDeveloped + fem_fitness$NumUndeveloped

#Calculate Male Fitness
mal_fitness<-do.call(rbind,by(mom_counts, mom_counts$maleID,function(dat){

  dad_fitness<-data.frame(
    maleID=unique(dat$maleID),
    MatingSuccess=length(unique(dat$momID)),
    NumDeveloped_Calc=round(sum(dat$num_dev)),
    NumUndeveloped_Calc=round(sum(dat$num_und))
  )
  return(dad_fitness)
}))

mal_fitness$totalEggs <- mal_fitness$NumDeveloped_Calc + mal_fitness$NumUndeveloped_Calc
```

After running the above R script we have generated two datasets, `mal_fitness` and `fem_fitness`. These datasets include information about the mating success (number of mates) and reproductive success (Number of embryos transferred). We can split reproductive success up further later if we want to from the total number of embryos transferred to the number of embryos developed and the number that were undeveloped.

I want to include all of the other metadata that I have for these individuals (traits, collection location, latency to pregnancy, etc.) as well as tack on all of the information for the individuals who did not mate. To do that I am going to need to merge the fitness datasets with `fem_meso` and `mal_meso`.

```
#Make a column in *_meso that contains the full fishID (i.e. FU1M3) to match the
#formatting in the fitness datasets (make sure they have the same name for merging purposes)
fem_meso$momID <- paste0("FU", fem_meso$trial_num, "F",
                         fem_meso$fishID)
mal_meso$maleID <- paste0("FU", mal_meso$trial_num, "M",
                          mal_meso$fishID)

#Merge the datasets based on the columns created above
fem_all <- merge(fem_meso, fem_fitness, by = "momID",
                 all.x = TRUE, all.y = TRUE)
mal_all <- merge(mal_meso, mal_fitness, by = "maleID",
                 all.x = TRUE, all.y = TRUE)
```

There are a few trials that I want to remove from the analysis including all trials where there were no successful matings (4, 7, 8, and 9).

I also want to replace the NAs that were automatically added to the columns from the fitness dataset (MatingSuccess, NumDeveloped, NumUndeveloped, totalEggs) with 0s and add a column to the female dataset that tells me whether or not the female mated (with 1 or 0).

```
#Subset the merged datasets to remove trials without successful matings
fem_succ <- subset(fem_all, !(trial_num %in% c(4, 7, 8, 9)))
mal_succ <- subset(mal_all, !(trial_num %in% c(4, 7, 8, 9)))
```

```r
#Replace NAs with 0s in the columns related to fitness
mal_succ[,15:18] <- sapply(mal_succ[,15:18],
                           function(x)
                             ifelse(is.na(x), 0, x))

fem_succ[,11:14] <- sapply(fem_succ[,11:14],
                           function(x)
                             ifelse(is.na(x), 0, x))

#Add a column for females to denote mated or unmated
fem_succ$mated <- ifelse(fem_succ$MatingSuccess > 0, 1, 0)
```

Summary statistics for successfully mated individuals

Males

Across all 12 trials and 74 total males, there were 22 males that mated at least one time and 1 of those males had two mates.

Looking across all males, including the ones that did not mate, this is what we find as the mean, sd, and se for the number of embryos transferred and how many of those developed versus didn't:

	mean	SD	SE	max	min
Number of Embryos	68.2432432	138.1056585	16.0544567	607	0
Developed Embryos	63.6216216	130.3914762	15.1577012	566	0
Undeveloped Embryos	4.6216216	13.9889415	1.6261814	84	0

These values will be influenced by the number of 0s coming from males who did not mate. So let's look at the same thing, but this time for only males who had at least one successful mating:

	mean	SD	SE	max	min
Number of Embryos	229.5454545	165.8693483	35.3634639	607	0
Developed Embryos	214	159.3379159	33.9709578	566	0
Undeveloped Embryos	15.5454545	22.4132768	4.7785267	84	0

We can see from the bottom table that even when we only include males who mated there is still a wide range in the brood size. I want to see what relationship there is between brood pouch size (in terms of both total area and length) and brood size (total number of embryos).

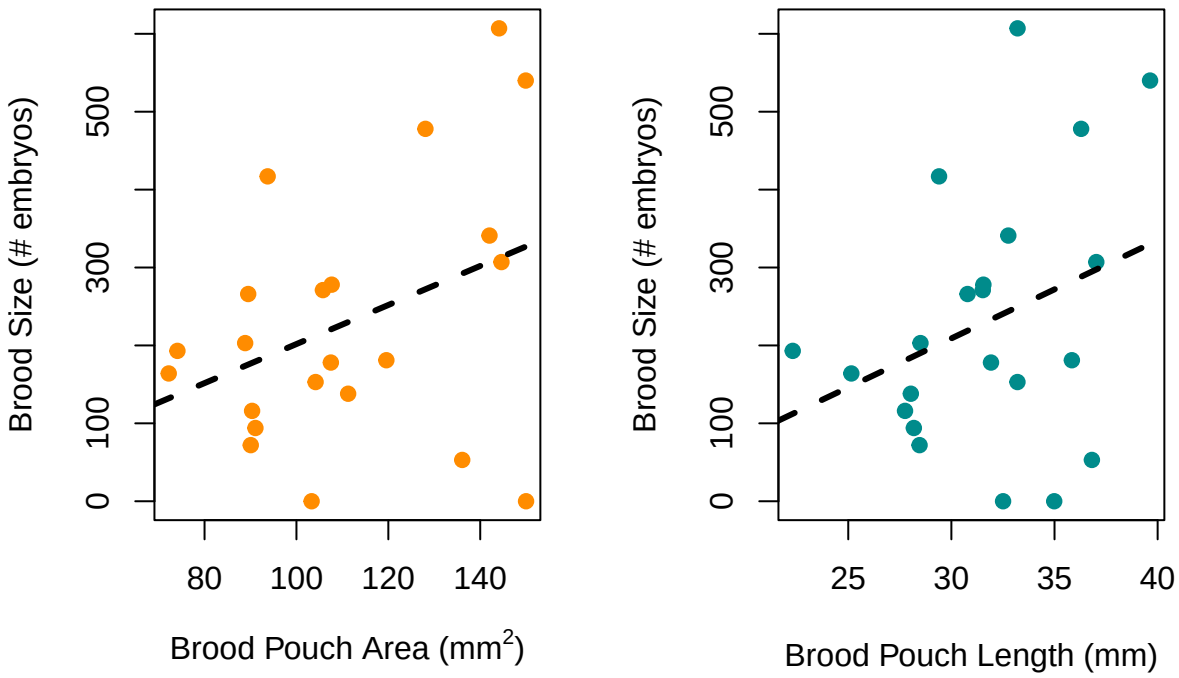


Figure 1: Scatterplot of the relationship between brood pouch size metrics and the number of embryos a male had.

There does appear to be some correlation happening here (Fig. 1). Let's run some correlations tests to see what they say.

```
##
##  Pearson's product-moment correlation
##
## data:  as.numeric(mated_mal$bp_area) and mated_mal$totalEggs
## t = 1.7941, df = 20, p-value = 0.08793
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.05845921  0.68621522
## sample estimates:
##       cor
## 0.3723259
```

```
##
##  Pearson's product-moment correlation
##
## data:  as.numeric(mated_mal$bp_length) and mated_mal$totalEggs
## t = 1.5076, df = 20, p-value = 0.1473
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  -0.1180647  0.6530941
## sample estimates:
##       cor
## 0.3194448
```

There is not a significant correlation between the number of eggs and size of the brood pouch when we look at brood pouch area OR brood pouch length.

Let's see if this changes if we just look at the overall size of the male

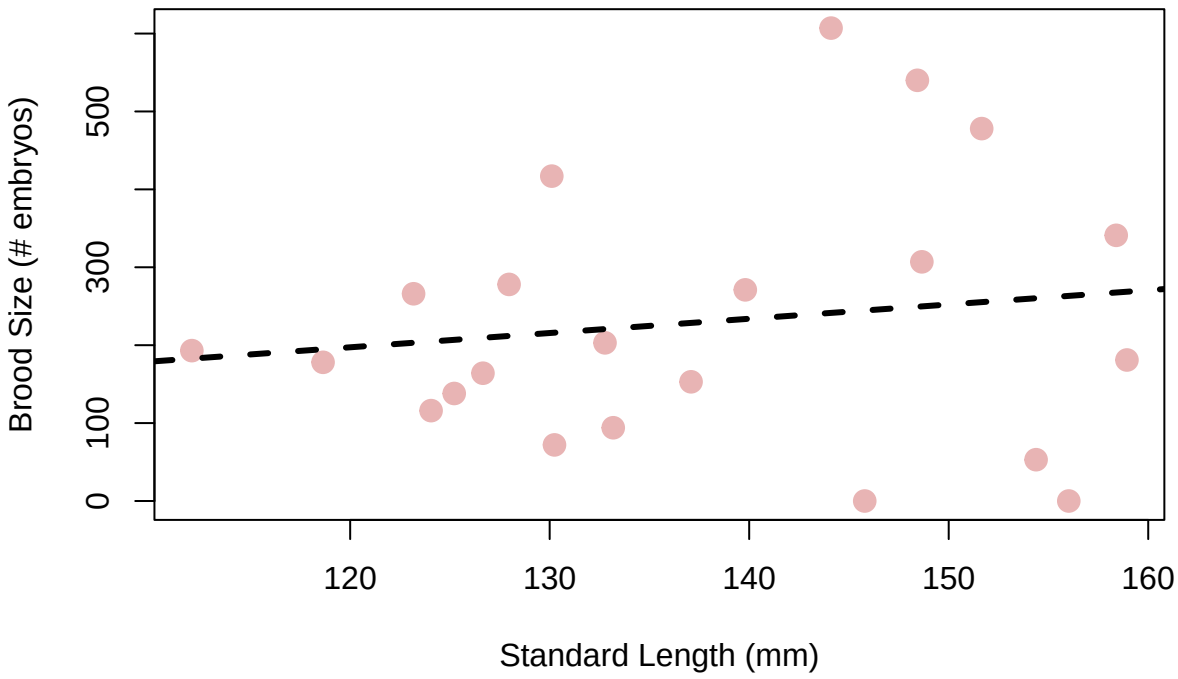


Figure 2: Scatterplot of the relationship between standard length (mm) and the number of embryos a male had.

```
##
##  Pearson's product-moment correlation
##
## data:  as.numeric(mated_mal$length) and mated_mal$totalEggs
## t = 0.68816, df = 20, p-value = 0.4993
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2879874  0.5391261
## sample estimates:
##       cor
## 0.1520871
```

The correlation actually decreases when we look at the overall size of the fish (Fig. 2).

Females

Across all 12 trials and 73 total females, there were 19 females that mated at least one time, 2 females that mated twice, and 0 that mated 3 times.

Looking across all females, including the ones that did not mate, this is what we find as the mean, sd, and se for the total number of embryos transferred from each female (across all of her mates if applicable) and how many of those developed versus didn't:

	mean	SD	SE	max	min
Number of Embryos	69.1780822	145.8034581	17.065004	607	0
Developed Embryos	64.4931507	134.6631727	15.7611322	566	0
Undeveloped Embryos	4.6849315	18.9442726	2.2172594	152	0

These values will be influenced by the number of 0s coming from females who did not mate. So let's look at the same thing, but this time for only females who had at least one successful mating:

	mean	SD	SE	max	min
Number of Embryos	265.7894737	172.8414749	39.6525538	607	53
Developed Embryos	247.7894737	156.826365	35.9784356	566	52
Undeveloped Embryos	18	34.3883055	7.8892183	152	0

We can see from the bottom table that even when we only include females who mated there is still a wide range in the number of eggs transferred. I want to see what relationship there may be between female body size (in terms of standard length, depth, and SVL) and the number of eggs she transferred. I also want to see on average how many eggs were transferred per mating. I'm going to calculate this by taking the total number of eggs and dividing it by the number of mates.

```
## [1] 244.2632
```

```
## [1] 18.6399
```

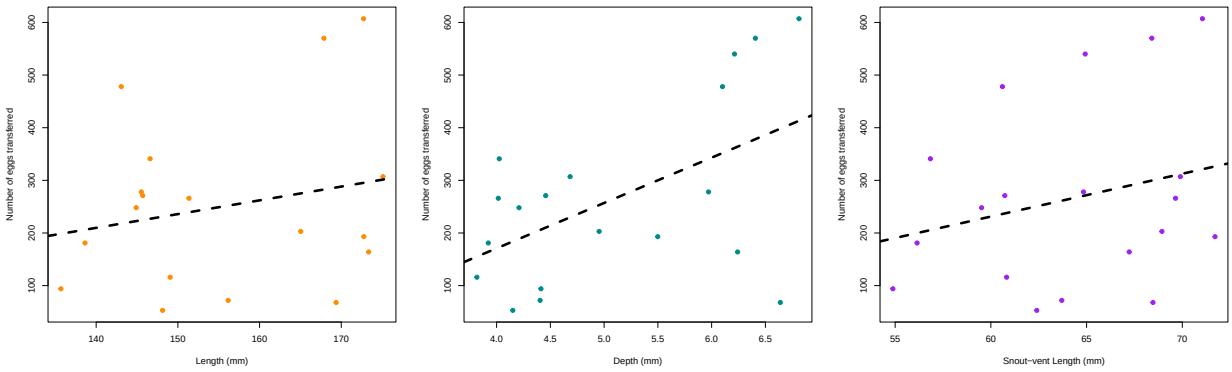


Figure 3: Scatterplot of the relationship between female size metrics and the number of eggs transferred.

There also appears to be a correlation between female body size and the number of eggs transferred, especially in terms of depth (Fig. 3). Let's run some correlations tests to see what they say.

```
##
##  Pearson's product-moment correlation
##
## data:  mated_fem$length and as.numeric(mated_fem$totalEggs)
## t = 0.87523, df = 16, p-value = 0.3944
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2811797  0.6188619
## sample estimates:
##       cor
## 0.2137503
```

```
##
##  Pearson's product-moment correlation
##
## data:  mated_fem$depth and as.numeric(mated_fem$totalEggs)
## t = 2.5535, df = 17, p-value = 0.02056
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.09504755 0.79145614
## sample estimates:
##       cor
## 0.5265259
```

```
##
##  Pearson's product-moment correlation
##
## data:  mated_fem$svl and as.numeric(mated_fem$totalEggs)
## t = 1.0581, df = 17, p-value = 0.3048
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2318064  0.6314863
## sample estimates:
##       cor
## 0.2485751
```

There is no sig. correlation between lenght or svl and the number of eggs transferred but we do see a significantly positive relationship between depth and number of eggs transferred!

Differences between mated individuals and unmated individuals

I want to now see if there are any significant differences in the sizes of individuals who mated vs individuals that didn't mate in males and females.

Males

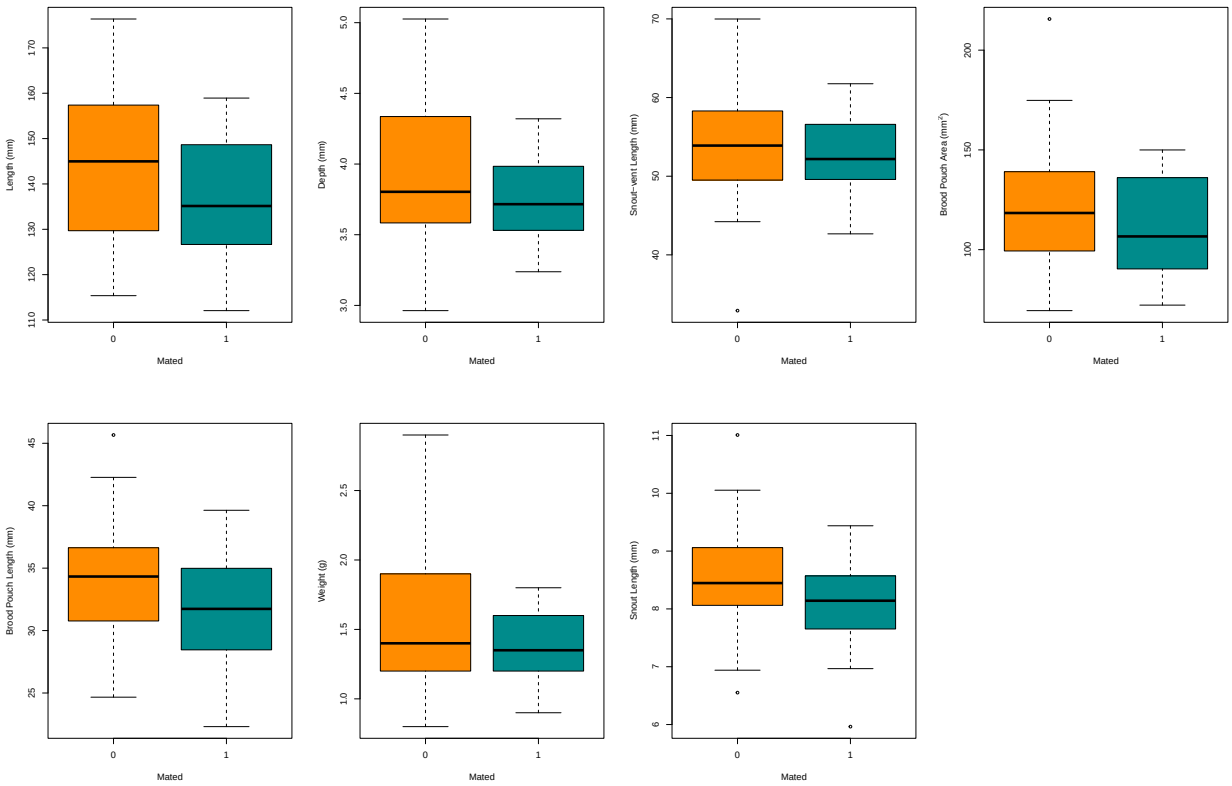


Figure 4: Six different morphometrics compared between males who sucessfully mated versus those that didn't. Orange represents unmated and blue represents mated males.

It appears that males who didn't mate are at least slightly larger than the males who did mate, I want to see if there is any significance behind that.

```
var.test(mal_succ$length ~ mal_succ$preg_status) #equal
var.test(mal_succ$depth ~ mal_succ$preg_status) #not equal
var.test(mal_succ$svl ~ mal_succ$preg_status) #not equal
var.test(mal_succ$bp_area ~ mal_succ$preg_status) #equal
var.test(mal_succ$bp_length ~ mal_succ$preg_status) #equal
var.test(mal_succ$weight ~ mal_succ$preg_status) #not equal
var.test(mal_succ$snout_len ~ mal_succ$preg_status)

t.test(mal_succ$length ~ mal_succ$preg_status,
       var.equal = TRUE)
t.test(mal_succ$depth ~ mal_succ$preg_status)
t.test(mal_succ$svl ~ mal_succ$preg_status)
t.test(mal_succ$bp_area ~ mal_succ$preg_status,
       var.equal = TRUE)
t.test(mal_succ$bp_length ~ mal_succ$preg_status,
       var.equal = TRUE)
t.test(mal_succ$weight ~ mal_succ$preg_status)
t.test(mal_succ$snout_len ~ mal_succ$preg_status,
```

```
        var.equal = TRUE)
```

There is no significant difference between the sizes of males who mated and males who didn't mate, but the unmated males were larger in every case. There was a nearly significant p-value for brood pouch length and snout length.

Females

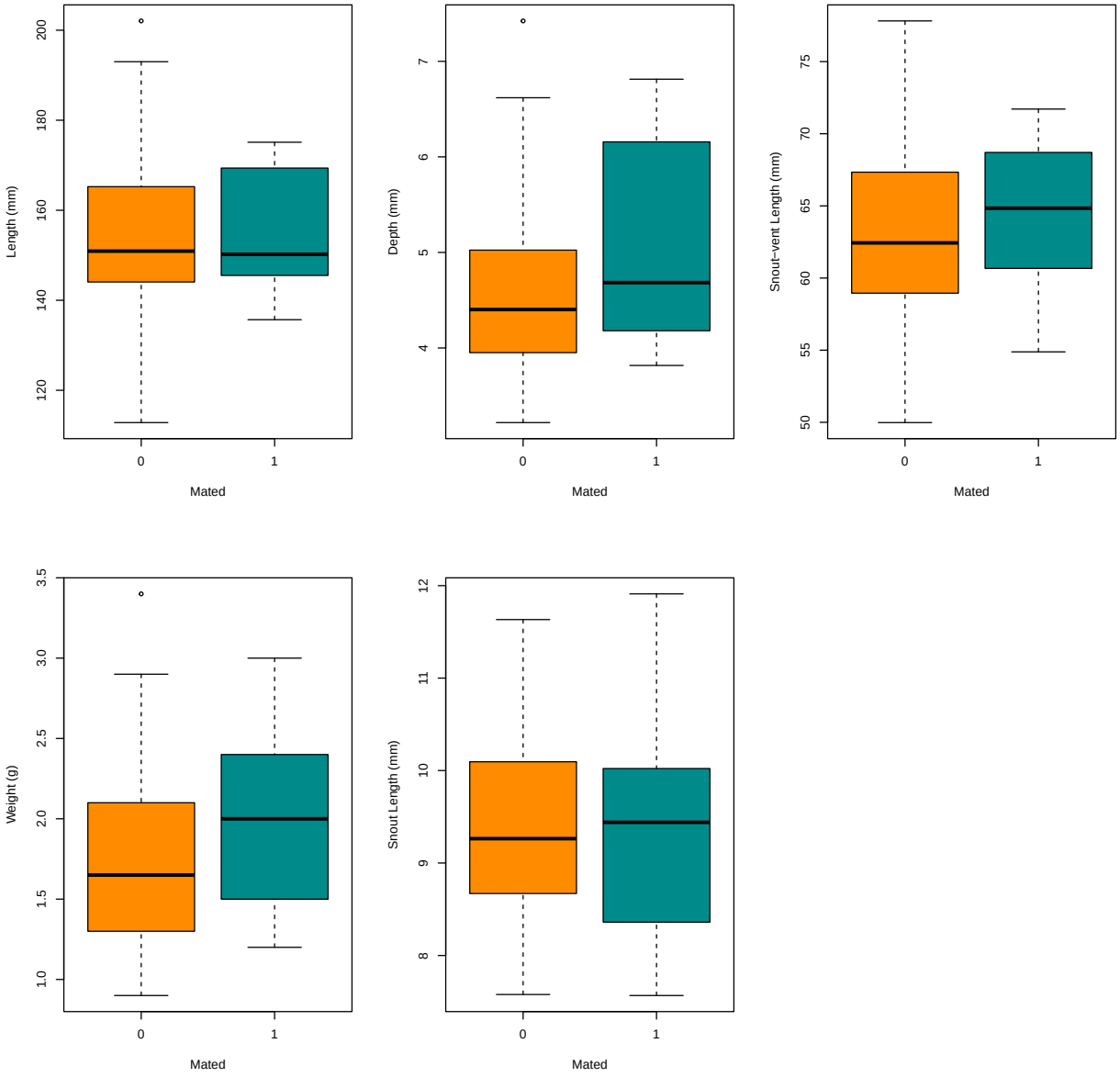


Figure 5: Four different morphometrics compared between females who sucessfully mated versus those that didn't. Orange represents unmated and blue represents mated females.

The pattern in which females are larger seems to swap for the different metrics and the differences aren't large at all. Let's do some tests to confirm this.

```
var.test(fem_succ$length ~ fem_succ$mated)
var.test(fem_succ$depth ~ fem_succ$mated)
var.test(fem_succ$svl ~ fem_succ$mated)
var.test(fem_succ$weight ~ fem_succ$mated)
var.test(fem_succ$snout_len ~ fem_succ$mated)

#All of the variances are equal
t.test(fem_succ$length ~ fem_succ$mated)
t.test(fem_succ$depth ~ fem_succ$mated)
t.test(fem_succ$svl ~ fem_succ$mated)
t.test(fem_succ$weight ~ fem_succ$mated)
t.test(fem_succ$snout_len ~ fem_succ$mated)
```

Unsurprisingly there is no sig. difference in females who mated vs. didn't mate for any of the size metrics, but in each case the females who did mate were just slightlyyyy larger than the ones who didn't.

Episode of Sexual Selection

Partitioning the Total Opportunity for Selection (I)

```
#Create a dataframe to store all of the intermediate values of fitness in
fem_succ_fitness <- data.frame(matrix(ncol = ncol(fem_succ) + 9,
                                      nrow = 0))
colnames(fem_succ_fitness) <- c(colnames(fem_succ),
                                "w1", "w1_squared",
                                "W2", "W2_bar", "w2",
                                "W3", "W3_bar", "w3", "i3")

#Create a dataframe to store the final calculations of I in
opp_selection_episodes_fem <- data.frame(matrix(ncol = 12,
                                                nrow = 0))
colnames(opp_selection_episodes_fem) <- c("trial_num", "I_1", "I_1per", "I_2", "I_2per",
                                          "I_3", "I_3per", "I_12", "I_12per",
                                          "I", "Iper", "I_s")

for (trial in unique(fem_succ$trial_num)) {

  #Subset the overall dataframe to work with an individual trial
  tmp <- fem_succ[fem_succ$trial_num == trial, ]

  #Calculate the absolute pre-copulatory fitness (Eq. 14 Arnold & Wade 1984)
  tmp$w1 <- tmp$MatingSuccess/mean(tmp$MatingSuccess) #Relative mating success
  tmp$w1_squared <- (tmp$w1)^2

  I_1 <- var(tmp$w1) #Variance in relative mating success

  #Post-copulatory selection event 1 (Number of eggs transferred) (Eq. 15 Arnold & Wade 1984)
  tmp$W2 <- ifelse(tmp$MatingSuccess > 0,
```

```
                  tmp$totalEggs/tmp$MatingSuccess,
                  0) #Number of eggs per mate
  tmp$W2_bar <- tmp$W2 * (tmp$w1/nrow(tmp)) #Number of eggs per mate adjusted by the # of individuals w
  tmp$w2 <- tmp$W2/sum(tmp$W2_bar)

  I_2 <- (sum((tmp$w1 * (tmp$w2)^2))/nrow(tmp) - 1) * nrow(tmp)/(nrow(tmp) - 1)

  #Post-copulatory selection event 2 (Number of eggs developed) (Eq. 16 Arnold & Wade 1984)
  tmp$W3 <- ifelse(tmp$totalEggs > 0,
                  tmp$NumDeveloped/tmp$totalEggs,
                  0) #Proportion of transferred eggs that developed
  tmp$W3_bar <- tmp$W3 * ((tmp$totalEggs/mean(tmp$totalEggs))/nrow(tmp)) #Prop. of eggs developed adjus
  tmp$w3 <- tmp$W3/sum(tmp$W3_bar)
  tmp$i3 <- ((tmp$totalEggs/mean(tmp$totalEggs))/nrow(tmp)) * ((tmp$w3 - 1)^2)

  I_3 <- sum(tmp$i3) * nrow(tmp)/(nrow(tmp) - 1)

  I_12 <- var(tmp$totalEggs)/(mean(tmp$totalEggs)^2)

  #Total selection
  I <- var(tmp$NumDeveloped)/(mean(tmp$NumDeveloped)^2)

  I_s <- var(tmp$MatingSuccess)/(mean(tmp$MatingSuccess)^2)

  #Calculating percentages for each selection event
  I_1per <- (I_1/I)*100
  I_2per <- (I_2/I)*100
  I_3per <- (I_3/I)*100
  I_12per <- (I_12/I)*100
  Iper <- (I/I)*100

  #Combining all of the selection values (Is) and saving the output
  trial_num <- trial
  selection <- cbind(trial_num, I_1, I_1per, I_2, I_2per, I_3, I_3per,
                     I_12, I_12per, I, Iper, I_s)

  opp_selection_episodes_fem <- rbind(opp_selection_episodes_fem, selection)

  #Save the intermediate values
  fem_succ_fitness <- rbind(fem_succ_fitness, tmp)
}
```

```
#Create a dataframe to store all of the intermediate values of fitness in
mal_succ_fitness <- data.frame(matrix(ncol = ncol(mal_succ) + 9,
                                      nrow = 0))
colnames(mal_succ_fitness) <- c(colnames(mal_succ),
                                "w1", "w1_squared",
                                "W2", "W2_bar", "w2",
                                "W3", "W3_bar", "w3", "i3")

#Create a dataframe to store the final calculations of I in
opp_selection_episodes_mal <- data.frame(matrix(ncol = 12,
                                                nrow = 0))
```

```r
colnames(opp_selection_episodes_mal) <- c("trial_num", "I_1", "I_1per", "I_2", "I_2per",
                                          "I_3", "I_3per", "I_12", "I_12per",
                                          "I", "Iper", "I_s")

for (trial in unique(mal_succ$trial_num)) {

  #Subset the overall dataframe to work with an individual trial
  tmp <- mal_succ[mal_succ$trial_num == trial, ]

  #Calculate the absolute pre-copultory fitness (Eq. 14 Arnold & Wade 1984)
  tmp$w1 <- tmp$MatingSuccess/mean(tmp$MatingSuccess) #Relative mating success
  tmp$w1_squared <- (tmp$w1)^2

  I_1 <- var(tmp$w1) #Variance in relative mating success

  #Post-copulatory selection event 1 (Number of eggs transferred) (Eq. 15 Arnold & Wade 1984)
  tmp$W2 <- ifelse(tmp$MatingSuccess > 0,
                   tmp$totalEggs/tmp$MatingSuccess,
                   0) #Number of eggs per mate
  tmp$W2_bar <- tmp$W2 * (tmp$w1/nrow(tmp)) #Number of eggs per mate adjusted by the # of individuals w
  tmp$w2 <- tmp$W2/sum(tmp$W2_bar)

  I_2 <- (sum((tmp$w1 * (tmp$w2)^2))/nrow(tmp) - 1) * nrow(tmp)/(nrow(tmp) - 1)

  #Post-copulatory selection event 2 (Number of eggs developed) (Eq. 16 Arnold & Wade 1984)
  tmp$W3 <- ifelse(tmp$totalEggs > 0,
                   tmp$NumDeveloped_Calc/tmp$totalEggs,
                   0) #Proportion of transferred eggs that developed
  tmp$W3_bar <- tmp$W3 * ((tmp$totalEggs/mean(tmp$totalEggs))/nrow(tmp)) #Prop. of eggs developed adjus
  tmp$w3 <- tmp$W3/sum(tmp$W3_bar)
  tmp$i3 <- ((tmp$totalEggs/mean(tmp$totalEggs))/nrow(tmp)) * ((tmp$w3 - 1)^2)

  I_3 <- sum(tmp$i3) * nrow(tmp)/(nrow(tmp) - 1)

  I_12 <- var(tmp$totalEggs)/(mean(tmp$totalEggs)^2)

  #Total selection
  I <- var(tmp$NumDeveloped_Calc)/(mean(tmp$NumDeveloped_Calc)^2)

  I_s <- var(tmp$MatingSuccess)/(mean(tmp$MatingSuccess)^2)

  #Calculating percentages for each selection event
  I_1per <- (I_1/I)*100
  I_2per <- (I_2/I)*100
  I_3per <- (I_3/I)*100
  I_12per <- (I_12/I)*100
  Iper <- (I/I)*100

  #Combining all of the selection values (Is) and saving the output
  trial_num <- trial
  selection <- cbind(trial_num, I_1, I_1per, I_2, I_2per, I_3, I_3per,
                     I_12, I_12per, I, Iper, I_s)
```

```
  opp_selection_episodes_mal <- rbind(opp_selection_episodes_mal, selection)

  #Save the intermediate values
  mal_succ_fitness <- rbind(mal_succ_fitness, tmp)
}
```

Decomposition of selection differentials (s)

```
#Create a dataframe to store all of the intermediate values of fitness in
fem_succ_select_diff <- data.frame(matrix(ncol = ncol(fem_succ) + 6,
                                          nrow = 0))
colnames(fem_succ_select_diff) <- c(colnames(fem_succ),
                                    "fit1", "eggs_per_mate","fit2", "prop_dev", "fit3", "StdLength")

#Create a dataframe to store the final calculations of I in
select_diff_fem <- data.frame(matrix(ncol = 11,
                                     nrow = 0))
colnames(select_diff_fem) <- c("trial", "s1", "s2", "s3", "s12", "s123",
                               "s1_prime", "s2_prime", "s3_prime", "s12_prime", "s123_prime")

for (trial in unique(fem_succ$trial_num)) {

  #Subset the overall dataframe to work with an individual trial
  tmp <- fem_succ[fem_succ$trial_num == trial, ]

  #Calculate fitness relating to pre-cop. selection (#matings)
  tmp$fit1 <- tmp$MatingSuccess/mean(tmp$MatingSuccess) #Relative mating success

  #Calculate fitness relating to post-mating selection (#eggs transferred)
  tmp$eggs_per_mate <- tmp$totalEggs/tmp$MatingSuccess
  tmp$fit2 <- ifelse(tmp$MatingSuccess > 0,
                     tmp$eggs_per_mate/mean(tmp$eggs_per_mate, na.rm = TRUE),
                     0) #Relative eggs transferred

  #Calculate fitness relating to post-mating selection (eggs that developed)
  tmp$prop_dev <- (tmp$NumDeveloped/tmp$MatingSuccess)/tmp$eggs_per_mate
  tmp$fit3 <- ifelse(tmp$MatingSuccess > 0,
                     tmp$prop_dev/mean(tmp$prop_dev, na.rm = TRUE),
                     0)

  #Standardizing the trait value to have a mean of 0 and sd of unity
  tmp$StdLength <- (tmp$svl - mean(tmp$svl))/sd(tmp$svl)

  #Calculating the absolute selection differentials (s)
  s1 <- cov(tmp$svl, tmp$fit1)
  s12 <- cov(tmp$svl, tmp$fit2)
  s123 <- cov(tmp$svl, tmp$fit3)
  s2 <- s12 - s1
  s3 <- s123 - s12

  #Calculating the standardized selection differentials (s')
  s1_prime <- cov(tmp$StdLength, tmp$fit1)
```

```
  s12_prime <- cov(tmp$StdLength, tmp$fit2)
  s123_prime <- cov(tmp$StdLength, tmp$fit3)
  s2_prime <- s12_prime - s1_prime
  s3_prime <- s123_prime - s12_prime

  #Combining all of the selection differentials (s, s') and saving the output
  selection <- cbind(trial, s1, s2, s3, s12, s123,
                     s1_prime, s2_prime, s3_prime, s12_prime, s123_prime)

  select_diff_fem <- rbind(select_diff_fem, selection)

  #Save the intermediate values
  fem_succ_select_diff <- rbind(fem_succ_select_diff, tmp)
}
```

```
#Create a dataframe to store all of the intermediate values of fitness in
mal_succ_select_diff <- data.frame(matrix(ncol = ncol(mal_succ) + 6,
                                          nrow = 0))
colnames(mal_succ_select_diff) <- c(colnames(mal_succ),
                                    "fit1", "eggs_per_mate","fit2", "prop_dev", "fit3", "StdLength")

#Create a dataframe to store the final calculations of I in
select_diff_mal <- data.frame(matrix(ncol = 11,
                                     nrow = 0))
colnames(select_diff_mal) <- c("trial", "s1", "s2", "s3", "s12", "s123",
                               "s1_prime", "s2_prime", "s3_prime", "s12_prime", "s123_prime")

for (trial in unique(mal_succ$trial_num)) {

  #Subset the overall dataframe to work with an individual trial
  tmp <- mal_succ[mal_succ$trial_num == trial, ]

  #Calculate fitness relating to pre-cop. selection (#matings)
  tmp$fit1 <- tmp$MatingSuccess/mean(tmp$MatingSuccess) #Relative mating success

  #Calculate fitness relating to post-mating selection (#eggs transferred)
  tmp$eggs_per_mate <- tmp$totalEggs/tmp$MatingSuccess
  tmp$fit2 <- ifelse(tmp$MatingSuccess > 0,
                     tmp$eggs_per_mate/mean(tmp$eggs_per_mate, na.rm = TRUE),
                     0) #Relative eggs transferred

  #Calculate fitness relating to post-mating selection (eggs that developed)
  tmp$prop_dev <- (tmp$NumDeveloped_Calc/tmp$MatingSuccess)/tmp$eggs_per_mate
  tmp$fit3 <- ifelse(tmp$MatingSuccess > 0,
                     tmp$prop_dev/mean(tmp$prop_dev, na.rm = TRUE),
                     0)

  #Standardizing the trait value to have a mean of 0 and sd of unity
  tmp$StdLength <- (tmp$svl - mean(tmp$svl))/sd(tmp$svl)

  #Calculating the absolute selection differentials (s)
  s1 <- cov(tmp$svl, tmp$fit1)
  s12 <- cov(tmp$svl, tmp$fit2)
```

```
  s123 <- cov(tmp$svl, tmp$fit3)
  s2 <- s12 - s1
  s3 <- s123 - s12

  #Calculating the standardized selection differentials (s')
  s1_prime <- cov(tmp$StdLength, tmp$fit1)
  s12_prime <- cov(tmp$StdLength, tmp$fit2)
  s123_prime <- cov(tmp$StdLength, tmp$fit3)
  s2_prime <- s12_prime - s1_prime
  s3_prime <- s123_prime - s12_prime

  #Combining all of the selection differentials (s, s') and saving the output
  selection <- cbind(trial, s1, s2, s3, s12, s123,
                     s1_prime, s2_prime, s3_prime, s12_prime, s123_prime)

  select_diff_mal <- rbind(select_diff_mal, selection)

  #Save the intermediate values
  mal_succ_select_diff <- rbind(mal_succ_select_diff, tmp)
}
```

Mate success versus Reproductive success (Bateman Gradient)

I now want to look at any relationship that may exist between mating success and reproductive success for males and females. The Bateman gradient will be calculated, which is the slope of the weighted least-squares regression of relative reproductive success (number of offspring divided by the mean) on mating success.

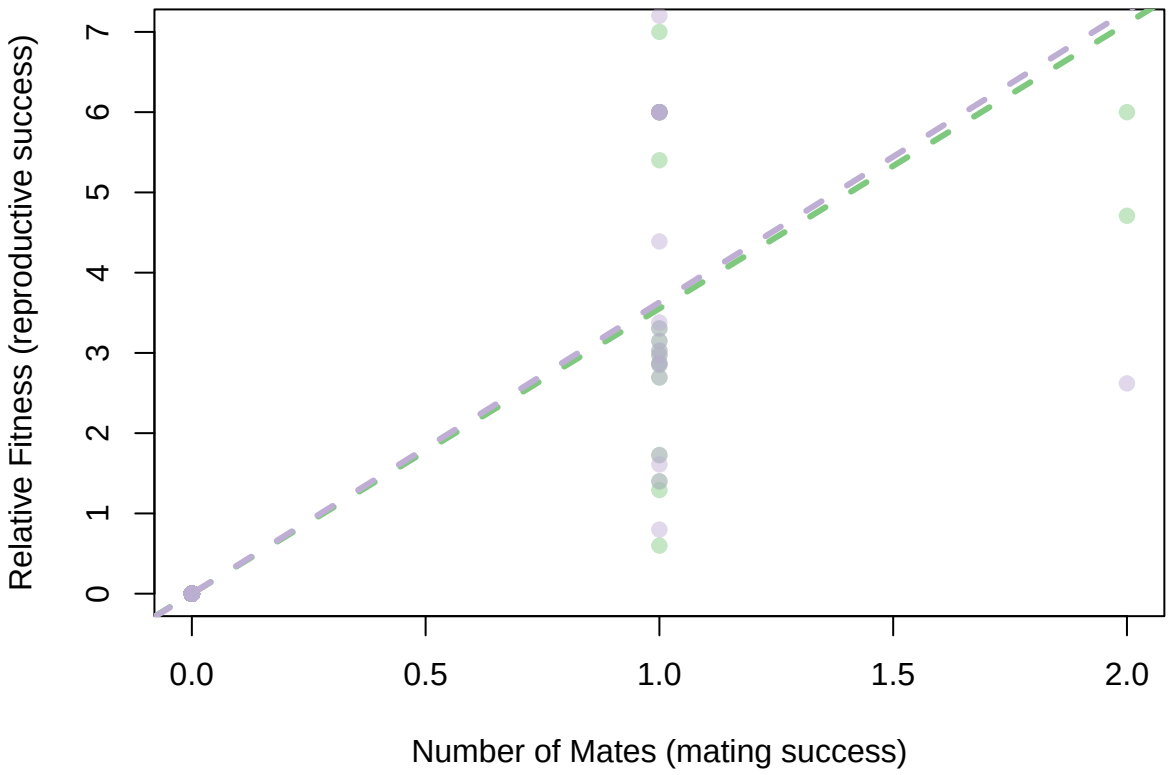


Figure 6: Relationship between reproductive success and mating success for male (purple) and female (green) *Syngnathus floridae*. Reproductive success is shown as relative fitness (i.e. number of offspring produced divided by the mean number of offspring produced). Bateman's gradient is shown as the weighted least-squares regression line (dashed) for males and females.

```
summary(wls_model_fem)
```

```
##
## Call:
## lm(formula = fem_bateman$rel_repo_fitness ~ fem_bateman$MatingSuccess,
##     weights = wt_fem)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -2.24470 -0.00095 -0.00095 -0.00095  2.61865
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                8.902e-05  9.333e-03    0.01    0.992
## fem_bateman$MatingSuccess 3.553e+00  2.196e-01   16.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

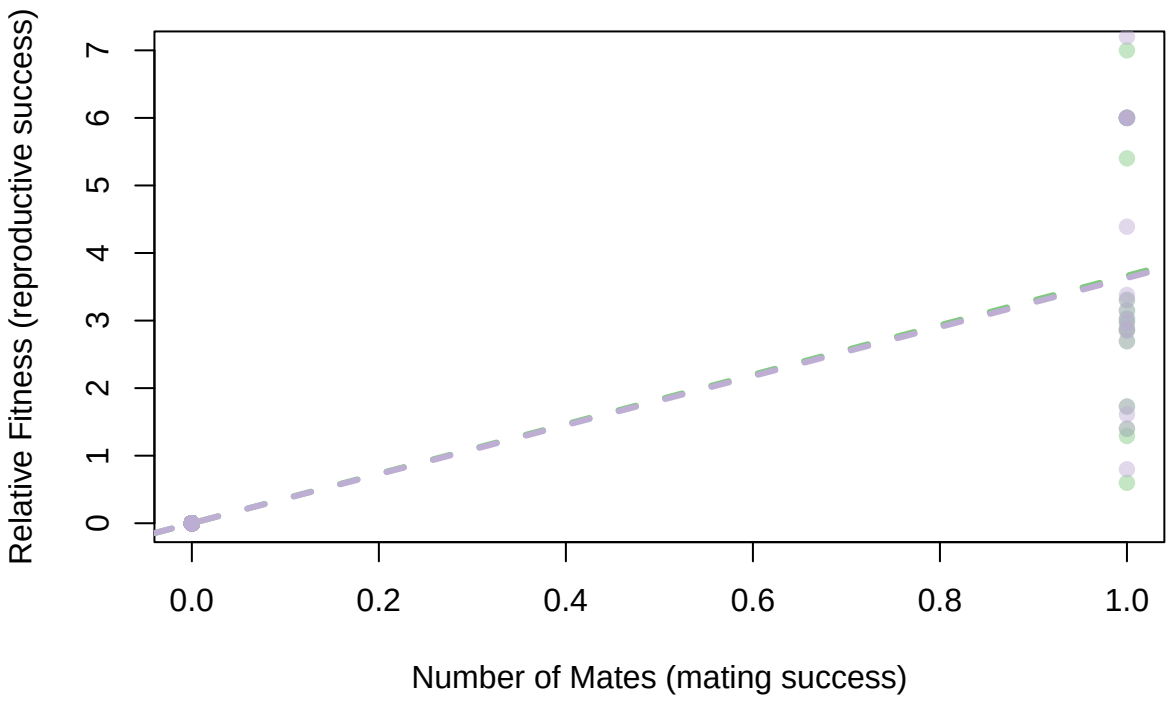
```
##
## Residual standard error: 0.7295 on 71 degrees of freedom
## Multiple R-squared:  0.7867, Adjusted R-squared:  0.7837
## F-statistic: 261.8 on 1 and 71 DF,  p-value: < 2.2e-16
```

```
summary(wls_model_mal)
```

```
##
## Call:
## lm(formula = mal_bateman$rel_repo_fitness ~ mal_bateman$MatingSuccess,
##     weights = wt_mal)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -1.77685 -0.00051 -0.00051 -0.00051  2.24100
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                2.947e-05  4.912e-03   0.006    0.995
## mal_bateman$MatingSuccess 3.630e+00  2.218e-01  16.367   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6229 on 72 degrees of freedom
## Multiple R-squared:  0.7882, Adjusted R-squared:  0.7852
## F-statistic: 267.9 on 1 and 72 DF,  p-value: < 2.2e-16
```

Omitting females with high mating

To make sure the two females that mated 3 times are not significantly affect the Bateman gradient I am re-plotting and re-running the model with those points omitted.




```
##
## Call:
## lm(formula = fem_bateman$rel_repo_fitness[fem_bateman$MatingSuccess <
##     2] ~ fem_bateman$MatingSuccess[fem_bateman$MatingSuccess <
##     2], weights = wt_fem2)
##
## Weighted Residuals:
##    Min     1Q Median     3Q    Max
## -1.807  0.000  0.000  0.000  1.967
##
## Coefficients:
##                                                          Estimate Std. Error
## (Intercept)                                            -4.637e-47  4.429e-17
## fem_bateman$MatingSuccess[fem_bateman$MatingSuccess < 2]  3.664e+00  2.325e-01
##                                                        t value Pr(>|t|)
## (Intercept)                                               0.00        1
## fem_bateman$MatingSuccess[fem_bateman$MatingSuccess < 2]   15.76   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5651 on 69 degrees of freedom
## Multiple R-squared:  0.7827, Adjusted R-squared:  0.7795
## F-statistic: 248.5 on 1 and 69 DF,  p-value: < 2.2e-16
```

It doesn't look like omitting those few individuals has any effect on the results of the Bateman gradient.

Visualizing post-copulatory selection

As a way to visualize selection acting AFTER the mating event (post-copulatory selection) I am plotting the proportion of eggs that survived against mating success. Hopefully this will tell us if acquiring more mates is having any affect on the ability for the eggs to develop.

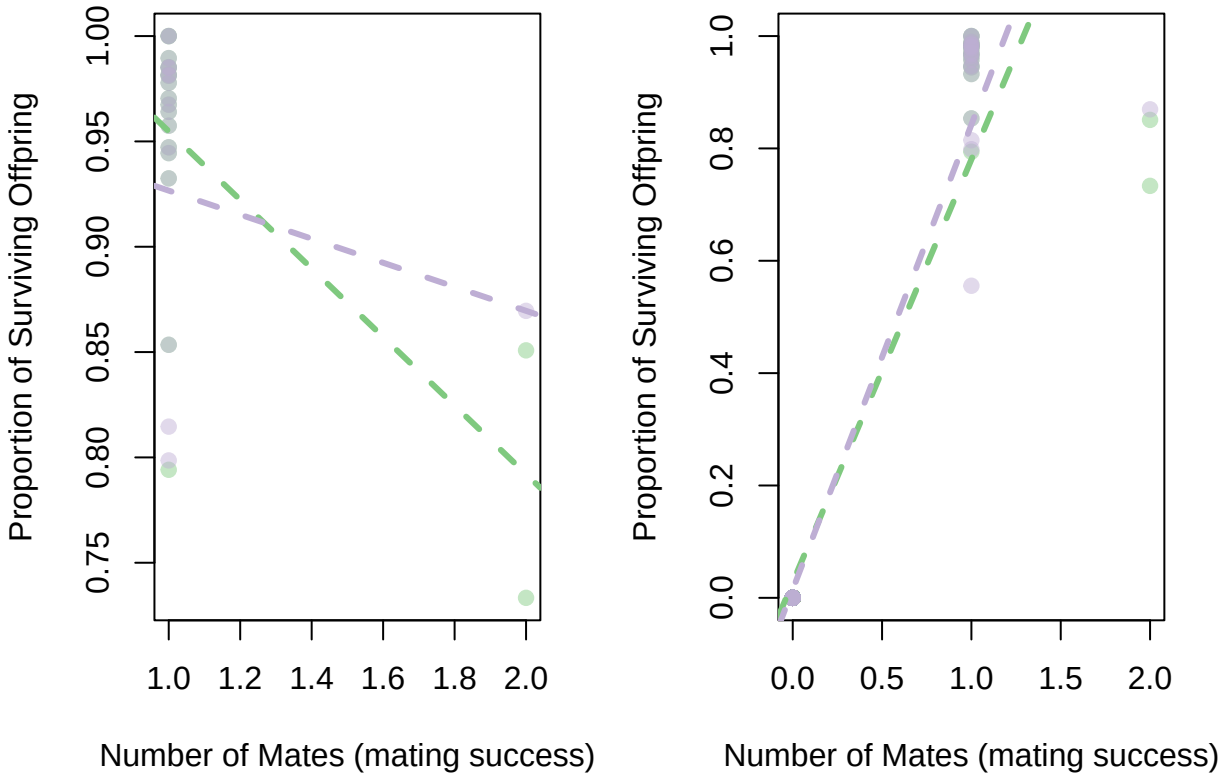


Figure 7: Plotting the relationship between the proportion of eggs that developed and the number of mates aquired for both males (purple) and females (green). This was done omitting the individuals that did not mate (left) and including those individuals (right).