FAS6337C - Lab 5

Marcel Gietzmann-Sanders

Testing for Differences in Length at Maturity

Data for this laboratory are from white grunt Haemulon plumieri, a popular reef fish in the Gulf of Mexico. Data were provided by Dr. Debra Murie. Fish maturity was determined by histology based on headboat collections from Tampa Bay, Florida.

```
setwd("/workspaces/schooling/population_dynamics/lab_5/")
data <- read.table("data/size_mat.txt", header=T, sep="")
head(data)
```

```
##      SEX MAT MTL Age
## 1 female   0  88   0
## 2 female   0  99   0
## 3 female   0 104   0
## 4 female   0 115   0
## 5 female   0 117   0
## 6 female   0 162   1
```

The objectives of this laboratory are:

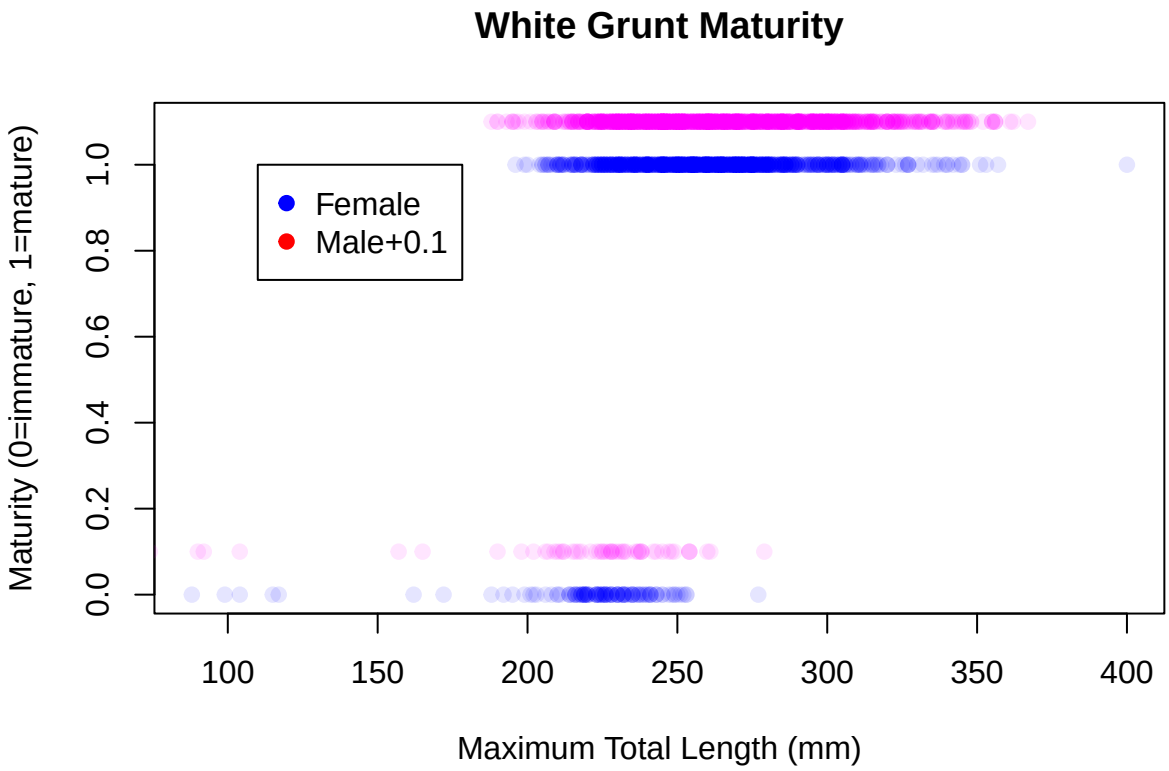
1. Identify whether size at maturity differs between male and female white grunt.
2. Explore using a logistic regression to model binomial data.

1. Plot the variable "MAT", which is maturity (0 = immature, 1 = mature), against maximum total length (MTL) for each sex (use either program).

```
colnames(data) <- tolower(colnames(data))
females <- data[data$sex=="female",]
males <- data[data$sex=="male",]
```

```
plot(
  females$mtl,
  females$mat,
  xlab="Maximum Total Length (mm)",
  ylab="Maturity (0=immature, 1=mature)",
  main="White Grunt Maturity",
  col=rgb(red = 0, green = 0, blue = 1, alpha = 0.1),
  pch = 19,
  ylim=c(0,1.1)
)
points(
  males$mtl,
  males$mat+0.1,
  col=rgb(red = 1, green = 0, blue = 1, alpha = 0.1),
  pch = 19
```

```
)
legend(x=110, y=1, pch=19, legend=c("Female", "Male+0.1"), col=c("blue", "red"))
```



Are there any observable differences evident between sexes?

- Some males reach maturity at a lower length than females.
- However males stay immature to longer lengths as well.
- So I suppose there is just more variation in time to maturity in males.

3. Estimate L_{mat} (length at 50% maturity) and σ (the steepness of the curve) for each sex by: 1) Inputting starting values for L_{mat} and σ; 2) Calculating the log likelihood of the binomial distribution for each data point, and then sum those to obtain the total log likelihood of all the data; 3) Maximizing the sum of the log-likelihoods by solving for the L_{mat} and σ parameters for each sex using Solver in Excel and optim in R.

Report L_{mat} and σ for each sex.

Do the predictions of length at maturity differ between sexes?

Yes males have a smaller L_{mat} than the females. And the steepness of their curve is less as well due to the higher σ.

What is the minimum predicted length of a fully mature fish (p > 0.99)?

- Females: $280mm$

- Males: $275mm$

4. Repeat questions 2 & 3 in R. Answer the same questions in Q3

```
get_nll <- function(theta) {
    f_lmat50 <- theta[1]
    f_sig <- theta[2]
    m_lmat50 <- theta[3]
    m_sig <- theta[4]

    prob_mat_female <- 1 / (1 + exp(-(females$mtl - f_lmat50) / f_sig))
    prob_mat_male <- 1 / (1 + exp(-(males$mtl - m_lmat50) / m_sig))
    nll_female <- -1*sum(dbinom(females$mat, size=1, prob=prob_mat_female, log=T))
    nll_male <- -1*sum(dbinom(males$mat, size=1, prob=prob_mat_male, log=T))
    return(nll_female + nll_male)
}
```

```
theta <- c(200, 20, 200, 20)
for (i in 1:10) {
    fit <- optim(theta, get_nll, hessian=T)
    theta <- fit$par
}
theta
```

```
## [1] 212.79359  14.50659 183.73361  19.67672
```

The predictions are the same as in Excel. All of the answers to the questions are the same.

Using the code used in Lab 2, part 2 for obtaining confidence intervals from the Hessian matrix, report the upper and lower 95% confidence intervals as well.

```
sterr <- sqrt(diag(solve(fit$hessian)))
sterr
```

```
## [1] 2.429862 1.362100 6.891295 2.605947
```

```
ALPHA = 0.05
U_f_lmat50 = theta[1] + qnorm(1-(ALPHA/2)) * sterr[1]
L_f_lmat50 = theta[1] - qnorm(1-(ALPHA/2)) * sterr[1]
U_f_sig = theta[2] + qnorm(1-(ALPHA/2)) * sterr[2]
L_f_sig = theta[2] - qnorm(1-(ALPHA/2)) * sterr[2]
U_m_lmat50 = theta[3] + qnorm(1-(ALPHA/2)) * sterr[3]
L_m_lmat50 = theta[3] - qnorm(1-(ALPHA/2)) * sterr[3]
U_m_sig = theta[4] + qnorm(1-(ALPHA/2)) * sterr[4]
L_m_sig = theta[4] - qnorm(1-(ALPHA/2)) * sterr[4]

results <- rbind(
    cbind(U_f_lmat50, L_f_lmat50),
    cbind(U_f_sig, L_f_sig),
    cbind(U_m_lmat50, L_m_lmat50),
    cbind(U_m_sig, L_m_sig)
)
colnames(results) <- c("Upper", "Lower")
rownames(results) <- c("f_lmat50", "f_sig", "m_lmat50", "m_sig")
results
```

```
##              Upper     Lower
## f_lmat50 217.55604 208.03115
## f_sig     17.17626  11.83692
## m_lmat50 197.24030 170.22692
## m_sig     24.78428  14.56916
```

5. Use an AIC model comparison procedure to test whether the parameters are shared between males and females (revisit Lab 2, Part 1 for help).

```
subset_theta <- function(theta, params_to_fit) {
    input <- c()
    for (i in 1:length(params_to_fit)) {
        input <- append(input, theta[params_to_fit[i]])
    }
    return(input)
}

update_theta <- function(theta, input, params_to_fit, index_to_share) {
    v = theta
    for (i in 1:length(params_to_fit)) {
        v[params_to_fit[i]] <- input[i]
    }
    for (i in 1:length(v)) {
        if (!(i %in% params_to_fit)) {
            v[i] <- v[index_to_share[i]]
        }
    }
    return(v)
}

do_fit <- function(theta, params_to_fit, index_to_share) {

    fun <- function(input) {
        v = update_theta(theta, input, params_to_fit, index_to_share)
        return(get_nll(v))
    }

    input <- subset_theta(theta, params_to_fit)
    for (i in 1:10) {
        fit <- optim(input, fun, hessian=T)
        input <- fit$par
    }
    theta <- update_theta(theta, input, params_to_fit, index_to_share)
    return(theta)
}

get_aic <- function(theta, params_to_fit) {
    nll <- get_nll(theta)
    k <- length(params_to_fit)
    aic <- 2*k + 2*nll
    return(c(nll, k, aic))
}
```

Report your results in a table (Include the model formula, the AIC value, the ΔAIC value, and the parameter estimates for each sex).

```
starting_guess <- c(200, 20, 200, 20)
index_to_share <- c(3, 4, 1, 2)
row_names <- c()
col_names <- c("f_lmat50", "f_sig", "m_lmat50", "m_sig", "nll", "params", "AIC")

params_to_fit <- c(1, 2, 3)
theta <- do_fit(starting_guess, params_to_fit, index_to_share)
aic_info <- get_aic(theta, params_to_fit)
row1 <- c(theta, aic_info)
row_names <- append(row_names, "L(h)sig(.)")

params_to_fit <- c(1, 2, 4)
theta <- do_fit(starting_guess, params_to_fit, index_to_share)
aic_info <- get_aic(theta, params_to_fit)
row2 <- c(theta, aic_info)
row_names <- append(row_names, "L(.)sig(h)")

params_to_fit <- c(1, 2)
theta <- do_fit(starting_guess, params_to_fit, index_to_share)
aic_info <- get_aic(theta, params_to_fit)
row3 <- c(theta, aic_info)
row_names <- append(row_names, "L(.)sig(.)")

params_to_fit <- c(1, 2, 3, 4)
theta <- do_fit(starting_guess, params_to_fit, index_to_share)
aic_info <- get_aic(theta, params_to_fit)
row4 <- c(theta, aic_info)
row_names <- append(row_names, "L(h)sig(h)")


results <- rbind(row1, row2, row3, row4)
colnames(results) <- col_names
rownames(results) <- row_names
results <- data.frame(results)

results <- results[order(results$AIC),]
results$delta_aic <- results$AIC - results$AIC[1]
results
```

```
##            f_lmat50    f_sig m_lmat50    m_sig      nll params      AIC
## L(h)sig(h) 212.7936 14.50659 183.7336 19.67672 425.2073      4 858.4145
## L(h)sig(.) 210.3023 16.33372 191.4329 16.33372 427.0518      3 860.1036
## L(.)sig(h) 199.7279 20.58221 199.7279 14.39045 437.8271      3 881.6543
## L(.)sig(.) 201.6183 16.82493 201.6183 16.82493 446.6029      2 897.2058
##            delta_aic
## L(h)sig(h)  0.000000
## L(h)sig(.)  1.689073
## L(.)sig(h) 23.239751
## L(.)sig(.) 38.791269
```

Does it appear that L_{mat} or σ differ between sexes?

- Both of the best models have L_{mat} as different between the two sexes.
- The model with σ free is only slightly better than the model with σ shared. So I'd conclude σ is not in fact different between the sexes.

Does this result make biological sense for White Grunt?

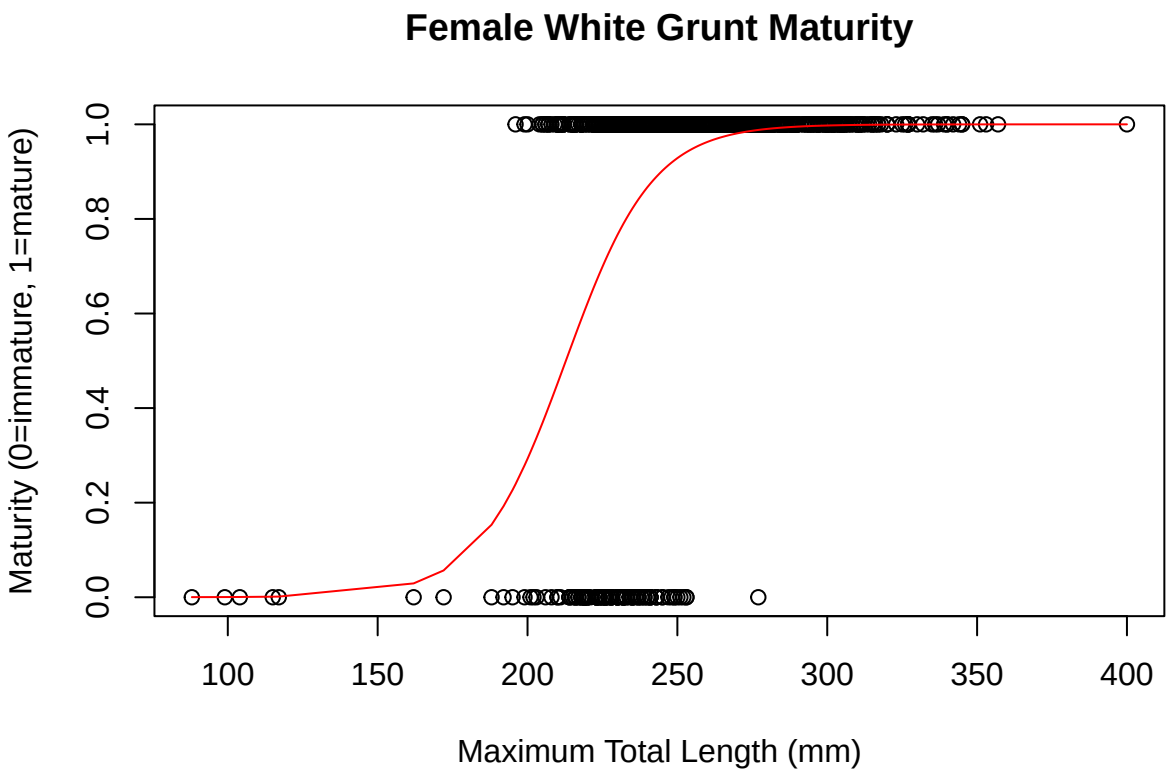
Nothing about this seems fishy to me. For the two sexes to mature at different ages seems perfectly reasonable.

6. Using the best model from question 5, predict maturity as a function of length for each sex using the parameter estimates from Solver in Excel or optim in R.

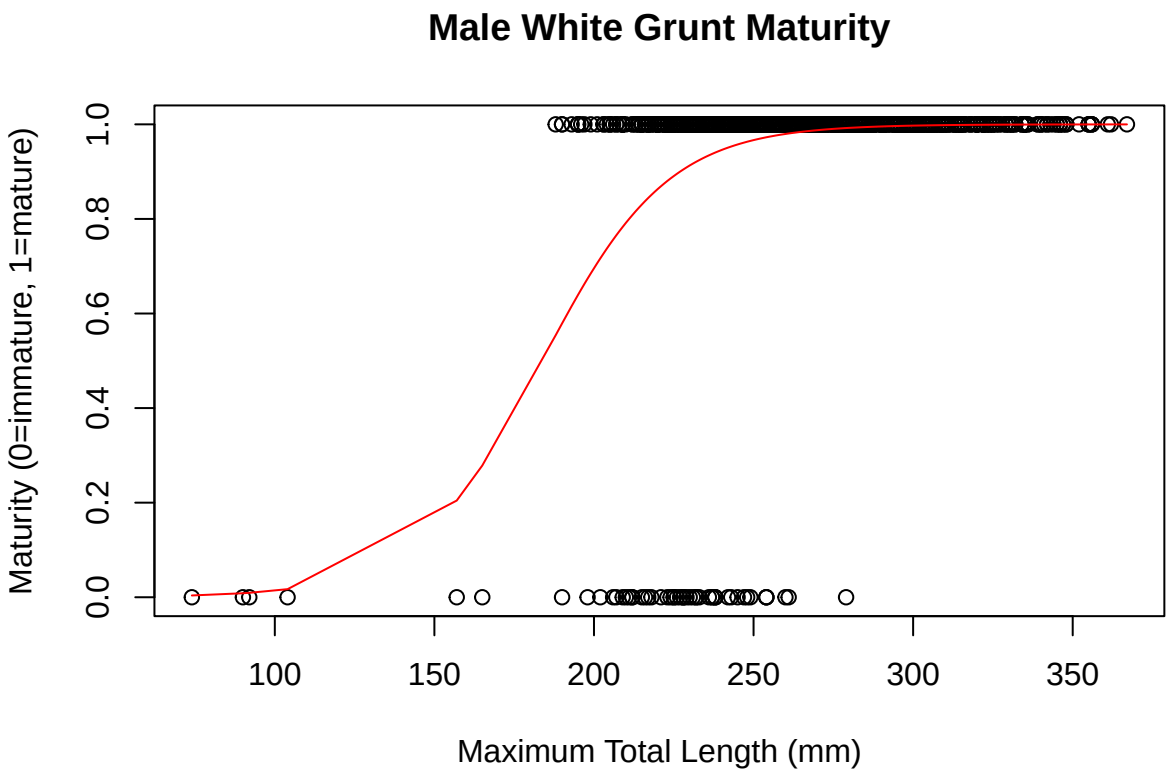
```
f_lmat50 = results[1, 1]
f_sig = results[1, 2]
m_lmat50 = results[1, 3]
m_sig = results[1, 4]

females$pred <- 1 / (1 + exp(-(females$mtl - f_lmat50) / f_sig))
males$pred <- 1 / (1 + exp(-(males$mtl - m_lmat50) / m_sig))
```

```
plot(
    females$mtl,
    females$mat,
    main="Female White Grunt Maturity",
    xlab="Maximum Total Length (mm)",
    ylab="Maturity (0=immature, 1=mature)"
)
lines(females$mtl[order(females$mtl)], females$pred[order(females$mtl)], col="red")
```




```
plot(
    males$mtl,
    males$mat,
    main="Male White Grunt Maturity",
    xlab="Maximum Total Length (mm)",
    ylab="Maturity (0=immature, 1=mature)"
)
lines(males$mtl[order(males$mtl)], males$pred[order(males$mtl)], col="red")
```



7. Explore the effect of sample size on the ability to reliably (you can turn sharing on then off to reset the simulation) tell the male and female parameters apart.

Adjust the female and male maturity length slider such that they are 10 mm apart, hold the sharpness at 0.05 for both, then determine the approximate sample size that results in $\hat{\theta}$ within 5 mm of the true (what you set in the slider).

- Female Maturity at Length - 255.0
- Male Maturity at Length - 265.0

This required that I put the sample size at 40 for each.

Increase steepness for either females or males, how does increasing steepness affect the sample size needed?

By dropping the sharpness to 0.03 I could drop to 30. I'm noticing a lot of variance though in the estimates each time I restart the simulation.

Increase the difference between female and male maturity length to 25 mm with steepness at 0.03, what is the sample size needed to have the 95% confidence intervals not overlap between sexes?

- Female Maturity at Length - 240.0
- Male Maturity at Length - 265.0

Seems like the point at with the 95% confidence intervals no longer intersect is at 70 samples for each sex.

8. Construct a generalized linear model (GLM) to fit the maturity data (for males and females), using the GLM call demonstrated in lab R code using equation 2.

```
logit <- function(x){
    return(log(x/(1-x)))
}
inv_logit <- function(x){
    return(exp(x)/(1+exp(x)))
}
```

```
glm_females <- glm(mat~mtl, data=females, family = binomial("logit"))
summary(glm_females)
```

```
##
## Call:
## glm(formula = mat ~ mtl, family = binomial("logit"), data = females)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9792   0.0987   0.2323   0.4541   1.6917
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -14.668121   1.505524  -9.743   <2e-16 ***
## mtl           0.068931   0.006472  10.651   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 739.61  on 976  degrees of freedom
## Residual deviance: 520.58  on 975  degrees of freedom
## AIC: 524.58
##
## Number of Fisher Scoring iterations: 6
```

```
c(
    (logit(0.5) - glm_females$coefficients[1]) / glm_females$coefficients[2],
    logit(0.5) - glm_females$coefficients[1]
)
```

```
## (Intercept) (Intercept)
##   212.79303    14.66812
```

```
glm_males <- glm(mat~mtl, data=males, family = binomial("logit"))

summary(glm_males)
```

```
##
## Call:
## glm(formula = mat ~ mtl, family = binomial("logit"), data = males)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.11422   0.08152   0.17420   0.30992   1.08676
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.336528   1.559198  -5.988 2.12e-09 ***
## mtl          0.050817   0.006731   7.550 4.35e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 436.70  on 1159  degrees of freedom
## Residual deviance: 329.83  on 1158  degrees of freedom
## AIC: 333.83
##
## Number of Fisher Scoring iterations: 7
```

```
c(
    (logit(0.5) - glm_males$coefficients[1]) / glm_males$coefficients[2],
    logit(0.5) - glm_males$coefficients[1]
)
```

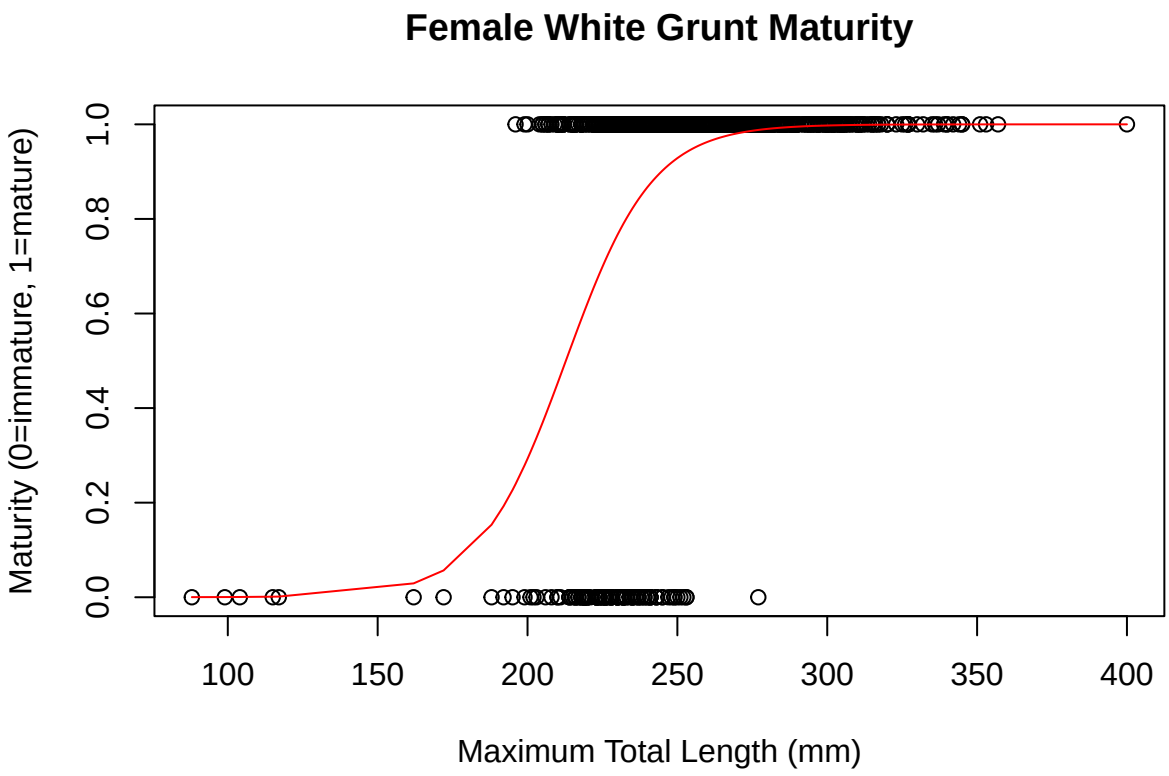
```
## (Intercept) (Intercept)
##  183.729868    9.336528
```

How is the parameter estimates of L_{mat} using Equation 3 different from the parameter estimates constructed 'by hand' for question 4?

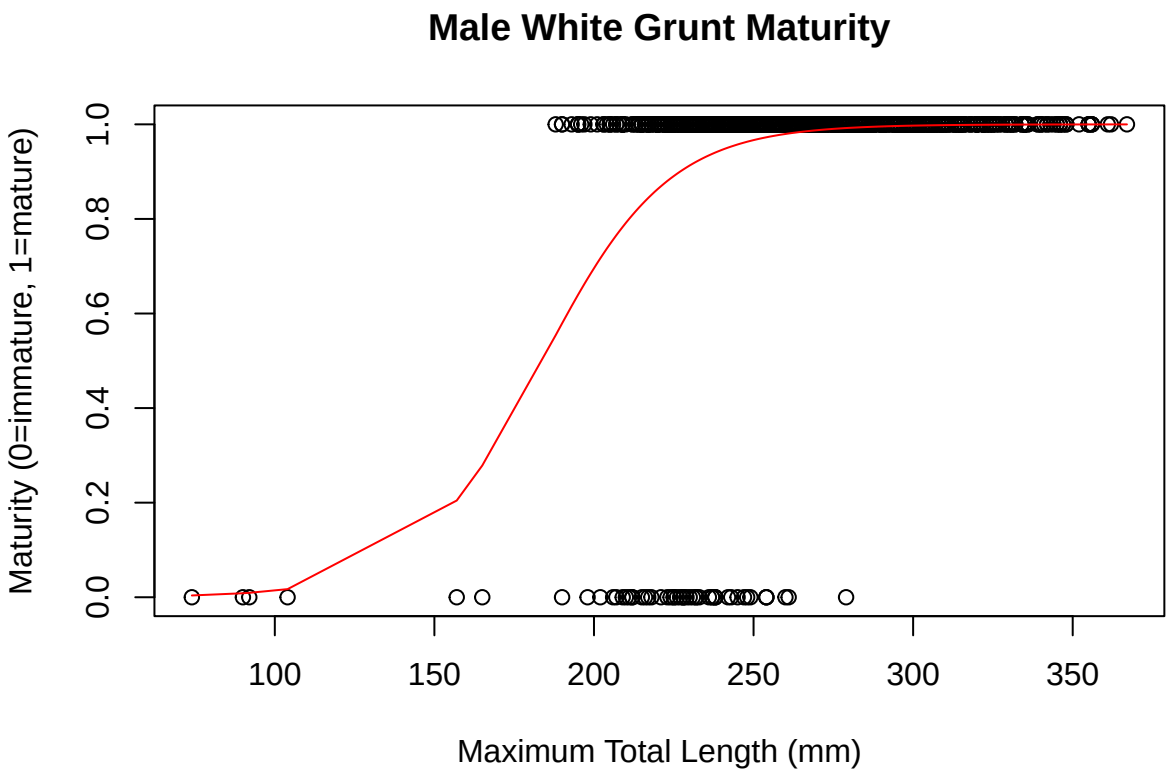
They are the same up to the third decimal place.

For each sex, plot the predictions from each analysis with the observed data

```
females$glm_pred <- predict(glm_females, type="response")
plot(
    females$mtl,
    females$mat,
    main="Female White Grunt Maturity",
    xlab="Maximum Total Length (mm)",
    ylab="Maturity (0=immature, 1=mature)"
)
lines(females$mtl[order(females$mtl)], females$glm_pred[order(females$mtl)], col="red")
```




```
males$glm_pred <- predict(glm_males, type="response")
plot(
    males$mtl,
    males$mat,
    main="Male White Grunt Maturity",
    xlab="Maximum Total Length (mm)",
    ylab="Maturity (0=immature, 1=mature)"
)
lines(males$mtl[order(males$mtl)], males$glm_pred[order(males$mtl)], col="red")
```



9. Answer the following:

What are the analytical benefits of each approach?

The GLM is nice in the sense that anything that can be treated as a probability can be modeled directly using very simple formulations. One can simply add additional terms to the model to account for additional factors. With the other approach you have to specify the formula yourself and that formula is going to get more and more unwieldy as you add more factors. However using optim means you get to control the formulation exactly so if you were wanting to do something more custom it enables you to do that.

The GLM is nice if you want to understand things in terms of odds ratios as it's going to give you everything more or less directly as that. However if you're really just interested in things like L_{mat50} then you can more easily extract that information from the optim approach.

What specifically does the GLM tell you that the 'by hand' approach does not, what specifically does it not give that the 'by hand' approach does? (if unfamiliar look here: https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faq-how-do-i-interpret-odds-ratios-in-logistic-regression/) The key, here, is to understand what information is given by very related, but different types of analyses when applied to the same data.

As mentioned above, GLMs for this problem think in terms of log odds rather than in terms of our σ and L_{mat50} parameters. What's really cool about this is the fact that we can compute other percentiles if we like or understand how changes in our factors change the odds of maturity (or whatever binary variable we're working with). However it also means that we have to do a bit more work to get the information we want if odds is not what we are after.