Lab 6 - Mark-recapture Estimation with Linear Models

Erik Blomberg

04/01/2021

Contents

1 Lab overview

As the title suggests, today's lab will center on incorporating concepts of linear modeling into our demographic analyses. We will start by reviewing the concept of a statistical linear model, drawing from principles of linear regression that you should already be familiar with. We will then apply these principles to the same concepts of categorical effects that we covered last week, using a new form of demographic analysis, the Known Fate Estimator, which is one type of analysis that is commonly applied to data from radio-marked animals.

This lab will be the first of a two-part series working with the same dataset, which I'll describe more below. For this first lab, we will focus on using linear models to test differences between groups and among time periods - basically the same approach we used last week with the dippers, but we will dig deeper into the use and interpretation of some new summary statistics that are inherent to the linear models. In next week's lab, we will extend these principles to the use of **continuous predictor variables**, which will allow us to ask questions about species-habitat relationships.

1.1 Learning Objectives

1) Review the use of linear models to reflect biological hypotheses, in this case related to variation in survival.

2) Become familiar with interpreting 'Beta' coefficient estimates, and learn how these relate to the underlying structure of the linear model, via the link function.

3) Introduce the concept of link functions, and the logit-link function specifically, for constraining linear models to produce probability estimates.

4) Use RMark to conduct a known fate analysis of survival data collected via radio-telemetry.

2 Analyzing mark-recapture data in a generalized linear modelling framework

During lab 5, we got our first look at estimating demographic parameters from mark-recapture data using the package RMark. We ran some pre-defined model structures, such as group and time effects, used AICc to evaluate support for these models and the hypotheses they represented, and interpreted the resulting survival probability estimates.

With this week's lab, we will build on these concepts by introducing a new approach for running, evaluating, and interpreting survival analyses: that of the **Generalized Linear Model**, or **GLM**. This won't be too different from the approaches we used last week in RMark in terms of how we implement the code, but it will add some extra dimensions to how we interpret the results. It will also give us a much more flexible framework for testing biological hypotheses, which this and next week's lab will cover.

Before we get into the RMark interpretation and coding, we need to take some time to understand GLMs in practice.

2.1 A refresher on linear models

You are all familiar with the concept of a linear model, the simplest form of which is a linear regression, such as those we applied on count data during Lab 3. As a refresher, the form of a linear regression is

$$y = mx + b \tag{1}$$

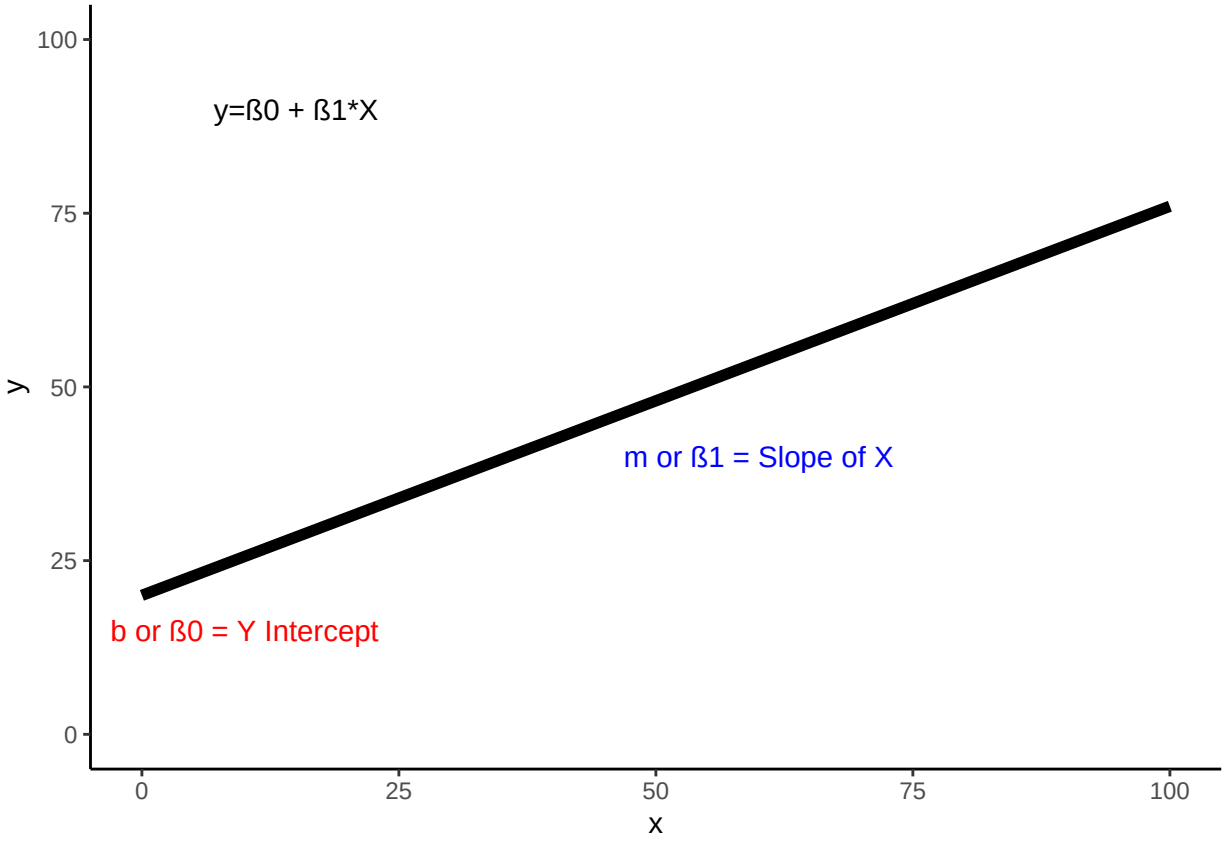
where y is the response or dependent variable, x is a predictor or independent variable, m is the model slope, or the expected change in y as a function of change in x, and b is the model intercept, or the expected value of y when x=0. As we move into implementation of GLMs in RMark, we will tweak the form of the model slightly to match the naming and

labeling conventions of variables in a GLM framework. Here is a mathematically identical form of equation 1, but adjusted to use β values in place of m and b

$$y = \beta_0 + \beta_1 * x_1 \tag{2}$$

where now β_0 is our model intercept (the equivalent to b) and β_1 gives the slope of the effect of x on y (the equivalent to m). We use the subscripted 1 for x, *x1*, to indicate that this is the first predictor variable in the model; as we will see later on, under this framework we can add additional terms to the model.

Just for completeness, lets take a second to remind ourselves what this looks like in practice.



2.2 Extending linear models to categorical data

2.2.1 The case of two groups

In the example above, x is a continuous predictor variable. However, in many cases we may have predictor variables that are **discrete**. We already saw some of these in Lab 5, for example, the effects of group membership (sex) on dipper survival. We can extend the principles of the linear model to accommodate categorical effects rather easily using the concept of **dummy variables**. A dummy variable is a strategy where we use values of 1s and 0s to represent group membership, and apply those numeric values (0,1) into our linear model, which is what gets the math to work. Lets use the above example of a male vs a female grouping variable, where we assign the female group a value of 1, and the male group a value of 0. Using our basic linear model (equation 2), we can substitute the values of 1 and 0 for x, so that for female dippers, the equation to estimate y would be:

$$y = \beta_0 + \beta_1 * 1 \tag{3}$$

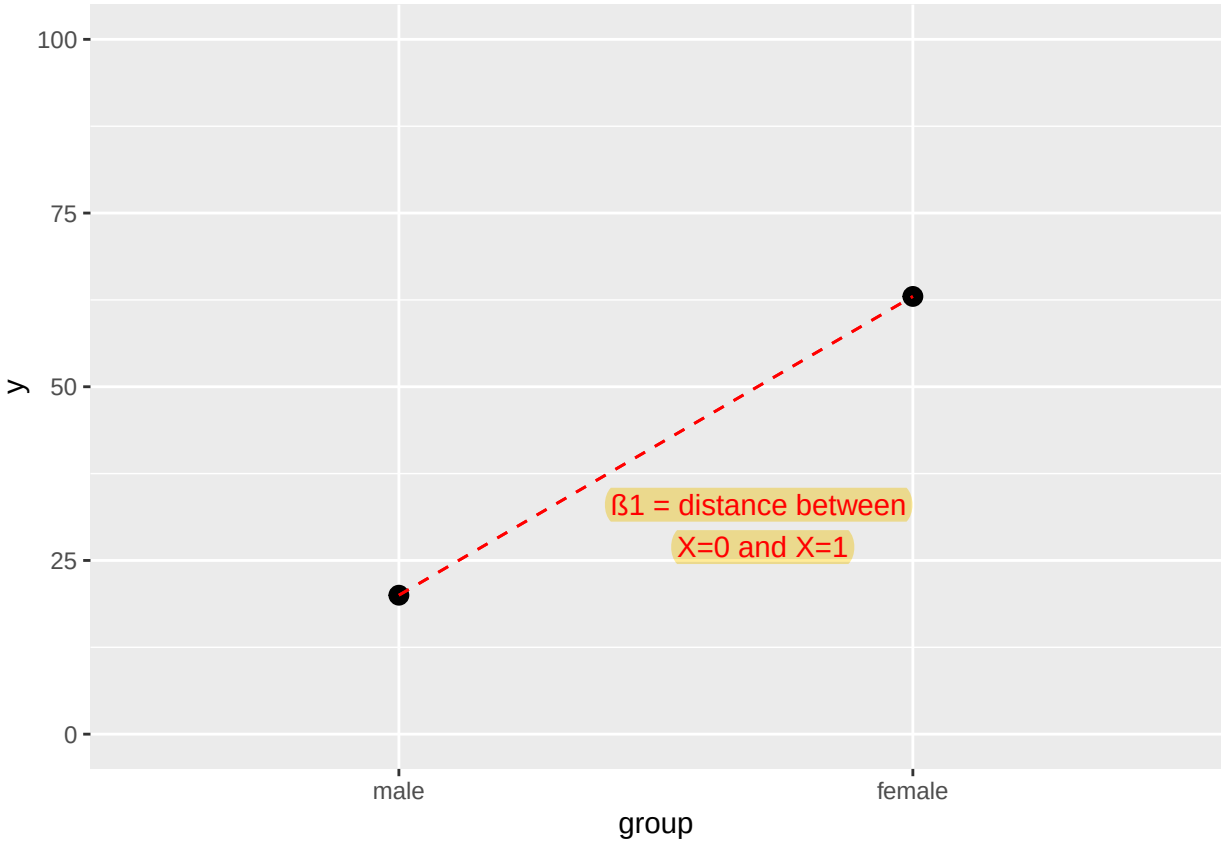
and for male dippers, the equation would be:

$$y = \beta_0 + \beta_1 * 0 \tag{4}$$

Note that in this case, the value $ _{1}*0 $ cancels out to 0, so for simplicity we could just write this equation as

$$y = \beta_0 \tag{5}$$

where now y is predicted in the model based solely on the intercept term. Use of the dummy variable is what therefore allows the model to distinguish between the expected mean value of females (x=1), and the expected mean value for males (x=0).If we consider what this looks like graphically,



Where you can imagine that β_0 now defines the expected value of y for the male group, and β_1 still gives us a 'slope' of sorts, as suggested by the red dashed line, but now tells us how much the expected value of y for the second group, females, deviates from that of the first group, males.

2.2.2 Extending to more than two groups

Now lets imagine a scenario where we want to distinguish among more than two groups - we'll start with three, call them Groups A, B, and C. This scenario could occur, as we will see in lab today, when we have three distinct study areas. Or it could fit the case of three different age classes, or any number of different categorical effects. We can extend our concept of dummy variables and linear models to accommodate >2 groups, with a goal of developing a model structure that allows groups A, B, and C to have different predicted values of y. I'll cut to the chase and show you what the model looks like, then we will cover how it works in practice.

$$y = \beta_0 + \beta_1 * x_B + \beta_2 * x_C \tag{6}$$

Here we still have the intercept term, β_0, and we have added two $\beta * x$ terms to distinguish among the 3 groups, with one $\beta * x$ describing the 'effect' of group B membership and a second describing the effect of group C membership. This allows us to distinguish unique responses, or estimates of y, for each of the three groups.

Building off (equation 6) we can develop individual notation for predictions for each group using this dummy variable concept. For Group A the equation is:

$$y = \beta_0 + \beta_1 * 0 + \beta_2 * 0 \tag{7}$$

For Group B the equation is:

$$y = \beta_0 + \beta_1 * 1 + \beta_2 * 0 \tag{8}$$

And for Group C, the equation is:

$$y = \beta_0 + \beta_1 * 0 + \beta_2 * 1 \tag{9}$$

As we saw with equation 5, we can simplify any of these expressions to remove the terms that are multiplied by zero, because they cancel each other out. The major take away here is that we can define the response variable y across 3 levels, A, B, and C, as a function of one intercept term (β_0) and two additional slope coefficients (β_1 and β_2). In this context, the parameter β_1 can be thought of as the mean difference in the response variable y between groups A and B. Similarly, β_2 reflects the mean difference in y between groups A and C. The distance between the values for β_2 and β_3 reflect the mean differences between groups B and C with respect to Y.

> ***Sidebar - Dummy variables and N-1 groups -*** But, wait, why aren't there 3 $\beta * x$ terms, you might be asking? One for each group? Again, let's define membership for group as a function of a combination of "dummy variables". A member of group A could have the combination (1 0 0), a member of group B (0 1 0), and group C (0 0 1). This totally works, but notice though that the third series of ones and zeros is actually redundant, since one group can always be defined as not belonging to they other two. So instead we can derive dummy variables where A= (0,0), B= (1,0), and C = (0,1).

There is a catch to applying these linear models for survival data, but I want to break up

the monotony of equations a bit, so lets get started with the lab exercise so we can look at some of these things in practice. Then we'll circle back to the math.

3 Lab Excecise 1 - Ruffed Grouse and Known Fates

The dataset you will be working with for this lab and the next is designed to reflect a typical survival analysis from radio-telemetry data, using ruffed grouse as a model organism. For full disclosure, these data are simulated. You'll be evaluating ecologically plausible relationships using these data, and I've designed the dataset based on what's known about the biology of the species. But please keep in mind that these data are ultimately simulated (i.e. I made them up) for the purpose of illustration.

3.1 A primer on Known Fate models

Generally speaking, most radio transmitters have the ability to transmit the movement status of the animal carrying it, which allows observers to tell whether the animal is alive (moving regularly) or dead (not moving). Given that we routinely obtain signals for all animals, we can assume that we know the live or dead status of each birds at all times. That is to say with radio survival data we assume that our detection probability is equivalent to 1.0, and we do not need to account for imperfect detection when estimating survival, nor do we have to infer a latent mortality process, as we had to do in the CJS. This means that we can define the survival of birds using the same binomial likelihood function that we used for the coin flip, and we can each bird's data as a sequence of coin flips that determine whether each individual dies, or continues living, during each interval.

However, there are a few complicating factors inherent to estimating survival using telemetry data. First, we may not capture all animals instantaneously. If the capture, collaring, and release of animals occurs through time then there are a different number of animals available to either live or die as new individuals are added. Similarly, as animals die and are removed from the sample, fewer individuals are available to live and die during subsequent occasions. Coming back to the coin flip analogy, these cases, termed **staggered entry** and **staggered exit** (respectively), mean that the total number of coin flips during any interval is likely to change throughout time, and this must be accounted for.

There are a number of different analytical methods available to analyze these data appropriately, and the one we will be using is termed a "Known Fate" analysis. In a known fate

analysis the encounter history is set up slightly different than what we saw for the CJS. Instead of a series of 1s and 0s, which indicated alive or dead status, we now have 3 alternative options for information contained in the history:

10 : This reflects a case where an animal entered an interval alive, and was known to be alive at the end of the interval.

11: This reflects a case where an animal entered an interval alive, and then died during the interval.

00: This reflects the case where the animal's status was not recorded during the interval, typically because it had not yet been captured, or because it died previously.

So, a history that looks like the following:

001010101100

would represent an animal that was originally captured sometime during the second interval, survived intervals 2, 3, and 4, died during interval 5. Because of the death in interval 5, the animal was therefore was not available during interval 6. Notice that now in order to determine the study length, we need to count the total number of digits in the encounter history and divide by 2.

3.2 Overview of the "study system" and data

We monitored the survival of ruffed grouse through time using VHF radio collars that are equipped with a motion-sensitive mortality switch that changes the signal pulse rate when the collar remains motionless for a period of time (6 hours in the case of our collars). By monitoring the signals of each bird, we can determine whether the bird is alive or if it has died on any particular occasion. Once a mortality signal was recorded, we followed up on the ground and confirm that the bird was in fact killed.

The dataset you'll be working with contains an 8-month survival history for 279 radio-collared ruffed grouse that were monitored at three different study areas. The breakdown of sex and age classes in the dataset are as follows: 69 Adult Male; 62 Adult Female; 80 Juvenile Male; 68 Juvenile Female. The purpose of using an 8-month history is that for this study we are interested primarily in fall/winter survival of grouse and the factors that affect their survival during these seasons.

As a first step, we need to clear your R workspace to remove any old models you ran (such as from lab 5) because these will confuse some of our later operations. This will

use the rm(list=ls()) operation, and its important to note this will wipe clean your entire workspace. So, for example, if you have not finished lab 5 it will clear all your objects associated with that lab. The beauty of R is that if that happens, and your code is saved, you can just re-run and be right back where you left off in a matter of seconds. But, you should recognize there is no 'undo' for this command.

We will then reload RMark, and also clean up some of the temporary files from last week that you might have noticed clogging up your folder.

```
library(RMark)

# this command should delete any temporary mark
# files no longer in use from within your working
# directory

cleanup(ask = FALSE)
```

To start with, lets import the data and take a look at its structure:

```
## read in the file as a csv, where the colClasses
## argument tells R the format of each column

RUGR.ch <- read.csv("C:/Users/Erik/Desktop/R/RUGRData.csv",
    colClasses = c("numeric", "character", "factor",
        "factor", "factor", "numeric", "numeric", "numeric"))

## look at the first 15 rows

head(RUGR.ch, 15)

## or view the entire history in a separate window

View(RUGR.ch)
```

Using the head() or view() commands, take a look at the dataset and compare it to both the description of the known-fate data and the description of the dataset above. The first column labeled 'ID' is simply a unique numeric identifier for each individual. Think of this as a band number.

When looking at the capture history, column ch, you will notice that all individuals "enter" the history in the first occasion, and that is because capture for this study occurred during month 1 - September. So we have no staggered entry in this study, only staggered exit. Read through the first few lines of the history and see if you can pick out cases where animals died, versus those that survived the entire 8-month monitoring period.

The next 3 columns reflect categorical variables of the study area (Study.Area; labeled A, B, and C), the age of each bird (AgeClass), and the sex of each bird. In the case of age, a juvenile grouse is one who is in their first year of life, whereas an adult is in their second or later year of life.

Finally, we have three columns labeled DS, CS, and CT. These are numeric variables that represent measured habitat characteristics within the birds' home ranges. We will come back to these in next week's lab.

3.3 Running a model

Alright, lets setup our RMark analysis and run a model or two, then come back to the linear model concepts from earlier in the lab. As we did last week, first we will process the data and build the design data object.

```
# these are the same commands applied in lab 5

RUGR.process <- process.data(RUGR.ch, model = "Known",
    nocc = 8, groups = c("Study.Area", "AgeClass",
        "Sex"))

RUGR.ddl <- make.design.data(RUGR.process)
```

And from there, we can run a model. To start with, we will run a basic model that asks whether there is a difference in survival among the 3 study areas, using the grouping variable 'Study.Area'.

```
S.Study.Area <- mark(data = RUGR.process, ddl = RUGR.ddl,
    model.parameters = list(S = list(formula = ~Study.Area)),
    brief = TRUE, silent = TRUE)

summary(S.Study.Area)
```

We can then use the summary() command to look at the results.

```
summary(S.Study.Area)
```

```
## Output summary for Known model
## Name : S(~Study.Area)
##
## Npar :  3
## -2lnL:  1101.332
## AICc :  1107.347
##
## Beta
##                  estimate        se        lcl       ucl
## S:(Intercept)  2.1775885 0.1668319  1.8505981 2.5045790
## S:Study.AreaB  0.0094837 0.2113884 -0.4048376 0.4238051
## S:Study.AreaC -0.1439566 0.2114506 -0.5583998 0.2704866
##
##
## Real Parameter S
##                                                        2         3         4
## Group:Study.AreaA.AgeClassAdult.SexFemale    0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexFemale    0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexFemale    0.8842832 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassJuvenile.SexFemale 0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexFemale 0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexFemale 0.8842832 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassAdult.SexMale      0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexMale      0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexMale      0.8842832 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassJuvenile.SexMale   0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexMale   0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexMale   0.8842832 0.8842832 0.8842832
##                                                        5         6         7
## Group:Study.AreaA.AgeClassAdult.SexFemale    0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexFemale    0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexFemale    0.8842832 0.8842832 0.8842832
```

```
## Group:Study.AreaA.AgeClassJuvenile.SexFemale 0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexFemale 0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexFemale 0.8842832 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassAdult.SexMale      0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexMale      0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexMale      0.8842832 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassJuvenile.SexMale   0.8982188 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexMale   0.8990826 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexMale   0.8842832 0.8842832 0.8842832
##                                                       8         9
## Group:Study.AreaA.AgeClassAdult.SexFemale    0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexFemale    0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexFemale    0.8842832 0.8842832
## Group:Study.AreaA.AgeClassJuvenile.SexFemale 0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexFemale 0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexFemale 0.8842832 0.8842832
## Group:Study.AreaA.AgeClassAdult.SexMale      0.8982188 0.8982188
## Group:Study.AreaB.AgeClassAdult.SexMale      0.8990826 0.8990826
## Group:Study.AreaC.AgeClassAdult.SexMale      0.8842832 0.8842832
## Group:Study.AreaA.AgeClassJuvenile.SexMale   0.8982188 0.8982188
## Group:Study.AreaB.AgeClassJuvenile.SexMale   0.8990826 0.8990826
## Group:Study.AreaC.AgeClassJuvenile.SexMale   0.8842832 0.8842832
```

From top to bottom, the summary statement is showing us the likelihood and AICc values for the model, then it is printing a series of 3 "beta" (i.e. β) values for the model, followed by the 'Real Parameters'. The real parameters should be familiar from the last lab - they are the estimated survival probabilities for each group of birds, given this particular model structure. But the beta values are a new result, and I want us to take some time to unpack these a bit further because they relate to the linear modeling concepts above.

First, lets extract just the beta coefficients as their own object to make them a bit easier to work with

```
S.StudyArea.beta <- data.frame(S.Study.Area$results$beta)

S.StudyArea.beta
```

The three rows in the result correspond to the 3 β parameters in the model we just build, and they are how the model will delineate the difference in survival probabilities between each of the three study areas. This week, we're going to focus exclusively on the values in the first column, which are the actual estimates of β that the model derived from the data. Next week, we will focus a bit more on the other 3 columns that represent the SEs and 95% CIs of the β estimates.

Lets look back to equation 6 from earlier, which is the exact same model form as what is represented in our Study Area model. If we wanted to construct the linear model for each, study area, it would look something like the following:

$$logit(S_A) = 2.1775885 + 0.0094838 * 0 - 0.1439565 * 0 \tag{10}$$

$$logit(S_B) = 2.1775885 + 0.0094838 * 1 - 0.1439565 * 0 \tag{11}$$

$$logit(S_C) = 2.1775885 + 0.0094838 * 0 - 0.1439565 * 1 \tag{12}$$

Carefully compare each of the three equations above with the first column of the beta coefficient outputs. You will see the only difference in each equation is the combinations of dummy variables - i.e. - which 1s and 0s are applied to which beta coefficients.

Qualitatively, this allows us to start to understand the differences in survival between the three study areas. Study area C, with a negative beta coefficient, has a lower survival probability than Study Area A, which is defined just by the intercept. Study Area B, in contrast, has a positive beta, suggesting greater survival of grouse in those areas compared to Study Area A. By extension, we would also infer that grouse in area B also have greater survival than grouse in area C.

Now that we've explored the βs a little bit, we need to step back into the math in order to understand how they allow us to derive the actual survival probabilities, which is what we were after to begin with.

3.4 The Logit Link Function

In a simple linear regression, the response variable y is assumed to be a continuous value with a normal distribution that can take on both positive and negative values. Because of

these properties, when we run a linear regression, we can apply simple math using equation 2 to directly estimate an expected value of y for a given value of x. In survival estimation, however, we do not have that luxury, because the data we collect are generally **Bernoulli Trials** with discrete outcomes that are determined by an underlying probability structure. As a probability value, a survival estimate by definition cannot be less than 0.0 and it cannot be greater than 1.0. Said differently, you can't have fewer than 0% of animals dying, and there is no way for more than 100% of them to survive. Because of this we have to place an additional structure on the model so that it's predictions will be constrained to fall between 0 and 1. This is done by specifying the model within a **link function**. There are a number of different link functions that are possible in MARK, but for simplicity sake we will focus on the **logit link**, which is the most commonly used:

$$S = \frac{\exp(\beta_0 + \beta_1 * x_B + \beta_2 * x_C)}{1 + \exp(\beta_0 + \beta_1 * x_B + \beta_2 * x_C)} \tag{13}$$

This equation might seem somewhat intimidating at first glance, but with a little careful attention we can unpack its components. Here, exp() is the exponential function based on Euler's number (or e^x). You should notice that the basic linear equation provided from equation 6 is contained as the exponent term in both the numerator and denominator of the equation. By working in the logit link function, the parameter coefficients (β) will be estimated by maximum likelihood such that the response variable (survival, recapture, or some other probability value) will always be bounded by (0,1). In equations 10, 11, and 12, you'll see I denoted this by listing the response variable as $Logit(S)$, to indicate the equation should be solved under a logit link.

To illustrate how the beta coefficients from our survival analysis relate to the actual estimates of survival probability for our marked birds, let's go through a quick exercise using Microsoft Excel to calculate survival probabilities from the beta estimates of this model.

1. First, export the beta coefficient table to Excel. Hopefully this is old-hat for you by now with past week's labs, but remember you will need to revise the file pathway in the following line of code:

```
write.csv(S.StudyArea.beta, "C:/Users/Erik/Desktop/R/RUGRBeta.csv")
```

2. Navigate to the folder you stored the output in, and open it in Microsoft Excel. In column F of this new spreadsheet, give the column a header labeled Study Area, and create a sequence of values for A, B and C in cells F2, F3, and F4.

3. In column G, next to your labels for Study Area, reconstruct the logit link equation (equation 13) for the Area A survival probability using it's corresponding linear model (equation 10). In completing this step, you should build a formula in Excel that links to the correct values of the beta coefficients in the spreadsheet, and fills in the correct x values (0) as shown in equation 10. If you did this correctly, you should find that it produces an estimate that is <1.0 but >0.0, i.e., it is a probability value. If your result falls outside the 0 to 1 range, you've got an error in your formula (hint - be sure to pay attention to your order of operations!).

4. Once you are returning a probability value, repeat the formula for Study Areas B and C. You should be able to use the same formula you did for A, keep the values for the beta coefficients anchored, and just change the sequence of dummy variables (ones and zeros) to match equation 11 and equation 12 for study areas B and C respectively.

5. Next, you should check the resulting estimates from your formula against the 'Real Parameter Estimates' from RMARK. You can obtain these as follows, which is review from last week:

```
S.StudyArea.real <- S.Study.Area$results$real

S.StudyArea.real
```

6. You should find that the real survival estimates from the model, and those you calculated based on the beta estimates, are identical out to at least the 6th decimal place or so. This is because when we build models in RMark under a GLM framework, Program MARK is working behind the scenes to use Maximum Likelihood to estimate the beta coefficients, rather than the probabilities directly. Then, it does the math to reconstruct the logit link and report the probability value for the estimated betas, just like you've just done in Excel. Save your completed spreadsheet, as you'll turn it in as part of your assignment this week.

7. Before we leave the logit link, I want to take a moment to illustrate the utility of R and reinforce the concept that just about anything you can do in Excel we can replicate in R. In this case, lets repeat our Excel exercise of reconstructing the survival probabilities, but do it using command line in R. I will get you started, and we will be working with the S.StudyArea.beta object we made earlier.

8. First, we will add a new column to our dataframe for each study area, and fill it with labels for A, B and C:

```
S.StudyArea.beta$Study.Area <- c("A", "B", "C")
```

9. Now, we will create an empty column in the dataframe for our survival predictions - the 'NA' in this following code just signifies that the column is empty for the time being

```
S.StudyArea.beta$Survival <- NA

S.StudyArea.beta
```

10. Just as with Excel, we can now write an individual equation to populate each of the three values for Survival. We will start with the survival for birds in study area A, which sits in position 1 of the new column

```
# here placing the [1] beyond survival says to fill
# in just the first element of the column

S.StudyArea.beta$Survival[1] <-
exp(S.StudyArea.beta$estimate[1])/(1 + exp(S.StudyArea.beta$estimate[1]))

S.StudyArea.beta
```

11. Compare this piece of code to the first cell in your Excel spreadsheet - it is executing the same process, and produces the same result, but for convenience notice that I ignore β_1 and β_2, which would be multiplied by 0 and thus canceled out. Lets try the next value, Study area B, which sits in the second row of the spreadsheet.

```
S.StudyArea.beta$Survival[2] <-
exp(S.StudyArea.beta$estimate[1] + S.StudyArea.beta$estimate[2])/(1 +
    exp(S.StudyArea.beta$estimate[1] + S.StudyArea.beta$estimate[2]))

S.StudyArea.beta
```

12. The equation has gotten a bit clunkier, and you should notice I am still dropping β_2 from the equation. Using the above code as your starting point, replicate the code and

fill in the last estimate for study area 3 - you should find it matches both your Excel-based estimate and the one produced by RMark. Be sure to include this completed sequence of code when you hand in the R file for this lab's assignment.

4 Bringing it all together

We've now worked through the process of evaluating survival in a known fate analysis under a GLM framework. Using the code-based implementation of RMark, this isn't really much different in practice than what we did last week under the CJS analysis. But the goal of the lab so far has been to introduce you, conceptually, to building a survival model in a GLM framework. We've estimated beta coefficients that are used to derive real survival probabilities under the logit-link function. These are obtained based on principles of maximum likelihood, such that the resulting survival probabilities are the most likely values, given the structure of our model and the radio-telemetry data we collected.

If you can see these connections, at least in concept, you've gone a long way towards understanding the process of demographic estimation more generally. A major advantage of working in a GLM framework will come next week, as it will allow us to add an extra dimension of statistical inference to our analyses and hypothesis testing.

But for the time being, we need to bring this lab full circle because, despite all the time we've spent working to understand the math behind the models, we haven't actually asked any questions yet! Also if you were paying attention, you probably noticed that the ACTUAL differences in monthly survival probabilities between the three study areas were pretty trivial. The Study Area model was a useful example to illustrate the concepts we covered, but we haven't yet asked if provides any useful or meaningful information about ruffed grouse survival. To do that, we'll need to run a series of competing models that we can evaluate with AICc. This will largely be review from last week's lab, as we'll ask RMark to run a series of models with additive structures, and use collect.models() to give us an AICc ranking of those models.

For this week, we will focus on the categorical effects in the dataset, and address the following central questions:

1) Are there times during the fall/winter when ruffed grouse have lower survival?

2) Are young grouse (juveniles) more vulnerable to mortality than adults?

3) Are female grouse more vulnerable to mortality than males?

4) Is there spatial variability in survival - i.e. - do grouse in some study areas survive at a higher rater than others.

Importantly, each of these questions are unique and do not represent alternative hypotheses, so to fully evaluate them in a multi-model comparison, we will need to consider a larger combinations of model structures:

Model
S(Month + Sex + Age Class + Study Area)
S(Month + Sex + Age Class)
S(Month + Sex)
S(Month+ Age Class)
S(Month + Study Area)
S(Month)
S(Age Class)
S(Sex)
S(Study Area)
S(.)

Luckily, building and running even a large number of models is fairly easy in RMark. Lets start with the most general model, that of S(Month + Sex + Age + Study Area), and run it using the mark() command.

```
S.Month.Sex.Age.Study.Area <- mark(data = RUGR.process,
    ddl = RUGR.ddl, model.parameters = list(S = list(formula = ~time +
        Sex + AgeClass + Study.Area)), brief = TRUE,
    silent = TRUE)
```

Remember that RMark uses the term 'time' to represent each interval of a study, which in this case is a 1-month interval length. Thus the variable 'time' in the formula= statement is synonymous with month, and allows for an independent estimate of survival for each month between September-April. To run the second model on our list, S(Month + Sex + Age), we can run exactly the same code, but remove the Study.Area variable and also rename the model.

```
S.Month.Sex.Age <- mark(data = RUGR.process, ddl = RUGR.ddl,
    model.parameters = list(S = list(formula = ~time +
        Sex + AgeClass)), brief = TRUE, silent = TRUE)
```

From here, I want you to replicate this code, but run each of the additional models listed in the table above. Accomplish this in practice by pasting copies of the code into your RStudio script so that you have one run of the mark() command for every model. Be sure to change both the object name (e.g. S.Month.Sex.Age) AND the model formula for each.

You should remember from lab 5 that the S(.) is the 'null' model, which reflects a scenario where there is no variability in survival. Here you will be you'll be fitting an intercept-only model, which in R we accomplish using ~1 in the model.parameters= component of the code (substitute formula=~1).

Once you've setup your models and run them, recall that you can obtain your AIC table using the collect.models() command. See Lab 5 if you need a reminder on how to export your AIC table for completing the assignment below.

```
RUGR.AIC <- collect.models(type = "Known")

RUGR.AIC
```

> ***Sidebar - A quick note about the collect.models() command -*** This command searches your R workspace for mark() objects, and returns AICc information for all of them. If you have models in your workspace from past analyses (like, lab 5) it will not know that you don't want to those models run. One option, as I specify above, is to include 'type=Known' in the statement to let it know to only collect Known Fate analysis models. Alternatively, you can use the command rm(list = ls()) to clear your R workspace before you start a new analysis, as we did at the beginning of this lab. Just realize this will clear everything you've got in the workspace, not just your RMark models.

5 The Three Big Things You Should Have Learned Today

FIRST Extending what you've already learned about linear regression to linear models as a flexible modeling framework.

SECOND How we can apply linear models to discrete variables by the use of unique combinations of β coefficients and dummy variables.

THIRD How to use a logit link function to constrain a linear model to produce probability values as a model outcome.

6 Lab 6 Assignment

For the assignment for this lab, please submit a word document that includes an AIC table showing your model selection results. You don't need to worry about formatting this, we just need a table in the word document that clearly demonstrates you have completed the exercise. In addition, please provide a brief answer to each of the following questions:

1) Was there support for an effect of individual age on survival?

2) During which time period(s) was the monthly probability of survival the lowest?

3) What did you find for the survival probability of juvenile males during December? Describe this value in a way that highlights the biological interpretation of the number.

4) What was the probability that a female ruffed grouse would survive the entire 8-month study period? For this question, you will want to reference your notes from WLE 410 lecture materials.

In answering the questions, be sure to use evidence from the model results (AIC, real parameter estimates, or beta coefficients) as appropriate to support your answers. Please also include your completed R script, and the Excel file containing your survival estimates as a component of your assignment.

You will probably want to look back at lab 5 to remind yourself how to export the real parameter estimates (the Survival probabilities in this case) to help you with answering questions 2, 3, and 4. Also remember that we use both AIC as a hypothesis testing tool as well as interpreting the actual probability values.

Extra Credit: If you completed the extra credit assignment described below, in addition to turning in your Excel file, please also paste a table of your results (survival estimates for each age class and month) into your word document. You do not need to worry about formatting this table, but you should make sure it is readable.

7 Extra Credit Opportunity - Linear Models for Multiple Effects

There's one more area I'd like to cover with respect to linear models, and that is how they can work to give us estimates of survival probability when more than 1 particular categorical effect is in the model. For example, when you have effects of both time and sex in the model, how do you derive an estimate of survival for a female in time period 2? We've covered a lot in the lab up to this point, so if you feel a bit fatigued you can stop now and skip to the lab assignment and complete it for up to full credit. But, read forward for an optional opportunity for up to 3 extra credit points.

The example I give above based on the Study Area variable and equation 6 gives us a means of modeling survival when each animal belongs to one discrete group. Animals are either found in Study Areas A, B, or C, but only one of these three options. A number of our models, however, place animals into multiple categories simultaneously. Lets use the S(time + AgeClass) model as an example, which you should have found fell within 2.0 δAICc of the best model. In this case, for a particular month, say October, and a particular age class, say juveniles, in order to derive a survival probability, we need to reconstruct the model that includes both the effect of October and the effect of being in the juvenile age class. The simplification of the linear model that represents this case is:

$$Logit(S_{Sep,Juv}) = \beta_0 + \beta_{Sep} * 1 + \beta_{AgeClass} * 1(\#eq : eq14) \tag{14}$$

where we include the beta for the month of October and the beta for age, where a dummy variable of 1 indicates a juvenile. If we wanted to calculate the survival probability for an adult, we use the same equation but we substitute a 0 for 1 for the AgeClass dummy variable.

$$Logit(S_{Oct,Juv}) = \beta_0 + \beta_{Oct} * 1 + \beta_{AgeClass} * 0(\#eq : eq15) \tag{15}$$

Which, given the 0 multiplied by $\beta_{AgeClass}$, would simplify to

$$Logit(S_{Oct,Juv}) = \beta_0 + \beta_{Oct} * 1(\#eq : eq16) \tag{16}$$

So we can see that the difference in the estimated survival for adults vs juveniles during October differs based on whether or not the effect of juvenile membership, represented by $\beta_{AgeClass}$, is applied to the estimate. For any given combination of two or more variables,

we can repeat this process by including the relevant combination of beta coefficients to the model.

For the extra credit assignment, I would like you to use these principles to recreate the Excel-based exercise we conducted earlier, but do so using the beta coefficients from the S(time + AgeClass) model. From those betas, create a spreadsheet that uses the logit-link equation equation 13, but correctly modified to calculate the monthly survival probabilities for each age class (juvenile and adult) and month (Oct through Apr) combination.

As a specific hint, I will let you know that September in that model is represented by the Intercept term only, and the beta associated with 'time3' is October, 'time4' is November, and so on. As stated earlier, for AgeClass, juvenile grouse receive a dummy variable of 1.

To receive extra credit, turn in a completed Excel sheet titled 'Lab 6 Extra Credit' along with the word document described below.

8 Lab Appendix

9 Quick reference for important R commands in this lab

Below are a few key commands you will find useful in completing your assignment. You should also make reference, if necessary, to the Quick Reference guide provided at the end of Lab 5.

Output the beta parameter estimates from a particular model, create a data frame for that table, and export it as a csv file to access in Excel. Note you'll need to change the file path in the write.csv statement and the "Model_Name" to the correct name for the model object you ran:

```
# Return and view the beta parameter estimates of a
# model object


Model_Name$results$beta

# create dataframe of estimates
```

```
Survival.Estimates <- data.frame(Model_Name$results$real)

# write to csv

write.csv(AIC.Table, file = "C:/Your Path/Subdirectory/subdirectory/RealEstimates.csv")
```

View an overall summary of a model object, which will include it's AIC scores, beta coefficients, and real parameter estimates. Note that if you want to view the real parameter estimates with their SEs and CIs, you will need to extract the results individually (see Lab 5 Quick Reference)

```
# Return a full summary of a model object

summary(Model_Name)
```

To access just the beta coefficients, create a new object that extracts just the betas from the model results.

```
Model_Name.beta <- data.frame(Model_Name$results$beta)

Model_Name.beta
```

Access an AIC table for all models you have run, create a dataframe for that table, and export it as a csv file to access in Excel. Note you'll need to change the file path in the write.csv statement:

```
# pull the models together

AIC.Table.Object = collect.models(type = "Known")

# print the AIC table

AIC.Table.Object

# create a dataframe of the table
```

```
AIC.Table <- data.frame(AIC.Table.Object$model.table)  ## extract the table as a datafr

# write to csv

write.csv(AIC.Table, file = "C:/Your Path/Subdirectory/subdirectory/AICTable.csv")
```

10 For more information

10.1 Background materials

For complete understanding, I recommend that you review Chapter 6 in Powell and Gayle on linear models, which will be slightly repetitive (hopefully in a good, totally reinforcing way) to some of the background materials below. This chapter will cover components relative to both this week's lab and next.

For the known fate component of this analysis, you can also see Powell and Gayle's Chapter 9 which provides background on survival estimation via known fate.

10.2 More advanced resources

A deeper dive on linear models in a demographic estimation framework can be found in Chapter 6 of The MARK Manual.

Similarly, see Chapter 16 for much more detail on the Known Fate.

For a discussion of alternative approaches to survival estimation from radio-marked animals, see Chapter 6 of Murray and Sandercock, *Population Ecology in Practice*

11 GLOSSARY OF KEY TERMS

Bernoulli Trials

Term for an event with two possible outcomes, defined by a binomial probability

Continuous Predictor Variable

An independent variable measured on a continuous (i.e. numeric) scale

Discrete Predictor Variable

An independent variable comprised of discrete categories or groups.

Dummy Variable

Term for a sequence of 1s and 0s used to deliniate group membership. For discrete variables with only 2 levels (e.g., heads or tails), a single dummy variable is required to define membership in each. For >2 variables, group membership can be signified with combinations of N-1 dummy variables.

Generalized Linear Model

A linear model form that accommodates response variables with error distributions other than a normal distribution

GLM

Abbreviation for Generalized Linear Model

Link Function

A mathamatical function imposed to a GLM that constrains the response in some way, such as bounding it between 0 and 1.

Logit Link

A link function that constrains a model outcome to fall between 0 and 1, often used to represent a binomal probability outcome for GLMs

Staggered Entry

In survival estimation, the act of individual animals entering a capture history for the first time during different occasions.

Staggered Exit

In survival estimation, the act of animals exiting a capture history (either through mortality or continued failed detection) at different time intervals.