Survey Designs for Distance Sampling: A Study of Zebra Mussels

Alana Danieu, Nick Fredrickson, Emily Kaegi, Clara Livingston

Carleton College
Senior Integrative Exercise
Advisor: Katie St.Clair
February 22, 2018

Abstract

The zebra mussel (*Dreissena polymorpha*), an invasive species from Europe, has spread throughout the Great Lakes and down the Mississippi River, infesting hundreds of lakes across North America, including Minnesota. Our group worked with researchers at the University of Minnesota to help develop the best lake sampling methods to accurately determine the abundance of zebra mussels in infested lakes. Using lake survey data from the University, we implemented the principles of line transect distance sampling to fit a model that describes the probability of detection for each zebra mussel in a given lake. Then, using the Horvitz-Thompson estimator, we made inferences about the population abundance of zebra mussels in the infested lake. Once we had a better understanding of estimating abundance in a single lake, we explored which survey design would most accurately predict the population abundance and reduce error. Because we only had data from one survey design, we took advantage of a package in R called DSsim, which allowed us to run simulations on generated populations. We constructed different survey designs under various conditions (including population size, number of transects, and hotspot presence) to investigate if and how the results would change. Finally, we conducted an experiment to analyze the relationship between time on a transect and detection. By evaluating the error, bias, and predictive statistics provided by the simulation and model fitting, we are able to provide ecologists with a recommendation for the optimal zebra mussel sampling design.

Acknowledgements

We would like to thank our advisor Katie St. Clair for her knowledge and guidance leading us through this project, as well as Dr. John Feiberg and Dr. Jake Ferguson from the University of Minnesota for all their expertise and recommendations. We would also like to thank Mike Tie for his assistance in running simulations as well as all the participants of our experiment. Finally, thank you to all our family and friends for their support.

1 Introduction

Zebra mussels *(Dreissena polymorpha)* are small freshwater shellfish, named for the zebra-striped pattern found on their shell. Native to the Black, Caspian, and Azov Seas, zebra mussels were most likely introduced into Lake Erie in the late 1950s as a result of the opening of the St. Lawrence waterway (Hebert, 1989). Zebra mussels have managed to spread throughout the Great Lakes and down the Mississippi River, infesting hundreds of lakes across North America. Able to survive out of water for several days, they attach themselves to the bottom of boats and spread to new lakes via roadways on boat trailers (USGS). During their reproductive cycle, a single female zebra mussel can lay over one million eggs a year, making any lake with even a couple zebra mussels in immediate danger of infestation (Virgina DGIF).

Nonnative species are introduced into ecosystems and habitats all the time. The invasive zebra mussels, however, have a particularly detrimental effect on North American freshwater ecosystems. In many infested lakes, zebra mussels have become so pervasive that they easily out-compete other species for nutrients. Since the introduction of zebra mussels, zooplankton biomass has substantially decreased, impacting food-web interactions and reducing biodiversity (Miller 2007). Similarly, the zebra mussel has strained the survival of the native mussel, *Amblema plicata*. In some lakes, zebra mussels have even caused the extirpation of *A. plicata* (Hart 2001). As zebra mussels filter water, they remove particles, increasing the water clarity (Qualls 2007). At first, this effect may be viewed as beneficial. The increased water clarity, however, encourages extensive algae growth, particularly of the nuisance algae, *Cladophora glomerata* (Limburg 2010). These algal blooms can lead to dead zones in lakes and foster the growth of toxic bacteria (Vanderploeg 2001).

To mitigate the effects of this invasive species, managers and scientists need an accurate estimation of the population of zebra mussels in a given lake. Lakes that contain too many mussels are very difficult to treat, while lakes with too few mussels are not worth the extensive effort it would require to eliminate the population. For an infested lake to be an ideal candidate for treatment, the population of zebra mussels must between these two levels of infestation, a "sweet spot." To accurately estimate a population, there must be a standardized method of surveying these lakes. Our research investigated the strengths and weaknesses of different survey designs. To answer the question regarding proper sampling methods, we took advantage of technology and ran simulations of different survey designs, as we could not experiment with sampling designs within lakes. We used data of already surveyed lakes from a research group at the University of Minnesota, which we fit models to estimate the population of that lake. We then used these estimates to build fake populations of zebra mussels for simulations of different survey designs and compared results. Another question of interest was the cost of sampling these lakes. One major factor in cost is the time spent on a transect. To determine how time would affect detection, we ran an experiment to mimic sampling zebra mussels and were able to draw conclusions on the relation of time and the accuracy of predicting a population. In this paper, we will walk through the steps we completed to achieve this task and discuss our findings.

2 Gathering our Data

Our data was gathered as part of a research project led by Dr. John R. Feiberg and Dr. Michael McCartney at the University of Minnesota, funded by the Minnesota Aquatic Invasive Species Research Center. Due to the lack of standardized sampling methods for zebra mussels, their project focused on developing survey methods for estimating population and abundance.

The research team created a survey design using distance sampling techniques which a team of divers implemented in 12 lakes throughout central Minnesota. In this section, we will give a brief overview of distance sampling and summarize the data that was used for the rest of this project.

2.1 Distance Sampling

The majority of this paper focuses on distance sampling. Specifically, we were interested in a design using line transects, a common sampling technique for investigating wildlife densities. In such a design, observers set up a series of line transects, usually using a systemic design. Then, the observer travels along the transect, whether it be by walking, driving, flying, or in our case, swimming, and count the total number of animals detected as they travel the line. For each detection, the observer records the perpendicular distance to the animal from the transect and the distance of the animal along the transect. Other variables can also be recorded, including time of detection, habitat, size of animal or cluster of animals, number of animals in a cluster, and type of substrate on which the animal was found.

Transect sampling operates under four main assumptions:

1. Animals are distributed independently of the transects.
2. Objects on the transect are detected with certainty (i.e. 100% of the time).
3. Distance measurements are exact.
4. Objects are detected at their initial location.

To fit the first assumption, the animals, or in our case mussels, are distributed on the lake floor independent of the transects. Therefore, as an observer looks within the area of the survey, the mussels should be evenly and independently distributed across the transects. The second assumption indicates that when swimming on the transect, divers must see *all* of the mussels located on the line. The third and fourth assumption state that the mussels found are measured exactly in the original location where the observer finds it.

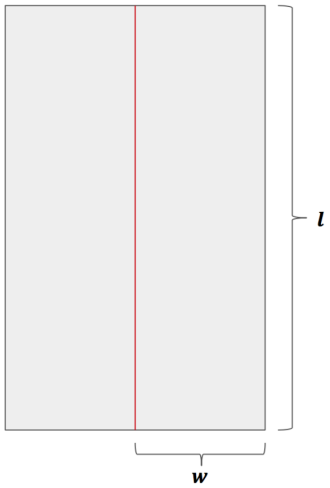


Figure 1: Visualization of a single line transect. Area (a) shown in gray

Before moving in to the mathematics of distance sampling, there are important terms to define. First, we have w, which is the truncation distance on each side of the transect. Next, l is

the length, or effort, of the transect. The area of the surveyed region can be defined as $a = 2wL$. L is the sum of all the transect lengths ($L = \sum_{i=1}^{K} l_i$), where K is the number of transects in the survey, and A represents the area of the region of interest. Generally, this region is the area of the entire lake. The relationship between some of the parameters is visualized in Figure 1.

2.2 Our Data

Of the twelve lakes surveyed by the University of Minnesota, only two produced adequate data for analysis. These lakes, Lake Burgan and Lake Sylvia, are both located in central Minnesota near the town of Alexandria. Each survey consisted of 24, 30 meter transects systematically placed around the lake, perpendicular to the shore. The transects are arranged corresponding to three "zones" of high, medium, and low infestation. Because it is thought that more mussels live in an area called the "infestation zone." To compensate for this, the team surveyed more heavily in this area by placing down 8 transects 3 meters apart. Next, the team placed 3 transects 3 meters apart 150 meters to the left and right of these infested regions, the medium area. Finally, they placed the remaining 10 transects evenly around the rest of the lake, which is believed to have less mussels.

The divers were instructed to swim directly above the transect line, recording every cluster of zebra mussels they encountered. When they found what they believed to be a zebra mussel cluster, they swam out, inspected what kind of substrate it was on, the number of zebra mussels in the cluster, the distance perpendicular to the transect, and various other variables. We used the Lake Burgan data for the bulk of our analysis because the research team believed it is the most accurate representation of an infested lake. This data set contained 52 observations of zebra mussels, on 18 transects. Additionally, only one observation recorded the cluster to have two mussels, while all other observations were "clusters" of one. A visual representation of the transects in the lake can be found in Figure 2.

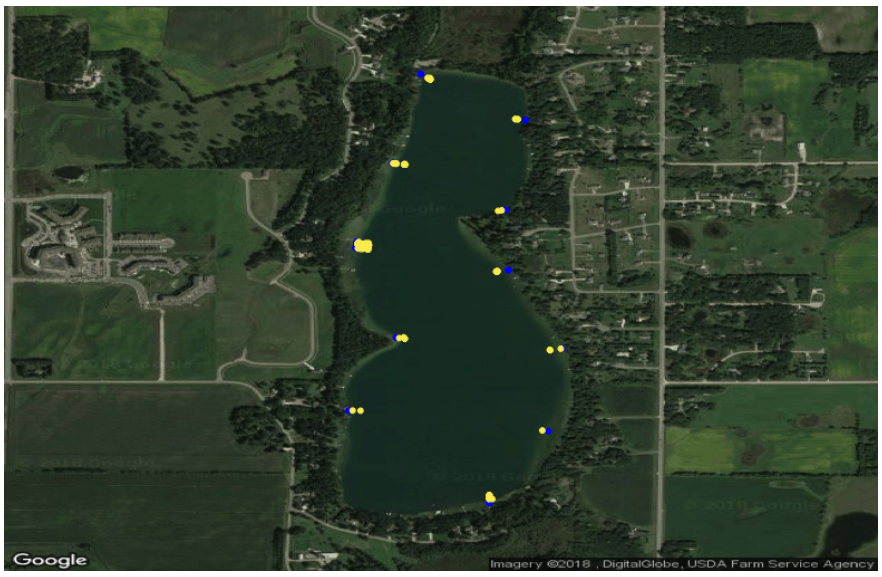


Figure 2: Lake Burgan and locations of transects

3 Estimating Mussel Abundance and Density

After collecting the data, we needed a way to estimate the abundance and density of the surveyed region. This was a two step process. First, we estimated the probability of detection for a mussel using our data and then calculated the abundance of the region. To estimate a detection probability, we fit our data to a model using maximum likelihood estimation to fit the parameters for the distributions of interest. Using the parameters from these models, we estimated the total abundance and density of zebra mussels for the lake using a Horvitz-Thompson estimator. In this section, we will outline this process and describe the results of our own data. Note, that most of the mathematics for this section comes from the book *Distance Sampling: Methods and Applications* by Buckland, et al., 2015.

3.1 Different Distributions

To estimate the probability of detection, we begin with an assumption previously stated: objects on the transect are detected with certainty. If x is the perpendicular distance an object is from the transect, then let $g(x)$ define the detection probability with $g(0) = 1$. Now, we assume that as distance increases, $x > 0$, the probability of detection decreases. Three common models are used to describe $g(x)$. The first model, the uniform distribution, assumes perfect detection for all objects within the transect area (all objects have been found within the transect area). While this model is simple and easy to work with, it is unrealistic in most scenarios.

The next model, the half-normal distribution, can be written as

$$g(x) = exp\left[\frac{-x^2}{2\sigma^2}\right], 0 \leq x \leq w, \tag{1}$$

where x is the distance from the transect, w is the truncation distance, and σ is the scale parameter used to change the shape of the curve. The half-normal maintains the $g(0) = 1$ assumption but alters the detection function to account for non-perfect detection as objects are further from the transect. The parameter σ indicates the rate at which the detection decreases.

Finally the hazard-rate model, can be denoted as

$$g(x) = 1 - exp\Big[(-x/\sigma)^{-b}\Big], 0 \leq x \leq w, \tag{2}$$

with the parameters having the same meaning as the half-normal, with the addition of b. This added variable is a shape parameter that offers even more flexibility in changing the structure of the model. These basic models explain the way detection probability changes with distance, with their differences highlighted in Figure 3.

The half-normal and hazard-rate allow for other variables to be added to our model. The parameter σ then can be a function of a vector of covariates, $\mathbf{z_i}$. This relationship can be expressed as $\sigma(\mathbf{z_i}) = exp(\alpha + \sum_{q=1}^{Q} \beta_q z_{iq})$, where Q is the number of covariates in the vector, z_{iq} is the value for the qth covariate for the ith observation, and $\alpha, \beta_1, \ldots, \beta_Q$ are the different coefficient values estimated. Our data was limited, so these covariate additions were not utilized in our analysis.

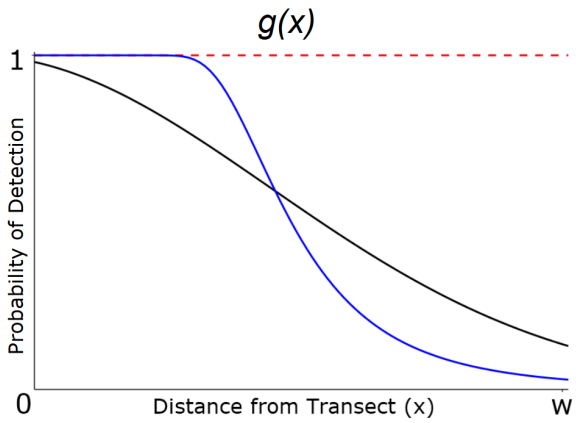


Figure 3: The half-normal curve with $\sigma = 0.5$ (black), hazard-rate with $\sigma = 0.5$ and $b = 5$ (blue), and uniform (dashed red) distribution.

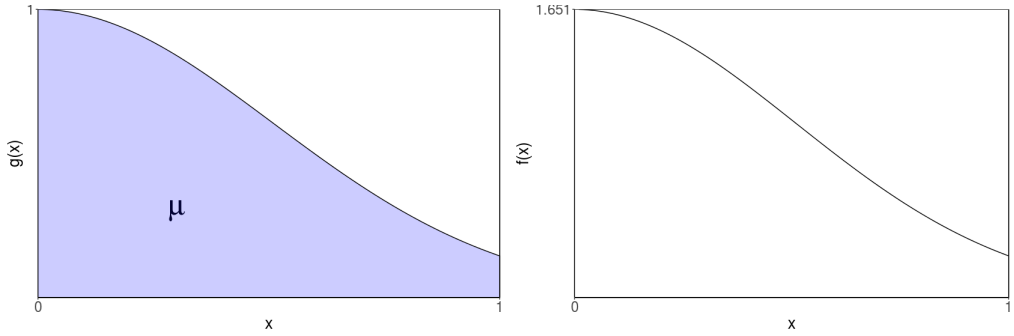
3.2 Estimating Detection Parameters

From these detection distributions, we can fit a function $g(x)$ to our recorded distances and use maximum likelihood estimation (MLE) to estimate σ. We need a proper probability density function $f(x)$ for the observed distances, however, in order to use MLE methods to estimate the parameter σ. Here we define $f(x)$ to be

$$f(x) = \frac{g(x)}{\mu}, \tag{3}$$

with a normalizing constant μ denoted as

$$\mu = \int_0^w g(x)dx. \tag{4}$$



(a) Visualization of $g(x)$ with $\sigma = 0.5$ and $\mu = 0.6$

(b) Visualization of $f(x)$ with $\sigma = 0.5$ and $\mu = 0.6$

Figure 4: Relationship between $g(x)$ and $f(x)$

As Figure 4a displays, the normalizing constant μ is the area under the curve $g(x)$. This constant μ is often called the effective half-width, and turning to Figure 5, we can visualize why.

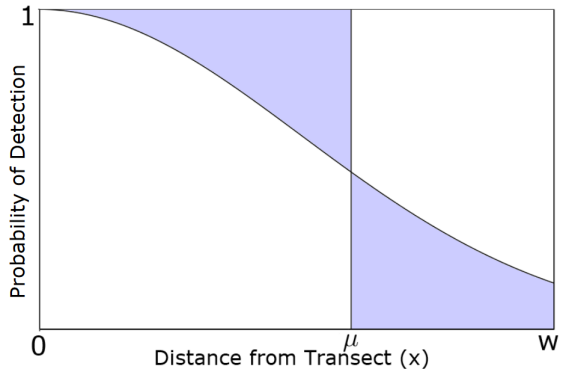


Figure 5: Visualization of μ. $\sigma = 0.508$, $\mu = .606$, and $w = 1$.

If the rectangle is the population of the mussels in the transects, then the area underneath the curve is the proportion of observed mussels. μ is the distance from the transect that divides the population such that the area underneath the curve past μ is equal to the area above the curve before μ. By searching all the transects up to the truncation distance w, we have effectively found all the mussels up to distance μ. Additionally, unless $w = \infty$, μ must be solved numerically.

The functions $f(x)$ and $g(x)$ have the same shape, as they are proportional. To use MLE, we need $f(x)$ to be a probability density function whose area under the curve is 1, which is satisfied by Equation 3. The majority of the data obtained in this study fit a half-normal distribution, so we will discuss that example in more detail. Using a maximum likelihood function, we have

$$L_x = \Pi_{i=1}^{n} f(x_i) = \frac{\Pi_{i=1}^{n} g(x_i)}{\mu^n}, \tag{5}$$

where n is the number of observations and the x_i's are the observed distances. Plugging in the half-normal PDF from Equation 1, we then solve for the $\hat{\sigma}$ that maximizes our function, which is given as

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{n} x_i^2}{n}} \; when \; w = \infty. \tag{6}$$

Because $\hat{\sigma}$ is estimated using maximum likelihood, the standard error is found by using the inverse of the Information matrix, which is ultimately found by taking second derivatives of the likelihood function with respect to the estimated parameter. This estimate is also only true if we let $w = \infty$. Therefore if we add a truncation distance to our design and use Equation 6 to find $\hat{\sigma}$, the result may be underestimated. We utilized the Distance package in R which can take in truncation distances and uses numerical methods to give a less biased estimate for $\hat{\sigma}$ than Equation 6. Using this value of $\hat{\sigma}$ we can derive the value of $\hat{\mu}$ using Equation 4.

Now, we have estimates for both $\hat{\sigma}$ and $\hat{\mu}$ to use within our model. The reason we fit a model for our data is to predict the total population, $\hat{N}$, and density, $\hat{D}$, of our surveyed region. To find these estimates, we need one last parameter, P_a, the expected proportion of objects found within survey. Thus, P_a can be defined as

$$\hat{P_a} = \frac{\hat{\mu}}{w}. \tag{7}$$

In our case, the truncation distance, $w = 1$, so $\hat{P}_a = \hat{\mu}$, making for a simpler analysis. P_a can also be thought of as the proportion of the $1 \times w$ rectangle that is under the curve $g(x)$, or the proportion of the shaded region seen in Figure 4a.

A final parameter of interest is $\hat{f(0)}$, where the $f(x)$ curve intersects the y-axis. Because $g(0) = 1$, we can use the equation $f(x) = \frac{g(x)}{\mu}$ and replace it so that $\hat{f(0)} = \frac{1}{\hat{\mu}}$. In Figure 4b, we see that $\hat{f(0)}$ is 1.651 for the given parameters. We will use $\hat{f(0)}$ later as it pertains to the coefficient of variation for estimated density.

3.3 Estimating Abundance

3.3.1 Horvitz-Thompson Estimator

Non-biased Estimation Using the estimated parameters for the detection model, we can make inferences about our population abundance using a Horvitz-Thompson estimator, given as

$$\hat{N} = \sum_{i=1}^{n} \frac{1}{p_i} \tag{8}$$

In the simplest case of sampling, simple random sampling, a set of n objects is randomly sampled without replacement from a population of N objects. The probability that any given object i is included in the sample is denoted as $p_i = \frac{n}{N}$. Therefore in a simple random sample, using the Horvitz-Thompson estimator, we estimate

$$\hat{N} = \sum_{i=1}^{n} \frac{1}{\frac{n}{N}} = N \tag{9}$$

In simple random sampling, the Horvitz-Thompson estimator is unbiased because p_i is known. However, to implement this estimator for our data, we needed to estimate $\hat{p}_i$, which introduced bias to our estimates.

Biased Estimation In conventional distance sampling, the Horvitz-Thompson-like estimator is written as $\hat{p}_i = \frac{a\hat{g}(x_i)}{A}$ where $\hat{g}(x_i)$ is the estimated detection probability of animal i. Given that the distribution of mussels in the population is uniform, we are able to replace the estimated detection probabilities ($\hat{g}(x_i)$) with an estimate of their mean ($\hat{P}_a$).

To estimate $\hat{p}_i$, we have

$$\hat{p}_i = \frac{a\hat{P}_a}{A} \tag{10}$$

where a is the area of the sample plots and A is the size of the study area. As defined in section 3.2, $\hat{P}_a = \frac{\hat{\mu}}{w}$. If we define a sample area a_k where $k \in \{1, ..., K\}$ the equation can be derived as:

$$\begin{aligned} \hat{p}_i &= \mathrm{P}(Z_i = 1 | i \in a_k) P(i \in a_k) \\ &= (\hat{P}_a)(\frac{a}{A}) \\ &= \frac{a\hat{P}_a}{A} \end{aligned}$$

where Z_i indicates that mussel i in area a_k was detected. Substituting $\hat{p}_i$ back into our equation for $\hat{N}$, we find

$$\hat{N} = \frac{nA}{a\hat{P}_a} \tag{11}$$

We can find density by dividing this estimate by the total area of our study area so $\hat{D} = \frac{\hat{N}}{A}$. Because $\hat{D}$ is a function of area and $\hat{N}$ the estimated density can be defined as follows:

$$\hat{D} = \frac{n}{2wL\hat{P}_a} = \frac{n}{2\hat{\mu}L} = \frac{n\hat{f}(0)}{2L} \tag{12}$$

where L is the total length of all transects. As a side note, some populations of animals (like zebra mussels) are found in clusters. One cluster of mussels may have multiple mussels, usually denoted by s_i, or size. To account for this, we can add in mean cluster size, or $\hat{E}(s)$ to the abundance equations. However, our data from Lake Burgan only had one cluster that has more than one mussel, so we did not investigate the mean cluster size.

3.3.2 Standard Error of Density

When estimating any parameter it is worth understanding how precise our estimates are. These errors can be quantified as standard error or scaled as a coefficient of variation. By using coefficient of variation, we can normalize across simulations or samples to compare precision. Given the primary focus of transect sampling is to gain insight into the true population of an area, the standard error of our density, $SE(\hat{D})$ and coefficient of variation of our density, $CV(\hat{D}) = \frac{SE(\hat{D})}{\hat{D}}$, will provide a quantitative way to understand a sample's precision. Much like the relationship between the estimated density and the total population, the relationship for the standard error can be written $SE(\hat{D}) = \frac{SE(\hat{N})}{A}$.

The function used to quantify $SE(\hat{D})$ is a function of the $CV(n/L)$ and $CV(\hat{f}(0))$ which is expressed as

$$SE(\hat{D}) = \hat{D}\sqrt{[CV(n/L)]^2 + [CV(\hat{f}(0)]^2} = \hat{D}\sqrt{\frac{\frac{K}{L^2(K-1)}\sum_{k=1}^{K} l_k^2(\frac{n_k}{l_k} - \frac{n}{L})^2}{(n/L)^2} + \frac{1}{2n}}, \tag{13}$$

where L is the total length of all the transects, K is the number of transects, n is the total number of mussels found, n_k is the number of mussels on the kth transect, and l_k is the length of the kth transect.

The coefficient of variation of $\hat{D}$ provides a way to analyze the precision of estimates from a particular survey given a set of parameters. Keeping l_k fixed and varying n_k as a function of n, we can examine how $CV(\hat{D})$ changes with the ratio of n/K and for varying number of transects. n/K can be described as the average number of mussels found per transect. If each mussel has equal probability of being on a particular transect, n/K can also be written as $\bar{n_k}$. As we can see from Figure 6, $CV(\hat{D})$ decreases as $\bar{n_k}$ increases. In this example, we generated a random multinomial that gives equal probability of a mussel being on a particular transect, k. Keeping all else fixed, with all l_k set at 30 meters, resulting in $L = 30 * K$, we can focus on what causes a change in the $CV(\hat{D})$.

If a goal coefficient of variation for our density estimate for this example were set at 0.05, we could achieve this value multiple ways. If the survey design consisted of 48 transects, we would need to observe an average of about 14.8 mussels per transect to reach $CV(\hat{D}) = 0.05$. Alternatively, if the design consisted of only 8 transects, this average $\bar{n_k}$ would need to increase to about 78.1 mussels per transect. The exact value of n or n_k cannot be fully controlled in reality, but could be influenced by the precision of the investigator. Generally, the more time spent on a transect, the increased detection probability resulting in increased detections, or a higher n. Later

we will discuss the full implication of time as an influence on detection probability and how time alters half-normal parameters such as σ and μ. Thus, when considering reaching a goal precision, there is a trade off between the number of transects used and the number of mussels found for each transect. In a survey, K is determined before the sampling begins, but n is harder to control. As will be discussed later, there are potential solutions to control for the number of mussels found within a particular transect area.

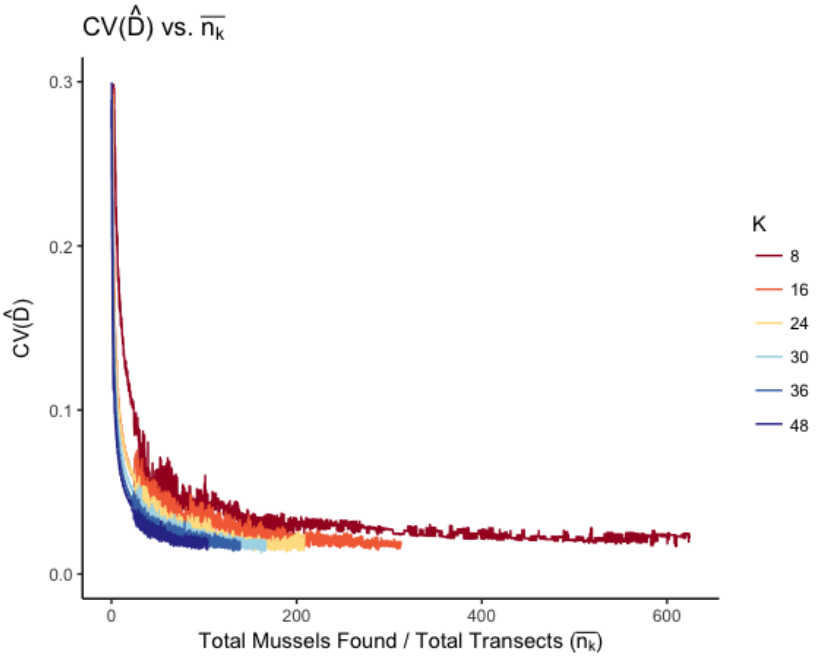


Figure 6: Simulated plots of the coefficient of variance of the predicted detection for varying number of total mussels found over the number of transects used

3.4 Fitting Our Data to a Model

Using the Distance package in R (Miller, 2017), we easily fit our data from Lake Burgan to a model which gave us the MLE estimation of $\hat{\sigma}$ and its standard error. In addition to this, the package also gave us the estimation of $\hat{P}_a$ which can be used to find all the parameters mentioned earlier. We also received estimates of $\hat{N}$ and $\hat{D}$. Usefully, this package is able to take into account transects that did not have any observations in their sample area, which helps give more accurate recordings of abundance.

After fitting the three types of models, we decided to use a half-normal model where the truncation distance was 1 meter. In Table 1, we can see the estimated values for all the different parameters and their respective standard errors and coefficients of variation, CV. Figure 7 displays the fitted model over a histogram of observed mussel distances from the transect. The points on the line are the observed distances that correspond to the estimated probability it is found on the y-axis. By default, the Distance package will give estimations of $\hat{N}$ for the covered region a, or only the area of the given transects, which is why the estimate may seem low. We used the density estimate to calculate $\hat{N}_A$, the estimated population for our region of interest, a 4000×30 meter^2 area representing a strip of the perimeter of the lake.

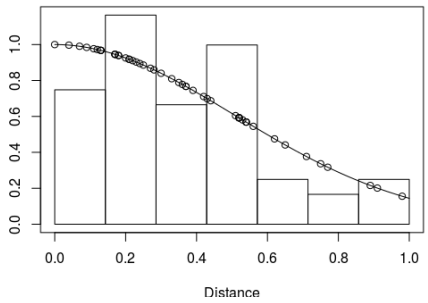


Figure 7: Fitted model over histogram of observed distances

Parameter	Estimate	Std. Error	CV
$\hat{\sigma}$	0.508	0.084	0.165
$\hat{\mu} = \hat{P}_a$	0.606	0.075	0.123
$\hat{f(0)}$	1.651	0.163	0.099
$\hat{D}$	0.090	0.0199	0.222
$\hat{N}_a$	89.584	19.921	0.222
$\hat{N}_A$	10,760	2,392	0.222

Table 1: Estimated Parameters from the Model

4 Simulations

Through our research, we hoped to determine the optimal survey design to predict $\hat{N}$. Because sampling lakes is costly, time consuming, and we only had data from one survey design, we decided to take advantage of a package in R called DSsim, which allows us to run simulations on our own generated populations to see how well different survey designs perform (Miller, 2017).

4.1 Overview of our Simulations

Using the estimates from the Lake Burgan model, we were able to generate our own populations of mussels over an area. We then implemented different survey designs under various conditions to determine if and how the results would change.
The variables we controlled in our simulations:

- Region Size A
- Population Size N
- Number of Transects K
- Detection Scale Parameter σ
- Number of Strata
- Addition of Hotspots (areas of elevated density)
- Number of Iterations B of a Simulation

If we ran a design with multiple strata, we could change N in each strata as well as K. For example, the design used by the University of Minnesota's team used three strata. One area with 8 transects 3 meters apart, two areas with 6 transects 3 meters apart, and a final area with 10 transects spread evenly around the remainder of the lake, for a total of 24 transects.

When running simulations, we decided to change only one variable at a time in order to see how an alteration in design affected the predictions of the model. The starting simulation had $N = 10,000$, $K = 24$, $\sigma = 0.7$, and an area of $4,000 \times 30$ meters^2 (the size of the survey area we can extrapolate from Lake Burgan). We based these values off the Burgan analysis estimates. The estimated $\hat{D}$ of Burgan was .09 as seen in Table 1. If we multiply that density by our $4,000 \times 30$ meters^2 space, we get a population of 10,760 mussels. We decided on 10,000 mussels for the starting N since it was close to the estimate but a simpler number to digest. We also

chose to use a slightly higher σ value in the majority of our simulations because distance sampling expert, Stephen Buckland recommends at least 60 to 80 observations in a sample in order to have more accurate results when estimating model parameters. Therefore, by using a higher σ value we allowed the n values to be above this range.

In each simulation, we changed one of the listed inputs to see how our results were altered. To investigate what would happen if we changed N, we ran simulations with 5 different Ns: 2,500, 5,000, 7,500, 10,000, and 12,500 keeping the other inputs constant. To see the effect of changing σ we ran simulations with values of 0.2, 0.5, and 0.7, keeping other inputs constant. We varied the number of transects K from 24 to 48 to see how more transects would effect our results. We also looked at what would happen if we split up our lake into two strata. In this stratified design, the overall area of the lake was constant, but we broke the region into two equal halves, each $2,000 \times 30$ meters^2. Keeping the overall transect count constant, we ran simulations with more transects in one strata than another. We also looked at how having different densities in the two strata could effect estimates, keeping the overall population N constant. Finally, we wanted to see the effect of adding "hotspots" or areas of more concentrated density to our population. Within our simulations, we could control the size of the hotspot and how dense it was compared to the rest of the region. Because we are not experts on spatial patterns of zebra mussels, we decided on a general hotspot with a density multiplier of 4 with a 30 meter radius.

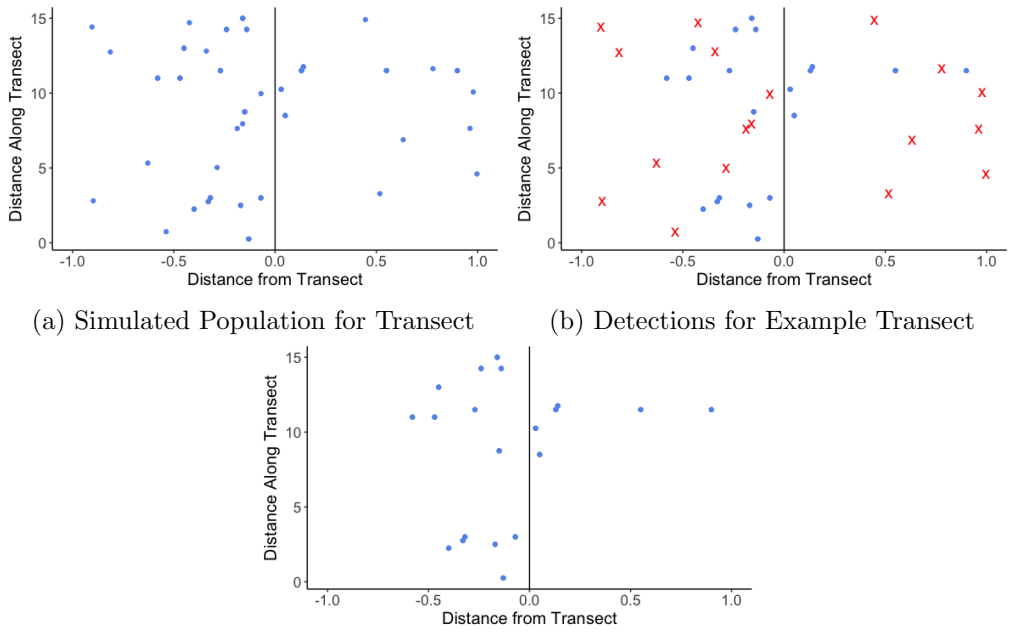
4.2 Simulation Structure

For each iteration of a given simulation, the chosen N objects are randomly placed according to the inputed density around the sample area. K transects are then laid down. The first transect is randomly placed perpendicular to the x axis at a distance between 0 and the spacing of the transects. The remaining $K-1$ transects are then placed evenly from there. For example, if $K = 24$, the first transect will be placed perpendicular to the x-axis between 0 and $4,000/24 = 166.667$. The next transect is placed 166.667 meters from the first transect, the third is placed 166.667 meters from the second transect and so on until K transects are laid down.

Then for each transect, any object within 1 meter of the transect (our truncation distance), is given a probability of detection by plugging in its distance from the transect to a half-normal distribution $g(x)$ with the inputted σ. Once each object has a probability of detection p_i, a Bernoulli distribution is used with success probability p_i to determine whether or not that object was found along a given transect.

Figure 8 shows this process taking place. Graph (a) shows all objects within 1 meter of the transect, graph (b) shows which objects are undetected based on their assigned Bernoulli random value (a red X means it was not observed), and graph (c) shows only the detected objects.

Finally, the observed data is fit to either a half-normal or hazard-rate model where estimates of $\hat{N}$ are calculated, just like the Burgan estimates in section 3.4. This process is then repeated for B iterations. For our simulations, we typically ran $B = 300$ iterations. Figure 9 is an example of an output of one such iteration. We have a given population shown in the upper-left graph, a transect design, shown on the upper-right, the transects with example detections in the bottom-left, and a histogram of the detection distances on the bottom-right.



(a) Simulated Population for Transect

(b) Detections for Example Transect

(c) Detected Clusters for Example Transect

Figure 8: Simulated Population Demonstration

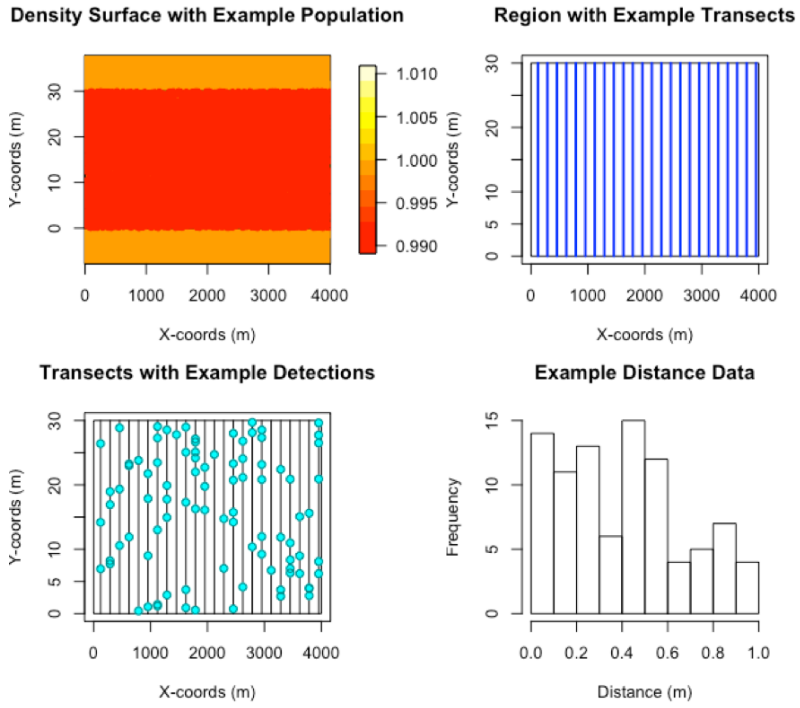


Figure 9: Plots of the population used in an iteration, position of transects, found mussels, and a plot of distance detections

4.3 Simulation Statistics

To compare our sampling design strategies, we looked at $\hat{N}$ estimates. When B iterations are run of our simulation, each realization $b \in \{1, 2, \ldots B\}$ produces an $\hat{N}_b$. We can find the mean of

all the realizations which we will refer to as $\hat{\bar{N}}$, which is our Monte Carlo simulation estimate of $E(\hat{N})$.

$$\hat{E}(\hat{N}) = \hat{\bar{N}} = \frac{1}{B}\sum_{b=1}^{B}\hat{N}_b \tag{14}$$

The first criteria for a good sample design that we considered was accuracy. In other words, how similar were the estimates of our simulation to the actual values ($\hat{\bar{N}}$ vs N)? In our research, we quantified accuracy by measuring bias. Because we are looking at multiple population scenarios in our analysis, percent bias was the most logical measure. The formula to calculate the percent bias of $\hat{N}$ from a simulation is as follows

$$\%\hat{Bias}_{\hat{N}} = \frac{\hat{\bar{N}} - N}{N} * 100\% \tag{15}$$

To compare whether two simulations actually have different values of bias, we needed to look at the simulation standard error for the estimated percent bias.

$$SE_{sim} = \frac{SD(\hat{N})}{\sqrt{B}} * \frac{100\%}{N} \tag{16}$$

Where $SD(\hat{N})$ is the SD of the simulated $\hat{N}_b$ values. This leads into the second way we measured the quality of the sampling design: precision. To measure precision we looked at the SE of our estimates. A smaller SE indicated we had less variability in an estimate. Notice that since we ran the simulations 300 to 500 times, the $SD(\hat{N})$ from the simulations can be used to estimate the $SE(\hat{N})$,

$$\hat{SE}(\hat{N}) = \sqrt{\frac{\sum_{b=1}^{B}(\hat{N}_b - \hat{\bar{N}})^2}{B-1}} \tag{17}$$

We can use Equation 16 to help determine if $SE(\hat{D})$, Equation 12, was a good measure of the true variability of $\hat{N}$. We expected that by running many simulations we would get a spread of $\hat{N}_b$ estimates close to the spread of the actual sampling distribution of $\hat{N}$. Therefore if the mean of the SEs calculated for each simulation, denoted as $\bar{SE}(\hat{N})$, was close to the $\hat{SE}(\hat{N})$, our formulas for SE were accurate estimates of variability.

$$\bar{SE}(\hat{N}) = \frac{1}{B}\sum_{b=1}^{B}SE(\hat{N}_b) \tag{18}$$

Just as we reported the estimated percent bias for $\hat{N}$, we also found the estimated percent bias for $SE(\hat{N})$ which is

$$\%\hat{Bias}_{SE} = \frac{\bar{SE}(\hat{N}_b) - SE(\hat{N})}{SE(\hat{N})} * 100\% \tag{19}$$

The last simulation statistic considered was the coefficient of variation. Because we ran simulations with different values of N, we needed to normalize our standard errors since a $SE = 5$ for a population of 50 is very different than a $SE = 5$ for a population of $5,000$. Therefore, we used estimated CV to compare simulations.

$$CV(\hat{\bar{N}}) = \frac{SE(\hat{N})}{\hat{\bar{N}}} \tag{20}$$

CV let us know how precise our estimates were for each simulation. Therefore we wanted simulations with bias close to 0 and low CV values.

4.4 Simulation Results

4.4.1 Effects of Changing N

The first variable we could change in a simulation was the size of the population N. Table 2 shows how changing the N input effects our simulations. As N increased, $\%\hat{Bias}_{\hat{N}}$ decreased. The $\%\hat{Bias}_{\hat{N}}$ estimates for $N \geq 5,000$, however, were not significantly different from 0 if we look at the SE_{sim}. Therefore, increasing N only helped reduce the bias of $\hat{N}$ to a point, but as long as the population was large enough, it did not effect estimates. As long as lakes were dense enough ($D \geq 0.042$ in our findings) we could assume we had accurate estimates of $\hat{N}$. The more interesting result that occurred when we change N was what happened to $CV(\hat{N})$: the precision of our estimates. In our simulations, $CV(\hat{N})$ constantly declined as we increased N. Therefore, in cases of larger N, our estimates were more precise. We can think of the change of $CV(\hat{N})$ as a change in the SE of mussels. For a population of $N = 10,000$, a .01 change in $CV(\hat{N})$ was the same as a $10,000 * .01 = 100$ mussel change in SE. Another important finding from these simulations was that no matter the value of N, the $\%\hat{Bias}_{SE}$ was negative and increased as N increased. This means that the $SE(\hat{N})$ calculated using Equation 12 under-predicted the true variation observed by running many simulations. This trend continued throughout the simulation results.

	N	$\bar{n}$	$\hat{\bar{N}}$	$\%Bias_{\hat{N}}$	SE_{sim}	$\%Bias_{SE}$	$CV(\hat{N})$
1	2,500	25	2,867	14.670	1.820	−5.820	0.302
2	5,000	45	5,033	0.660	0.980	−4.840	0.218
3	7,500	68	7,556	0.750	0.860	−10.630	0.192
4	10,000	91	10,208	2.080	0.790	−13.310	0.173
5	12,500	112	12,523	0.180	0.700	−15.160	0.157

Table 2: How N Effects Estimates

4.4.2 Effects of Changing σ

Next we looked at what happened when we changed σ. As noted before, we ran most simulations with a σ above what we observed in the Burgan data in order to have most iterations result in n above Buckland's suggested threshold. However, we also wanted to compare what would happen to our estimates if we did use $\sigma = 0.5$ as we observed and a low value of $\sigma = 0.2$.

Table 3 shows the results of these simulations for comparison. For both inputs of N, we see that as σ increased, the $CV(\hat{N})$ decreased, making our estimates more precise. This makes sense because a larger σ means better detectability and therefore better estimates of $\hat{N}$, so we would expect the variance of our estimates to be smaller. Bias on the other hand was a different story. Changing σ did not seem to have a large impact on $\%\hat{Bias}_{\hat{N}}$ except in the case of $\sigma = 0.2$ resulting in a high $\%\hat{Bias}_{\hat{N}}$ for $N = 7,500$. We can explain this because the $\bar{n}$ in these simulations was so low that many of the iterations failed to produce a $\hat{N}$ since the DSsim package will only compute if $n > 20$ for a given run. Therefore since we constructed our $\hat{N}$ estimate out of only the iterations that had $n > 20$ when many of the runs did not, our estimate was to larger than the truth. Tying our findings in these runs back to those in section 4.4.1, if N is large enough, changing σ might not result in a change of $\%\hat{Bias}_{\hat{N}}$ if the n was still large. Again, in all cases the $\%\hat{Bias}_{SE}$ was negative meaning we are under-predicting the $SE(\hat{N})$.

	N	σ	$\bar{n}$	$\hat{\hat{N}}$	$\%Bias_{\hat{N}}$	SE_{sim}	$\%Bias_{SE}$	$CV(\hat{N})$
1	7,500	0.200	25	8,051	7.340	1.460	−2.360	0.261
2	7,500	0.500	54	7,513	0.170	0.950	−10.230	0.213
3	7,500	0.700	68	7,556	0.750	0.860	−10.630	0.192
4	10,000	0.200	30	10,118	1.180	1.200	−14.160	0.263
5	10,000	0.500	72	10,018	0.180	0.880	−16.690	0.196
6	10,000	0.700	91	10,208	2.080	0.790	−13.310	0.173

Table 3: How Sigma Effects Estimates

The results from changing both N and σ can be visualized in Figure 10. As we increased $\sigma > 0.2$ $\%\hat{Bias}_{\hat{N}}$ did not decrease significantly. We see the same result as we increased N, above 2,500 objects.

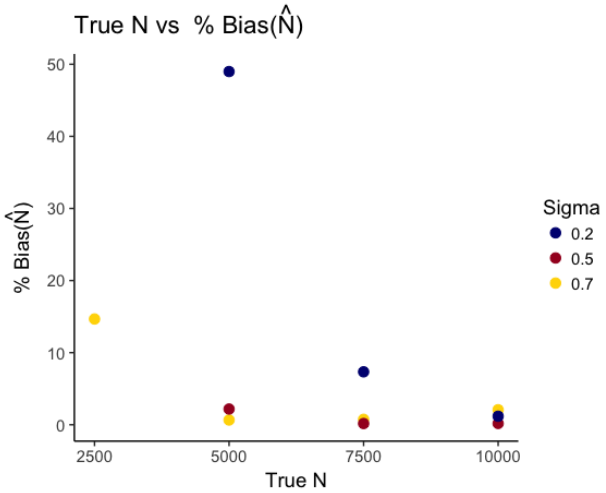


Figure 10: A visualization of how changing N changes Bias

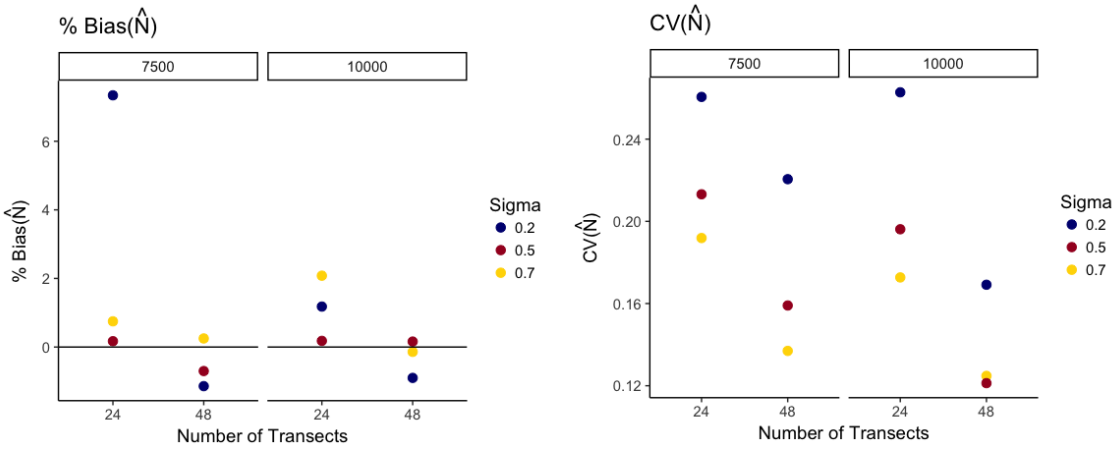
While these results are interesting, σ and N are not factors that we can explicitly control when we sample. The lake or area will have a total population we cannot change. While there might be ways to influence σ (one of these ways will be discussed in the next section), when

setting up a sample design we do not know the value. While these are out of our control, their implications on Bias as well as SE are important to understand. If we sample lakes with assumed smaller populations, or where detectability appears low, our estimates are more variable and less accurate. We should take this into account when evaluating estimates and know that if we are predicting a small $\hat{N}$ for a region, our prediction may not be accurate. Therefore there might be other ways in which we can change our survey designs to improve estimates. The next set of analyses will focus on variables in survey designs that we can control.

4.4.3 Effects of Changing Number of Transects K

One parameter we could change in our design is the number of transects, K. Table 4 presents the results of simulations where we doubled the number of transects from 24 to 48. For population sizes of 7,500 and 10,000 we ran simulations with both variations of K as well as the three σ values previously analyzed.

Figure 11a displays the results graphically. In both population facets we see that as K increased from 24 to 48, the $\%Bias_{\hat{N}}$ became closer to zero (highlighted by the black line). This also held for all three values of σ. Using the SE_{sim}, we also concluded that none of the $\hat{\bar{N}}$s were significantly different than 0 in the $K = 48$ simulations. Therefore increasing K did appear to decrease $\%Bias_{\hat{N}}$, but not in a statistically significant way if our N and σ were large enough. Increasing K did make a difference in the case of small σ and N values like the simulation with $N = 7,500$ and $\sigma = 0.2$.



(a) Effects on % Bias

(b) Effects on $CV(\hat{N})$

Figure 11: How changing K, σ, and N affect estimates

Intuitively, these results make sense. As one increases K, one expects more observations n. Table 4 also illustrates that as K doubled, n doubled in this constant density scenario. With larger n we expect to make more accurate predictions of N.

The other statistic of interest was the $CV(\hat{N})$. In all cases, as K increased, $CV(\hat{N})$ decreased. This change is visualized in Figure 11b. This effect can be explained by our analysis of CV in section 3.3.2. In these simulations, because both K doubled and n doubled, $\frac{n}{K}$ did not change. Therefore as we increased K, we slightly decreased $CV(\hat{N})$. From Figure 6, we know that as $\frac{n}{K}$ became large, the difference between values of K shrunk. Our largest n in these sets of simulations, however, was $n = 92$ with $K = 48$. Therefore $\frac{n}{K} = 3.75$ which is not a very large

value, so we would expect a difference in $CV(\hat{N})$ if we change K. Note that in all simulations we were under-predicting the true value of $CV(\hat{N})$ as seen in the negative $\%Bias_{SE}$.

N	σ	K	$\bar{n}$	$\hat{\bar{N}}$	$\%Bias_{\hat{N}}$	SE_{sim}	$\%Bias_{SE}$	$CV(\hat{N})$
7,500	0.200	24	25	8,051	7.340	1.460	−2.360	0.261
7,500	0.200	48	45	7,415	−1.140	1.260	−16	0.221
7,500	0.500	24	54	7,513	0.170	0.950	−10.230	0.213
7,500	0.500	48	107	7,448	−0.700	0.910	−17.190	0.159
7,500	0.700	24	68	7,556	0.750	0.860	−10.630	0.192
7,500	0.700	48	135	7,519	0.250	0.790	−10.380	0.137
10,000	0.200	24	30	10,118	1.180	1.200	−14.160	0.263
10,000	0.200	48	61	9,910	−0.900	0.970	−5.360	0.169
10,000	0.500	24	72	10,018	0.180	0.880	−16.690	0.196
10,000	0.500	48	144	10,016	0.160	0.700	−4.980	0.121
10,000	0.700	24	91	10,208	2.080	0.790	−13.310	0.173
10,000	0.700	48	180	9,986	−0.140	0.720	−13.980	0.125

Table 4: How the Number of Transects Effects Estimates

4.4.4 Stratified Design

The next question we hoped to answer, was how running a stratified design would change estimates. In this case, we held total K constant, but changed the number of transects between strata. In one design, $K_1 = 16$ and $K_2 = 8$ where K_i was the number of transects in strata i. Thus, one strata contained half as many transects as the other. Comparing a design with 24 evenly spaced transects to one with the two strata design, we did not see statistically different results. The $\%Bias_{\hat{N}}$ were within $2SE_{sim}$ from each other and the $CV(\hat{N})$ were the same. Therefore, we conclude that when you have a constant density, a stratified design does not help or hurt your estimates. The results can be viewed in the first two rows of Table 5.

4.4.5 Stratified Designs with Unequal Densities

The next simulations considered what would happen when the strata had different densities. In this case one strata had a larger density than the other with the overall population staying at $N = 10,000$. This aimed to mimic what might happen in a lake where half of the lake was more infested with zebra mussels than the other. We will refer to the more infested area as the "infestation zone" or "infest-strata."

Since our research team cared about whether a lake was too infested for treatment, they wanted to have the best estimate for the densest region of the lake. Therefore, if we were implementing a design where we placed a different number of transects in the different strata, we would want to place more transects in the "infest strata" because as we found earlier, more transects meant more accurate and precise estimates. For our simulations we placed 16 transects in the infestation zone and called this design "correctly identifying" the infestation zone. The remaining 8 transects were placed in the non-infestation zone, keeping the total number of transects constant at 24. Since researchers do not always know where the infestation zone is, we

also ran simulations where 16 transects were in the non-infestation zone and the 8 transects were in the infestation zone. We called this design, "incorrectly identifying" the infestation zone.

Table 5 includes the results of these transect designs in the last two rows, along with the third row that shows what would happen if we placed 12 transects in both the infestation zone and the non-infestation zone. In the case of incorrectly identifying the infestation zone, we found 25 less observations and under-predicted N. In the correctly identified design, we slightly over-predicted N. After analyzing the SE_{sim}, however, neither simulation produced estimates significantly different than the actual N. The $CV(\hat{N})$ of the two designs are close, with the correctly identified infestation zone having a slightly lower $CV(\hat{N})$. Also note that the correctly identified infestation zone design did not appear to preform any better than the case where we place 12 transects in both strata. In all cases, the $\%Bias_{SE}$ reported that we are under-predicting the true SE.

N_1, N_2	K_1, K_2	$\bar{n}$	$\hat{\bar{N}}$	$\%Bias_{\hat{N}}$	SE_{sim}	$\%Bias_{SE}$	$CV(\hat{N})$
10,000	24	91	10,208	2.080	0.790	−13.310	0.173
5,000,5,000	16, 8	90	10,103	1.030	0.780	−9.320	0.172
7,500, 2,500	12, 12	90	10,107	1.070	0.940	−9.860	0.161
7,500, 2,500	16, 8	105	10,032	0.320	0.740	−13.270	0.164
2,500,7,500	16, 8	75	9,985	−0.150	0.760	−4.390	0.170

Table 5: How Changing Density and Transect Placement Effects Estimates

4.4.6 Addition of Hotspots

Before drawing any conclusions about our different sampling designs, we also wanted to investigate what would happen to our estimates if hotspots were present. A "hotspot" is an area of increased density. Zebra mussels spread out from a starting location so it makes sense that there would be hotspots in different areas of the lake. Since we did not have expert knowledge on the size of a hotspot we would observe for zebra mussels specifically, our analysis was more broad.

For our simulations we added the same size hotspot to each of the designs analyzed in the previous section. Table 6 displays the results of these simulations. In the case of the original design, we had $N = 10,000$ and $K = 24$ with one strata. The hotspot was placed in the middle of the area. In the correctly identified infestation design, the hotspot was placed in the strata with $N_1 = 7,,500$ that also had $K_1 = 16$. In the incorrectly identified infestation design, the hotspot was placed in the strata with $N_2 = 7,500$ and $K_2 = 8$. Since our transects were randomly placed with each iteration, the position of the hotspot within the desired strata did not matter. In some cases a transect might have fallen directly on that hotspot and in other cases, missed the hotspot completely because of the spacing between transects. When sampling a lake, we most likely would not know where a hotspot was located, so we would be unsure if a transect had been placed on a hotspot, or if there were hotspots we missed. Therefore we wanted to especially take note of the $CV(\hat{N})$ in these simulations since, depending on where transects are placed, we may see very different estimates of $\hat{N}$.

Table 6 shows the results of the simulations. In the most simple case, we examined what happened when we added a hotspot to our original design. We did not actually see a significant difference in designs when looking at our resulting values. The most significant change occurred in the $CV(\hat{N})$, where the value dropped when a hotspot is added. The same can be said for the correctly identified infestation zone design, where only the $CV(\hat{N})$ seemed to decrease. Another

Design	Hotspot	$\bar{n}$	$\hat{\bar{N}}$	$\%Bias_{\hat{N}}$	SE_{sim}	$\%Bias_{SE}$	$CV(\hat{N})$
Original	No	91	10,208	2.080	0.790	−13.310	0.173
Original	Yes	92	10,198	1.980	0.980	−8.420	0.166
Correctly Identified Infestation	No	105	10,032	0.320	0.740	−13.270	0.164
Correctly Identified Infestation	Yes	105	10,085	0.850	0.900	−1.200	0.155
Incorrectly Identified Infestation	No	75	9,985	−0.150	0.760	−4.390	0.170
Incorrectly Identified Infestation	Yes	75	9,931	−0.690	1.280	−20.260	0.222

Table 6: How Hotspots Effect Estimates

way we could compare these scenarios was to look at the distribution of $\hat{N}$ from the simulations. Figure 12 shows the distribution for the correctly identified infestation zone design when a hotspot was and was not present. As the summary statistics suggest, there did not appear to be difference when a hotspot was present under the correctly identified infestation design.

In contrast, we did see a difference under the incorrectly identified hotspot design. While $\bar{n}$ remained the same and the $\%Bias_{\hat{N}}$ was not significantly different, SE_{sim}, $\%Bias_{SE}$, and $CV(\hat{N})$ all dramatically increased when a hotspot was present. If we only had 8 transects in the infestation zone and one of those transects fell directly on the hotspot, there was many more detections on that transect than others in the strata. Looking back at the $SE(\hat{D})$ Equation 12, if $\frac{n_k}{l_k} - \frac{n}{L}$ was large, which was the case if we had one transect with many more observations, it increased $CV(\hat{N})$. Figure 13 shows the distribution of $\hat{N}$s produced through the simulations for the incorrectly identified infestation zone both with and without a hotspot. The greater $CV(\hat{N})$ for the hotspot simulation is shown here as the distribution of $\hat{N}$s is greater. Notice that both are still centered around the actual value of $N = 10,000$, but when a hotspot was present, there was large $\hat{N}$ outliers. In these outliers, a transect was most likely falling directly on the hotspot skewing our estimates of $\hat{N}$. Therefore, if we cannot identify where hotspots are, using a stratified design runs us the risk of either vastly over-predicting $\hat{N}$ or under-predicting.

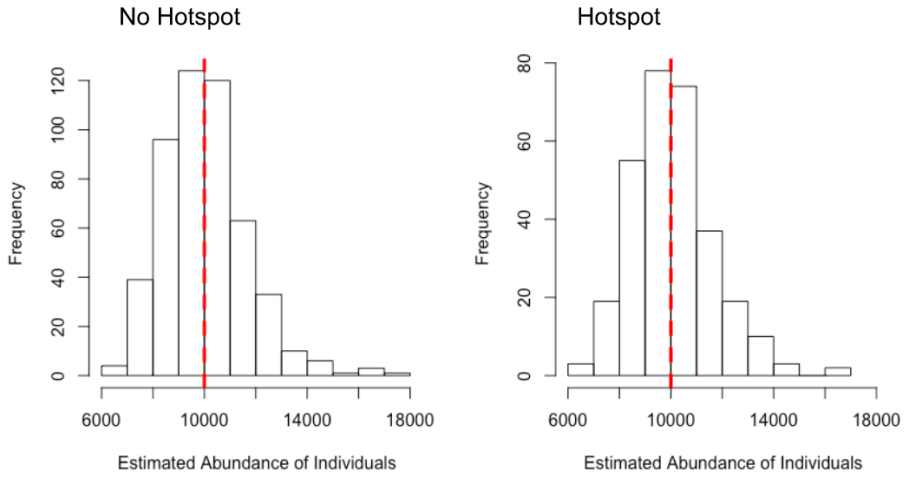


Figure 12: $\hat{N}$ Distribution for Correctly Identified Infestation Design

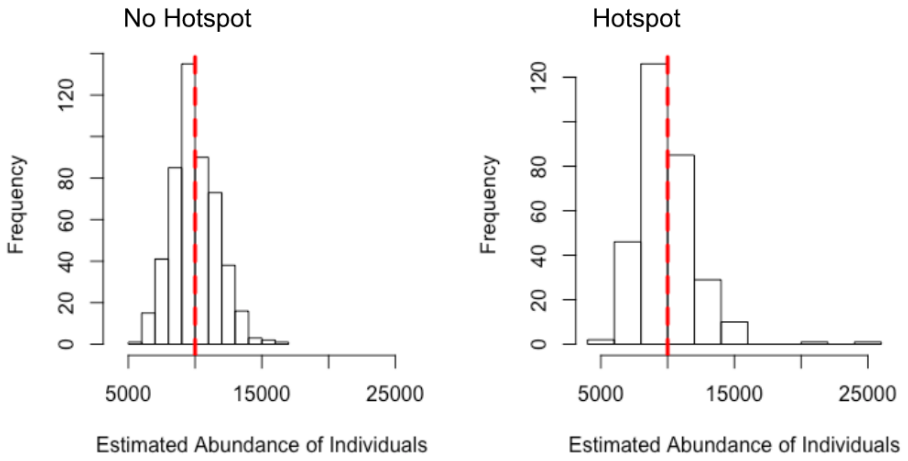


Figure 13: $\hat{N}$ Distribution for Incorrectly Identified Infestation Design

4.4.7 Comparing Designs

Based on these results, what is the best sampling design? Figure 14 displays the $\hat{N}$ distribution for a two-strata population with $N_1 = 7,500$, $N_2 = 2,500$ and a hotspot present in the infestation zone. Each graph however has a different transect design. The first has 12 transects in both strata, the second is the correctly identified infestation zone design and the final is incorrectly identifying the infestation zone. Looking at the distribution of $\hat{N}$, the incorrectly identified design preforms the worst as we have discussed. Comparing the correctly identified design against just using 12 transects in both strata, however, appears to not have a significant difference. Therefore, if we cannot identify the infestation zone, even if hotspots are present we are better off just using a constant transect placement across the entire lake so we do not mistakenly limit the number of transects in the infestation zone.

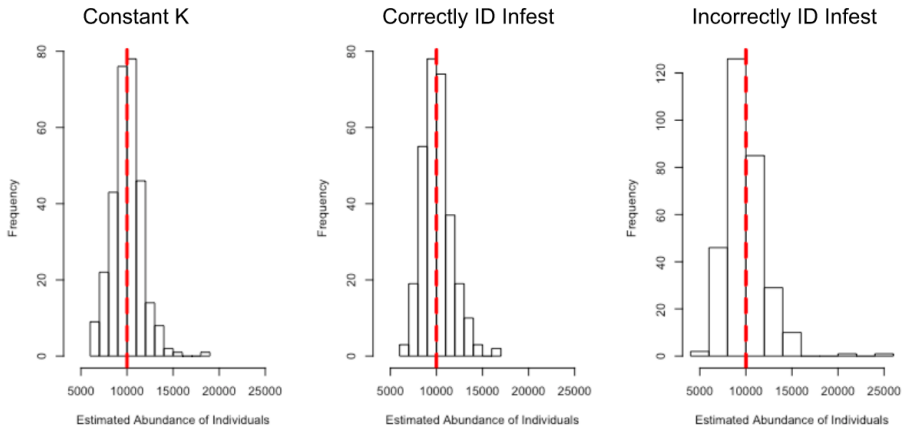


Figure 14: $\hat{N}$ Distribution for Different Designs

5 Time and Detection

After working with the Lake Burgan data provided by the researchers and running various simulations, our group became increasingly interested in what we can control to improve our estimates. Time spent on a transect was a factor that we thought had the potential to influence detection. We wanted to investigate if there was a trade off between faster, more efficient transects and slower, more methodical transects. Specifically, how would the half-normal model change as we varied time? When sampling lakes, the diving team only has so much time to spend at a given lake, so what is the best way to allocate their time? Should teams be sampling as many transects as they can, but not very thoroughly? Or should they be sampling less transects, but meticulously search for as many mussels as possible in the given transect area?

Because our lake data did not have a time component recorded for each transect, we were not able to use this data to investigate this trade-off. Instead, we decided to develop our own experiment to collect time data, mimicking a line transect by laying down a "line" on the floor of a tennis court and scattering debris (pillow batting and cotton balls) around the court. We then placed thirty mini-marshmallows around the court according to randomly generated "distance along" the transect and "distance from" the transect measurements. The marshmallows were nestled in the batting, placed next to cotton balls, and a few others free of debris. This design was meant to reflect substrates our "mussels" could attach to.

We developed two separate sets of instructions, one encouraging participants to walk the transect searching for marshmallows as quickly and efficiently as possible, and the other asking participants to be slow and methodical in their search. While we did not enforce any time restrictions, these differing instructions ensured that we would have a wide range of detection and time measurements. On the day of our experiment, participants arrived at the tennis court and we flipped a coin to determine the instruction treatment they would receive. The participant then completed the transect, and we recorded which mussels were detected and total time elapsed.

With the collected data, each participant (from here on to be referred to as an observer) was fitted to a half-norm model and $\hat{\sigma}$, $\hat{\mu}$, and estimated population were extracted. To investigate the relationship between time and detection, we plotted these fitted $\hat{\sigma}$, $\hat{\mu}$, n, and $\hat{N}$ against the time spent on the transect. We noticed one observer had a $\hat{\sigma}$ of nearly 600 compared to others' $\hat{\sigma}$s between 1 and 4. After further investigation, we realized this observer had been running to the outside edge and working her way back in, toward the transect when searching. Thus, she did not have a half normal function starting at the transect, but a reversed version of this model. As this will not be done during future surveys, her data was removed. Furthermore, it became apparent that many of the marshmallows near the edge of the 5 meter half width were very easy to find in comparison to those closer. As this is not expected to happen in reality, we truncated the data and half-normal model fit to only 4 meters.

As the estimated population is of interest to the scientists investigating a particular lake, we began our analysis by plotting each $\hat{N}$ against the total time for the observer. The relationship between $\hat{N}$ and time was determined to be positively linear. To investigate why this might be, we turned to inputs within this estimated parameter: n and μ.

The total number of mussels observed has a non-linear relationship with time, as seen in Figure 16. While the relationship for this data appears non-linear in nature, it is important to note the few data points with the longest time. As there were limited observers, these observations could be influencing the structure of the data. For this experiment, this is still a strictly increasing function, so as time increases, so does the number of observations made within the transect.

Another important parameter to consider in calculating the predicted population is $\hat{\sigma}$ and its

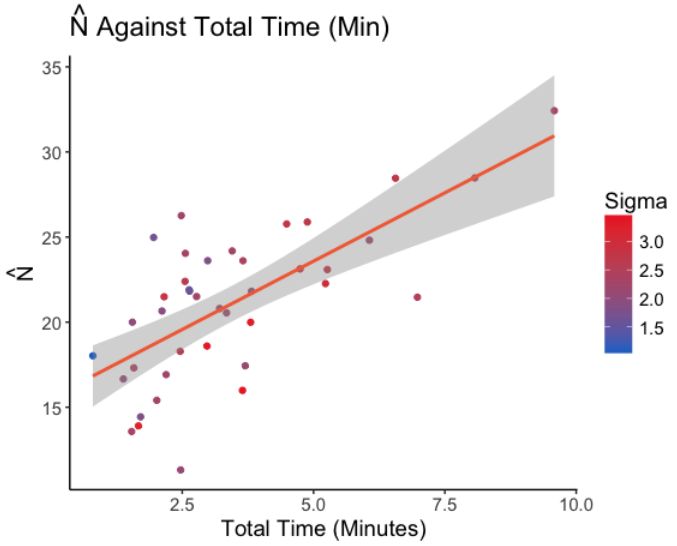


Figure 15: Fitted $\hat{N}$ against total time

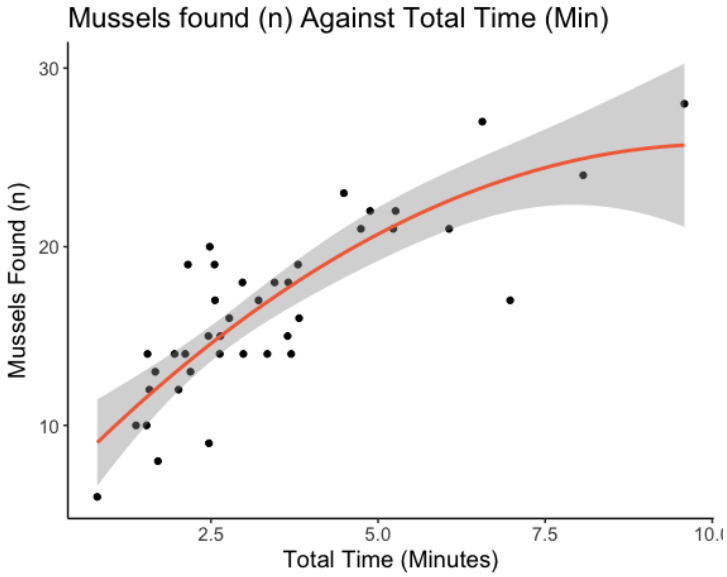
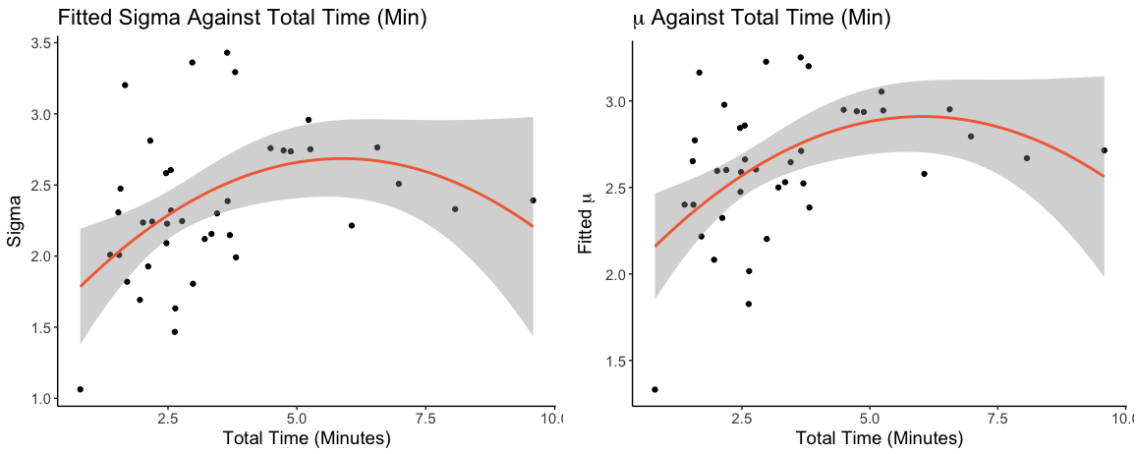


Figure 16: Total observations against total time

relationship with $\hat{\mu}$. As seen in Figure 17, $\hat{\sigma}$ and $\hat{\mu}$ share a very similar relationship to time. As time increases, both $\hat{\sigma}$ and $\hat{\mu}$ increase at a decreasing rate. Toward the higher values of time, there is a larger standard deviation, indicating these functions may not become negative after a certain point as displayed in the results of this experiment.

To further explore the relationship between $\hat{\sigma}$ and $\hat{\mu}$, we can return to the half-normal model as it pertains to detection probability. Figure 18 displays sigmas increasing from 2 to 4 by increments of 0.5, with the dark blue indicating the lowest σ and red indicating the largest. As σ increases, so does μ, but at a decreasing rate. μ, again, is the area under the whole curve. Each slice of color in this plot indicates the additive value of increasing σ by 0.5. As a result, μ has a non-linear relationship with σ.



(a) $\hat{\sigma}$ against total time

(b) μ against total time

Figure 17: $\hat{\sigma}$ and $\hat{\mu}$ against total time fitted with a quadratic model

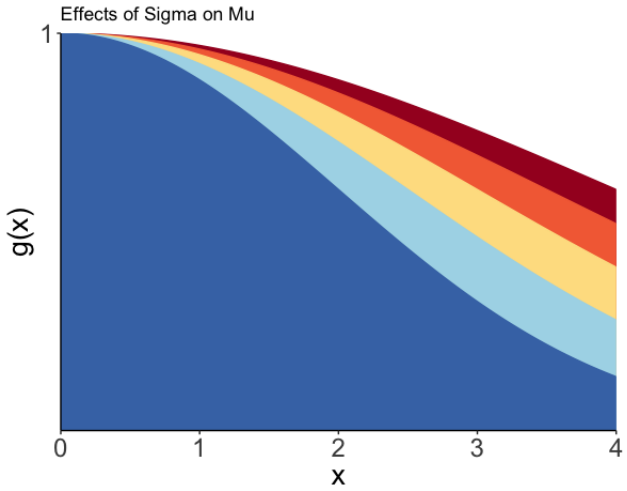


Figure 18: Varied Sigma Compared to Mu

The fitted models from the observers within the experiment indicate non-linear relationships between the parameters $\hat{\sigma}$, $\hat{\mu}$, and n and time. As shown previously in Equation 10, $\hat{N}$ is a function of $\hat{P}_a$, which is equal to $\hat{\mu}/w$. When we rearrange Equation 10, we get that the predicted population is a function of both n and $\hat{\mu}$:

$$\hat{N} = \frac{nA}{a(\hat{\mu}/w)} \tag{21}$$

Within the context of this experiment, both n and $\hat{\mu}$ have a non-linear relationship with time, thus this non-linear effect will cancel when used to calculate $\hat{N}$, resulting in a linear relationship between predicted population and time.

When considering different sampling designs, we want one that increases detection as much as possible, and gives us an estimate close to the parameter of interest. If time is to be used to control certain parameters within the model, we must consider what we wish to optimize. As it pertains to detection probability, the larger the σ, the closer we move toward a uniform detection

model. The larger the σ, the more detections are made, increasing n. As these values, σ and n, have a non-linear relationship, there exists some time that optimizes these parameters. If these functions do not become negative, but simply level off, there exists some point where the marginal gain in σ is not worth the extra time.

6 Discussion and Conclusions

The results of our simulations can provide researchers with more knowledge of how the estimators of abundance preform under different situations as well as suggest optimal sampling strategies. If we predict that a lake has a low density after surveying, our estimates are probably more bias as well as less precise, as we saw when we had low N values. So while we cannot control the N of a lake, it is important to understand how our estimators preform under low N values. We also saw in our simulations that when σ was low, our estimates were also more bias and less precise. Therefore if we are predicting low σ values, our estimates are less accurate.

A way to combat this potential bias and imprecision is to increase the number of transects. In our simulations where we doubled the number of transects in the sample area, we saw both accuracy and precision increase. If the population of the lake is large enough, however, such that we are finding Buckland's suggested number of mussels, more transects are not necessary to decrease our bias. All the simulations we ran predicted $SE(\hat{N})$ as smaller than the actual distribution of the $\hat{N}$ values from the 300-500 runs. Therefore we conclude that the Equation 12 to estimate $SE(\hat{N})$ is biased.

We also looked into the trade-offs of running a stratified design where one strata had more transects than the other. While there may be some benefit to placing more transects in the infested zone, especially when a hotspot is present, if we misidentify the infestation zone, we risk making large prediction errors. Therefore unless researchers know exactly which area of the lake is the most infested, we would recommend keeping with a constant transect design since it minimizes the chance of large outliers in estimates.

As it pertains to survey design, time has provided an avenue to better detection and improved precision. There exists some optimal time spent on a transect such that σ in the half-normal model is optimized, and the number of mussels found reaches a maximum (or a point where a marginal increase in n is not worth the extra time). The relationship between time and n supports the claim made previously: to control the precision of our estimates, $CV(\hat{D})$, we can increase or reduce the number of mussels found as we change the amount of transects in the design. If fewer transects are to be used to survey, more time must be spent on each to reduce error. If more transects are used, the divers are free to move faster to achieve the same level of precision. Before, the number of transects used in a design was the only way to influence estimators like $\hat{N}$ and $\hat{D}$. Now, by adjusting time spent on a transect, parameters like σ and μ can be influenced as well.

7 Further Research and Limitations of our Analysis

One potentially interesting area of further research for this topic could involve incorporating habitat covariates into the model. Because of the nature of the hazard rate and half-normal distributions, we can change σ according to different variables. Unfortunately, our data only had two habitat descriptors, sand or silt. It could be intriguing to look at the impact of more varied habitats on detection probability for zebra mussels, as zebra mussels are found in a wide range of lake habitats, including rocky or muddy zones.

One of the focuses of our research addressed the effect of hotspots on detection probability. While we were able to simulate a hypothetical hotspot, our created hotspot was circular, which may not be the most realistic shape. If we had more information about the actual structure of zebra mussel hotspots, for example how exactly they multiply and expand, our findings may be more realistic.

Lastly, we are interested in expanding on our time analysis. Because of our limited resources and time with this project, we believe there could be much to build upon with our experiment. The first would be to expand the number of transects. Instead of sampling from just one transect, it would be informative to have multiple transects the participants could walk along to gather more data. Because this experiment was conducted during the winter, there were limited locations to hold the experiment. We would have rather conducted the sampling in a more realistic setting, instead of a gymnasium. If someone with more resources were to conduct this experiment, it would be interesting to set up transects in the water where actual divers could sample zebra mussels.

Most of the limitations of our study stemmed from only having a limited amount of data. The literature recommends that the total line-transect sampling should have a minimum of 60-80 observations (Buckland, 2011). Our data from Lake Burgan, however, only contained 52 observations. While this detail should be noted, we do not believe that the 60-80 observation minimum is a harsh threshold. Our research still produced informative results, even with the low number of observations.

The DSsim package we utilized for our simulations would not complete a simulation if less than 20 observations existed for an iteration. Unfortunately, this prevented us from exploring data with a small amount of observations. The immense amount of time required to run many of our simulations also limited the amount of simulations we were able to complete. Some of the more elaborate designs took over 48 hours to run, which prevented us from running more simulations. Even with these limitations, we were able to make worthwhile insights into the line-transect distance sampling methodology. We believe that there is substantial potential in this field for further exploration and study.

References

Buckland, S.T., E.A. Rexstad, T.A. Marques, and C.S. Oedekoven. 2015. Distance Sampling: Methods and Applications. Switzerland. Springer International Publishing.

Hart, R.A., A.C. Miller, and M. Davis. 2001. Empirically Derived Survival Rates of a Native Mussel, Amblema plicata, in the Mississippi and Otter Tail Rivers, Minnesota. American Midland Naturalist 146: 254-263.

Hebert, P. D. N., B. W. Muncaster, G. L. Mackie. 1989. Ecological and genetic studies on Dreissena polymorpha (Pallas): a new mollusk in the Great Lakes. Can. J. Fish. Aquat. Sci. 46: 1587-1591.

Limburg, K. E., V. A. Luzadis, M. Ramsey, K. L. Schulz, and C. M. Mayer. 2010. The good, the bad, and the algae: perceiving ecosystem services and disservices generated by zebra and quagga mussels. Journal of Great Lakes Research 36:86-92.

Marshall, Laura. 2017. DSsim: Distance Sampling Simulations. R package version 1.1.2. https://CRAN.R-project.org/package=DSsim

Miller, David Lawrence. 2017. Distance: Distance Sampling Detection Function and Abundance Estimation. R package version 0.9.7. https://CRAN.R-project.org/package=Distance

Miller, E. B., M. C. Watzin. 2007. The effects of zebra mussels on the lower planktonic foodweb in Lake Champlain. Journal of Great Lakes Research 33(2):407-420.

Qualls, T. M., D. M. Dolan, T. Reed, M. E. Zorn, and J. Kennedy. 2007. Analysis of the impacts of the zebra mussel, Dreissena polymorpha, on nutrients, water clarity, and the chlorophyll-phosphorus relationship in Lower Green Bay. Journal of Great Lakes Research 33(3):617-626.

USGS Nonindigenous Aquatic Species. Dreissena polymorpha.
https://nas.er.usgs.gov/queries/factsheet.aspx?speciesID=5

Vanderploeg, H. A., J. R. Liebig, W. W. Carmichael, M. A. Agy, T. H. Johengen, G. L. Fahnenstiel, and T. F. Nalepa. 2001. Zebra mussel (Dreissena polymorpha) selective filtration promoted toxic Microcystis blooms in Saginaw Bay (Lake Huron) and Lake Erie. Can J. Fish. Aquat. Sci. 58: 1208-1221.

Virginia Department of Game and Inland Fisheries. Zebra Mussels (Dreissena polymorpha). https://www.dgif.virginia.gov/wildlife/zebra-mussels/