Assessment of the Risk of Cyanobacteria Blooms in Geist Reservoir

Polly Fairfield, Alyce Kanabrocki, Chris Round, Keyao Sun, Brandon Taylor

May 8, 2015

Contents

List of Figures

List of Tables

Note: this document is associated with many wide and long tables which are difficult to accommodate. They are electronically attached as csv files. As well, print outs of excel tables will be included with the final hard-copy.

1 Introduction

1.1 Statement of the Problem

Since the first census in 1790, the percentage of Americans living in urban environments has increased from 5.1% up to 80.7% in 2010. Along with this increase in urbanization, our nation's population has increased from just under 4 million to around 309 million in 2010 (*1990 United States Census, summary* 1993). While this incredible growth and development has allowed many opportunities for the United State's citizens and economy, it has also had a negative impact on the environment, particularly water systems. The direct pollution of urban streams and rivers is in fact what brought the need for environmental protection to the nation's attention when the Cuyahoga River in Cleveland, Ohio, caught on fire in 1969 (Adler, 2002).

In more recent times, the impact on larger bodies of water, such as lakes and coastal areas, from pollutants present in streams and rivers, has been a growing area of concern. Events such as the Gulf Hypoxic Zone and the 2014 algal bloom in Lake Erie that resulted in Toledo, Ohio losing their drinking water supply, have increased awareness of the negative impact development of development. This includes the impact of these blooms on the health of the water systems, human health, and the economy. As development increases, the amount of pervious surfaces, which allow water to infiltrate the ground, has decreased as it is replaced with impervious surfaces such as roads, parking lots, and buildings. This results in more runoff into water bodies, bringing with it high levels of sediment and pollution.

One of the leading sources of pollutants that impacts water quality, and leads to algal blooms, is nutrient runoff from agricultural lands. The influx of excess nutrients, particularly phosphorus and nitrogen, causes algae populations to spike. With certain species of algae, or cyanobacteria, the algae may then release toxins. These toxins are primarily neurotoxins, and have been responsible for serious illness in humans, the deaths of dogs and other animals, and negative impacts on the ecology of surrounding areas. Additionally, algal blooms can

have serious economic implications, as they can make tourist and recreational destinations inaccessible.

To reduce the risks associated with algal blooms, methods of decreasing the amount of pollutants reaching water bodies has been a popular topic of research. Best Management Practices (BMPs) are one of the most popularly researched and utilized methods. BMPs are a suite of methods that work towards decreasing the amount of pollutants that ultimately reach a specific water body. They range from methods such as stream fencing, to keep livestock (and their waste) out of streams, to "buffer strips" of vegetation around streams and rivers to help trap some of the pollutants.

BMPs have been shown to be very effective at reducing loads of certain pollutants, some on a short time-scale. For managers of lakes and municipalities who are concerned with the potential risks associated with algal blooms, implementing BMPs in the surrounding watershed may be a viable strategy. The goal of our research and analysis was to assess the possible impact BMPs can have on reducing the probability of an algal bloom, and therefore the reduced risk of harm to humans, animals, and the local economy. We chose Geist Reservoir, located in Indianapolis, Indiana, as our study site.

1.2 Site Characterization

1.2.1 Geist Reservoir

Geist Reservoir is a manmade reservoir located in northeastern Indianapolis, Indiana. The reservoir was built by damming Fall Creek and provides drinking water and recreational opportunities to Indianapolis and the surrounding population (watershed plan). Geist Reservoir provides drinking water to about 300,000 people, and about 200,000 people use the reservoir for recreation each year (*2013 drinking water report* 2013).

Figure 1.1: Geist Reservoir

1.2.2 Algal Blooms

Geist Reservoir is at risk for toxic cyanobacteria (blue-green algae) blooms. Monitoring by the Indiana Department of Environmental Management and Citizen's Energy has found high cyanobacterial cell counts in the reservoir for the past 5 years, (*2013 drinking water report* 2013). The high cell counts indicate the possibility that an algal bloom could form on the surface of the water and release toxins harmful to human and ecological health.

These blooms appear in lakes and reservoirs in summer months and grow best in warm, still, eutrophic waters (Persaud et al., 2015). Factors influencing the growth of blooms include light availability, nutrient availability, water flow, and water column stability (*Cyanobacteria/cyanotoxins, US EPA* 2015). However, the presence of a bloom does not necessarily indicate the presence of toxins; how toxins are formed is less well known (*Cyanobacteria/cyanotoxins, US EPA* 2015).

When toxins do appear, they can be dangerous to human health and the environment and can cause a wide range of symptoms. The population surrounding Geist Reservoir is particularly at risk since the reservoir is utilized both for drinking water and for recreation.

1.2.3 Watershed Land Use

The Geist Reservoir/Upper Fall Creek Watershed is approximately 140,000 acres and is composed of primarily agricultural uses (72.76%) (*Geist Reservoir/Upper Fall Creek Watershed management plan* 2011). Urban areas also comprise a

significant component of the watershed (16.47%). Table 1 shows a summary of land use by acres. The heavy agricultural and urban land use in the watershed contributes to the threat of harmful algal blooms by contributing to nutrient runoff (*Geist Reservoir/Upper Fall Creek Watershed management plan* 2011).

Urban	Cropland	Pastureland	Forest	Other	Feedlots
24837.2	95309.3	4766.1	8996	6251.6	4.8319

Table 1.1: Land Use in Geist/Upper Fall Creek Watershed in Acres

To provide an assessment of the potential risk to human health and the local ecology from toxins produced by cyanobacteria, a series of models were developed to model the potential for harmful algal blooms (HAPs) as well as the potential for human and ecological exposure to these blooms. The models consisted of a release model to characterize the likelihood of a toxin release, an exposure model to determine the likelihood and extent of human or ecological exposure, a toxicological model to characterize the potential human health and ecological effects, and a final risk characterization to summarize the interactions of these models and the final risk to humans and the local ecology.

2 Release Assessment

The release assessment models the inputs that contribute to potential algal blooms and subsequent toxin release into Geist Reservoir. First, the potential nutrient load from the watershed is calculated, Next, the nutrient load and other inputs are incorporated into an algal bloom model, and finally, into a toxin release model.

2.1 Nutrient Transport Model

Nutrients, particularly phosphorus, can play an important role in the appearance of algal blooms in lakes and reservoirs. In order to calculate the total nutrient load from the watershed, the STEPL (Spreadsheet Tool for the Estimation of Pollutant Load) model was used. The results of this model will be used as an input into the algal bloom model which will model the release of toxins into Geist Reservoir. STEPL uses algorithms to calculate nutrient loads from different land uses and load reductions from implementation of Best Management Practices (BMPs) using the Microsoft Excel (STEPL GUIDE). Inputs into this model include land use (in acres), precipitation (in inches), total agricultural animals by type, septic system parameters, and BMPs implemented. Once these inputs are entered, the model generates annual nutrient and sediment loading. For the purposes of this assessment, the annual phosphorus loading is the variable of interest.

2.1.1 Inputs

The various inputs for the model were taken from two sources, the Geist Reservoir/Upper Fall Creek Watershed Management Plan and the Geist Reservoir Restoration Long-Term Capital and Maintenance Plan. Tables 1-4 below summarize the inputs.

Watershed	Urban	Cropland	Pastureland	Forest	User Defined	Feedlots
W1	24837.2	95309.3	4766.1	8996	6251.6	4.8319

Table 2.1: Land Use Inputs

Watershed	Beef Cattle	Dairy Cattle	Swine (Hog)	Sheep	Horse	Chicken	Turkey	Duck
W1	376	293	12651	372	575	579	8	4
Total	376	293	12651	372	575	579	8	4

Table 2.2: Agricultural Animals

Watershed	No. of Septic Systems	Population per Septic System	Septic Failure Rate, %
W1	8312	2.43	1.09

Table 2.3: Septic Systems

Cropland	Forest	Urban
Conservation Tillage	Reforestation	Detention Basins
Cover Crops		Grassed Waterways
Livestock Exclusion		Golf Course Buffers
Bank Stabilization		Low Impact Development
		Pervious Pavement
		Vegetated Swale/LID
		Wetland/Bioretention
		Wetland Restoration
		Stream Buffers
		Two-Stage Ditch

Table 2.4: List of Assumed BMP's by Land Use Type

Land Use Type	Percent Area Implemented Scenario 2	Percent Area Implemented Scenario 3
Cropland	6.00%	100%
Forest	2.50%	100%
Urban	23.10%	100%

Table 2.5: Area Implemented for BMP's by Land Use Type

2.1.2 Scenarios

The model was run to find the phosphorous loading under three different scenarios.

2.1.2.1 No BMPs Implemented This scenario determined the baseline level of phosphorus loading into Geist Reservoir.

2.1.2.2 Geist Reservoir Restoration Long-Term Capital and Maintenance Plan Suggested BMPs This scenario is based on a watershed tour conducted in 2014 as a part of the Geist Reservoir Restoration Plan. The watershed tour identified areas where BMPs could be implemented and made recommendations. This scenario takes the recommendations identified and models the phosphorus load reduction if the BMPs were implemented. However, the identified areas comprise only a small part of the entire watershed.

2.1.2.3 Ideal BMP Implementation Scenario This scenario takes the suggested BMPs from Scenario 2 and assumes that they are applied on 100 percent of the potential area. This scenario is likely not realistic for the Geist Reservoir/Upper Fall Creek Watershed, but does provide an upper bound for potential phosphorus load reductions.

2.1.3 Results

2.1.3.1 BMPs The combined effect of the BMPs was modeled using STEPL's BMP calculator. Since we have limited information about where in the watershed each BMP will be implemented and how they will interact, we assumed that they all had parallel effects. This follows the recommendations of the STEPL model for how to model combined effects of BMPs with limited information.

2.1.3.2 Phosphorus Load Reduction

2.1.3.2.1 Scenario 1 Scenario 1 modeled phosphorus loading into Geist Reservoir with no BMPs implemented. Table 6 shows the results of the modeling.

P Load (no BMP)	P Reduction	%P Reduction
lb/year	lb/year	%
162435.649	0	0

Table 2.6: Phosphorus Loading with No BMP's

2.1.3.2.2 Scenario 2 Scenario 2 modeled phosphorus loading into Geist Reservoir with BMPs recommended in the Geist Reservoir Restoration Long-Term Capital and Maintenance Plan. Table 7 shows the results of the modeling. In this model, the BMPs reduced phosphorus loading by .4%.

P Load (no BMP)	P Reduction	%P Reduction
lb/year	lb/year	%
162435.649	714.7602239	0.440026699

Table 2.7: Phosphorous Loading with Recommended BMP's

2.1.3.2.3 Scenario 3 Scenario 3 modeled phosphorus loading into Geist Reservoir assuming that BMPs were implemented in 100% of the available land area in the watershed. Table 8 shows the results of the modeling. In this model, the BMPs were shown to reduce phosphorous modeling by 79.1%.

P Load (no BMP's)	P Reduction	%P Reduction
lb/year	lb/year	%
162435.649	128536.5984	79.13078149

Table 2.8: Maximum Phosphorus Loading Reductions from BMPs

2.2 Algal Bloom-Microcystin Model

There are a handful of algal bloom models in the literature, and they are included in the table below. There are two general take-aways.

First, models are remarkably inconsistent. For a good example of this, Persaud analyzed two Canadian lakes in the same region and using the exact same methodology, and found opposite but significant signs on the phosphorus coefficient in the two models (Persaud et al., 2015). This heavily influenced our decision to restrict our data analysis to one lake. Second, for many variables, it seems that a modal distribution fits best. In general, it seems that optimum growth for cyanobacteria occurs at the modal conditions, and growth is unfavorable either far above or far below this mode.

Second, predictor variables found to be significant are highly inconsistent between analyses. Sometimes phosphorus is found to be significant, and sometimes not.

Source	Significant Variables
(Onderka, 2007)	Temperature, phosphorus
(Persaud et al., 2015)	Temperature, wind-speed, phosphorus, water column stability
(Tao et al., 2012)	Conductivity, dissolved organic carbon, temperature, pH
(Carvalho et al., 2011)	Water color, pH

Table 2.9: Cyanobacteria Models

Microcystin models are in fact quite rare in the literature. Models that are used tend to be highly non-linear and have very low predictive power. These

models are also multivariate, and include not only microcystin but several other cyanotoxin response variables. Typically, the only predictors used are the populations of various clades of cyanobacteria (Alonso Fernández et al., 2013; Garcia Nieto et al., 2011; Vilán Vilán et al., 2013).

Our own models, unlike models in the literate, use step-wise prediction. This was based an underlying notion of flows, and not stocks, being the dependent variables. See the simulation section for more details.

3 Exposure Assessment

The exposure assessment consists of two components: the potential toxin movement in the reservoir and the likelihood of human exposure to these toxins.

3.1 Toxin Movement in Geist Reservoir

Due to a lack of available measurements and time to go and take these measurements, we have developed the following assumptions to govern toxin movement within Geist Reservoir. These assumptions are important as they address the limitations to the data we had access to, including the fact that we are not able to deal with the stratification of the reservoir in our analysis and that measurements and algal bloom monitoring happen at only a few stagnant parts of the reservoir.

To allow for the remainder of our risk assessment, we made the following assumptions:

3.1.1 Assumption 1

The reservoir is perfectly and instantly mixed, so when algae blooms in one location, people anywhere in the lake are exposed to that bloom and potential toxin release.

3.1.2 Assumption 2

Algal blooms surveyed at the monitoring stations happen at the same frequency and likelihood as everywhere else in the reservoir.

Though we are aware that these assumptions are improbable and unrealistic, we have made them to allow us to complete the rest of the risk assessment. If more resources (time, money, people, expertise, etc.) were available, we would want to include movement of water (and algal species when bloomed) within the lake, "hotspots" within the reservoir that may be more prone to blooms, and other factors that could influence the spread of the algal blooms and released toxins, and therefore, increased health risks.

3.2 Likelihood of Exposure

To assess the likelihood of exposure to cyanobacteria toxins, we began by estimating the likelihood of exposure to Geist's water. We looked at four main pathways of exposure to Geist's water: drinking water, fishing (from a boat and from the shore), boating, and swimming. Other than the drinking water data, the data used was collected from a 1965 U.S. Department of Health, Education and Welfare Public Health Service report, titled "Water Quality Recreational Project: Geist Reservoir, Indianapolis, Indiana." This was the only data that could be located containing specific recreational usage of Geist Reservoir. We made the assumption that the proportion of the surrounding population that engage in these activities has remained the same, and multiplied the data on recreational use by a factor of 1.5, representing the increase in population.

3.2.1 Drinking Water

Based on data acquired from Citizen's Energy database (*2013 drinking water report* 2013), approximately 300,000 people get their drinking water from the reservoir. Based on data from (*Exposure factors handbook, 2011 edition* 2011), mean water ingestion averaged over all ages is 1,426 mL per day, with a 90% upper and lower confidence interval of 1,377 to 1,474 mL per day (see Table 3.1). We assumed all water ingested by those residents who have access to Geist reservoir water was from the reservoir itself.

Table 3-27. Per Capita[a] Estimates of Combined Direct and Indirect[b] Water Ingestion Based on NHANES 2003–2006, Mean Confidence Intervals and Bootstrap Intervals for 90th and 95th Percentiles: All Sources (mL/day)

Age	Sample Size	Mean			90th percentile			95th percentile		
			90% CI			90% BI			90% BI	
		Estimate	Lower Bound	Upper Bound	Estimate	Lower Bound	Upper Bound	Estimate	Lower Bound	Upper Bound
Birth to <1 month	88	295*	208*	382*	852*	635*	941*	954*	759*	1,037*
1 to <3 months	143	385*	325*	444*	1,049*	929*	1,074*	1,084*	1,036*	1,099*
3 to <6 months	244	527*	466*	588*	1,045*	1,023*	1,126*	1,190*	1,088*	1,250*
6 to <12 months	466	461	417	506	995*	903*	1,057*	1,126*	1,056*	1,212*
1 to <2 years	611	370	339	401	762*	673*	835*	912*	838*	1,084*
2 to <3 years	571	435	397	472	920*	836*	987*	1,086*	973*	1,235*
3 to <6 years	1,091	498	470	526	925	888	1,009	1,181	1,068	1,250
6 to <11 years	1,601	660	617	703	1,184	1,117	1,294	1,567	1,411	1,810
11 to <16 years	2,396	885	818	952	1,821	1,678	2,114	2,595	2,280	2,807
16 to <18 years	1,087	1,113	1,027	1,199	2,289	2,055	2,412	2,652	2,502	2,868
18 to <21 years	1,245	1,240	1,128	1,352	2,569	2,377	2,991	3,346	3,044	3,740
≥21 years	8,673	1,700	1,641	1,759	3,085	3,027	3,147	3,727	3,586	3,858
≥65 years	2,287	1,498	1,442	1,555	2,582	2,470	2,671	3,063	2,961	3,328
All ages	18,216	1,426	1,377	1,474	2,836	2,781	2,896	3,412	3,352	3,499

[a] Includes all participants whether or not they ingested any water from the source during survey period.
[b] Direct water is defined as water ingested directly as a beverage; indirect water is defined as water added in the preparation of food or beverages. Does not include indirect consumption of bottled water.
* Estimates are less statistically reliable based on guidance published in the *Joint Policy on Variance Estimation and Statistical Reporting Standards on NHANES III and CSFII Reports: NHIS/NCHS Analytical Working Group Recommendations* (NCHS, 1993).
CI = Confidence Interval.
BI = Bootstrap Interval.

Source: U.S. EPA analysis of NHANES 2003–2006 data.

Table 3.1: Human Water Ingestion

3.2.2 Recreational Contact

3.2.2.1 Fishing

3.2.2.1.1 From a Boat According to (*Water quality recreational project, Geist Reservoir, Indianapolis, Indiana* 1965), the average amount of time spent on a fishing trip is five hours, and approximately 71,775 people participate in boat fishing a summer season. This means that there are approximately 358,875 hours spent fishing (number of people * five hours) during a summer season.

3.2.2.1.2 Shore According to (*Water quality recreational project, Geist Reservoir, Indianapolis, Indiana* 1965), the average amount of time spent on a shore fishing trip is four hours, and approximately 75,120 people per summer season engage in shore fishing. This means that approximately 300,480 hours total are spent shore fishing during the summer (number of people * four hours).

3.2.2.1.3 Total Adding the total number of people that fish from a boat with the total number of people that fish from the shore, there is a total of approximately 146,895 people that engage in fishing during the summer season; this equals approximately 659,355 hours spent fishing. According to the EPA, the mean ingestion rate is 3.7 mL per hour of fishing, with an upper 97.5% confidence interval of 11.2 (*Water quality recreational project, Geist Reservoir, Indianapolis, Indiana* 1965).

Table 3-93. Estimated Water Ingestion During Water Recreation Activities (mL/hr)

Activity	Surface Water Study				Swimming Pool Study			
	N	Median	Mean	UCL	N	Median	Mean	UCL
Limited Contact Scenarios								
Boating	316	2.1	3.7	11.2	0	-	-	-
Canoeing	766				76			
no capsize		2.2	3.8	11.4		2.1	3.6	11.0
with capsize		3.6	6.0	19.9		3.9	6.6	22.4
all activities		2.3	3.9	11.8		2.6	4.4	14.1
Fishing	600	2.0	3.6	10.8	121	2.0	3.5	10.6
Kayaking	801				104			
no capsize		2.2	3.8	11.4		2.1	3.6	10.9
with capsize		2.9	5.0	16.5		4.8	7.9	26.8
all activities		2.3	3.8	11.6		3.1	5.2	17.0
Rowing	222				0			
no capsize		2.3	3.9	11.8		-	-	-
with capsize		2.0	3.5	10.6		-	-	-
all activities		2.3	3.9	11.8		-	-	-
Wading/splashing	0	-	-	-	112	2.2	3.7	1.0
Walking	0	-	-	-	23	2.0	3.5	1.0
Full Contact Scenarios								
Immersion	0	-	-	-	112	3.2	5.1	15.3
Swimming	0	-	-	-	114	6.0	10.0	34.8
TOTAL	2,705				662			

N = Number of participants.
UCL = Upper confidence limit (i.e. mean +1.96 × standard deviation).
- = No data.

Source: Dorevitch et al. (2011).

Table 3.2: Fishing Exposure

3.2.2.2 Boating There were approximately 28,713 boat trips made, with an average of 2.5 people per boat, and an average trip length of five hours. This

means that there are approximately 71,783 people that engage in boating activities at Geist during the recreational season, for a total of 358913 hours. The mean water ingestion rate for boating is 3.7 mL per hour with an upper 97.5% confidence interval of 11.2 (*Water quality recreational project, Geist Reservoir, Indianapolis, Indiana* 1965) (see table 3.3).

Table 3-93. Estimated Water Ingestion During Water Recreation Activities (mL/hr)

Activity	Surface Water Study				Swimming Pool Study			
	N	Median	Mean	UCL	N	Median	Mean	UCL
Limited Contact Scenarios								
Boating	316	2.1	3.7	11.2	0	-	-	-
Canoeing	766				76			
no capsize		2.2	3.8	11.4		2.1	3.6	11.0
with capsize		3.6	6.0	19.9		3.9	6.6	22.4
all activities		2.3	3.9	11.8		2.6	4.4	14.1
Fishing	600	2.0	3.6	10.8	121	2.0	3.5	10.6
Kayaking	801				104			
no capsize		2.2	3.8	11.4		2.1	3.6	10.9
with capsize		2.9	5.0	16.5		4.8	7.9	26.8
all activities		2.3	3.8	11.6		3.1	5.2	17.0
Rowing	222				0			
no capsize		2.3	3.9	11.8		-	-	-
with capsize		2.0	3.5	10.6		-	-	-
all activities		2.3	3.9	11.8		-	-	-
Wading/splashing	0	-	-	-	112	2.2	3.7	1.0
Walking	0	-	-	-	23	2.0	3.5	1.0
Full Contact Scenarios								
Immersion	0	-	-	-	112	3.2	5.1	15.3
Swimming	0	-	-	-	114	6.0	10.0	34.8
TOTAL	2,705				662			

N = Number of participants.
UCL = Upper confidence limit (i.e. mean +1.96 × standard deviation).
- = No data.

Source: Dorevitch et al. (2011).

Table 3.3: Boating Exposure

3.2.2.3 Swimming There was no data available about the number of people who visited Geist that engaged in swimming. We made the assumption that on average, 1.5 people from every boat out for boating activities would swim, one person from every boat out for fishing would swim, and 0.5 people fishing from the shore would swim. This equals approximately 43,070 people from boating activities, 28,710 from people fishing from boats, and 30,048 people fishing from the bank, for a total of 101,828 people swimming during the season. We assume everyone swims once per year, and has an ingestion rate of 16 mL (upper 97% confidence interval of 90) per swim "event" for adults, and 37 mL (upper 97% confidence interval of 53) per swim "event" for children (*Exposure factors handbook, 2011 edition* 2011) (see table 3.4).

Chapter 3—Ingestion of Water and Other Select Liquids

Table 3-5. Recommended Values for Water Ingestion While Swimming

Age Group	Mean		Upper Percentile	
	mL/event[a]	mL/hour	mL/event[a]	mL/hour
Children	37	49	90[b]	120[b]
Adults	16	21	53[c]	71[c]

[a] Participants swam for 45 minutes.
[b] 97th percentile.
[c] Based on maximum value.

Source: Dufour et al. (2006).

Table 3.4: Swimming Exposure

4 Toxicological Assessment

4.1 Human Health Model

Cyanobacteria have been documented to release toxins including Microcystins, Anabaena, Oscillatoria, Aphanizomenon, Cylindrospermum, Cylindrospermopsis, Nostoc and Nodularia (Bell and G. A. Codd, 1994). Human effects and symptoms due to cyanobacteria exposure are diverse, including nausea, vomiting, flu-like symptoms, such as sore throat and fevers, kidney damage, and liver damage. Some of the toxins are also tumor promoter, which increases the probability of liver cancer when exposed to cyanobacteria in drinking water (G. Codd and Beattie, 1991). The first documented disease associated with cyanobacteria bloom was in London, where a female experienced severe abdominal pain after 12-hours of drinking some water with concentrated cyanobacteria. Additionally, the first documented death due to cyanobacterial poisoning was in Australia, where the livestock in a farm including sheep, cattle and horses died after drinking water from a lake with a bloom of Nodularia spumigena, which is one type of cyanobacteria (Hunter, 1998). Another study (Pilotto et al., 1997) documented studied a group of people who had swum in a contaminated lake. Symptoms included diarrhea, vomiting, skin rashes, mouth ulcers, fevers and eye or ear irritations.

Toxins released by cyanobacteria can be categorized by their impact on physiological systems and the potential target organisms or effects on humans. Based on this categorization, four main toxins have been documented, which are hepatotoxins, neurotoxins, cytotoxins and irritants and gastrointestinal toxins; the toxins that have been mostly studied are hepatotoxins and neurotoxins (G. Codd and Beattie, 1991).

The hepatotoxins are medium molecular weight peptide toxins, and microcystin is one of them. Microcystin is a seven amino acid ring, with several unique components in the microsystem. There are many molecular forms of microcystins, and microcystin LR is the most common structure related to animal poisoning (G. Codd and Beattie, 1991). The process for hepatotoxins to be transported into human bodies has three main steps: adsorption by the ileum cells,transported by blood to the liver organ, and then finally being taken up by the hepatocytes (Geoffrry A Codd et al., 1999). These toxins have a large effect on the human liver, by inhibiting the protein phosphatases—enzymes that are essential to human cell growth and suppression of tumors. Reported human symptoms include weakness, anorexia and pallor of the mucous membranes (Carmichael, 1992), while death is resulted from intrahepatic haemorrhage and hypovolaemic shock. Reported LD50 in mice is 50-150 µg/kg body weight by injection and 5000 µg/kg body weight by oral intake, where the toxicity from oral intake is one times higher than that by injection (Hunter, 1998).

Neurotoxins released by cyanobacteria are a producer of depolarizing neuromuscular blockade, and may cause potent nicotinic antagonistic effects. Acute poisoning in animals usually includes fasciculations, exaggerated abdominal breathing, collapse and convulsions, while fatalities are caused by respiratory failure. The toxicity reported by animal testing is lower than that of hepatotoxins. The LD50 is 375 µg/kg body weight by injection and 5000 µg/kg body weight by oral input (Hunter, 1998).

4.2 Measurement of Toxicity

For our analysis, we construct reference dose (RfD) for non-carcinogenic effects from microcystins, which is a measure of the average daily dose of a chemical associated with a minimum level of risk. In our analysis, we apply benchmark dose to replace the traditional NOAEL or LOAEL values to calculate the reference doses.

We calculated benchmark doses both for different combination of dose type and negative health outcomes.. The oral intake data is from (Fawell et al., 1999), while the injection data is from (Hooser et al., 1989). We also incorporate the background value that the individuals would die or have some targeted effects when the dose is zero. To acquire the positive benchmark doses, we used a lognormal probit model. Finally, we constructed the confidence interval for benchmark doses and picked the lower limit as our threshold. The effects we analyzed include fatalities, liver lesions, and specific liver damage problems, such as hepatocyte deposits, hepatocyte vacuolation and hepatocyte generation. The final results with different effects are demonstrated in Table 4.1 below. Furthermore, we incorporated uncertainty factors to extrapolate these effects from animal studies to humans.

The oral intake data is from (Fawell et al., 1999), while the injection data is from (Hooser et al., 1989).

Exposure Time	Intake Type	Response	BMD (µg/kg or µg/kg/day)
Acute	Oral	Death	610.01
Acute	Oral	Liver Lesions	256.2
Acute	Injection	Death	32.98
Chronic	Oral	Chronic Inflammation	16.08
Chronic	Oral	Chronic Congestion	2.17E-280
Chronic	Oral	Hepatocyte Deposits	153.13
Chronic	Oral	Hepatocyte Vacuolation	0.000000465
Chronic	Oral	Hepatocyte Generation	114.75

Table 4.1: Calculated benchmark doses

4.3 Ecological Model

4.3.1 Introduction

Microcystin released from cyanobacteria poses a threat to wildlife and pets near Geist reservoir, as it is toxic for all animals and fatal in most cases. The area is home to several endangered species from multiple taxa (see Table 4.3). Of particular concern is exposure and risk to the bird population near the reservoir. Bird deaths have been linked to cyanobacterial blooms in North America since the beginning of the 20th century. The severity of such events has varied from a small number of individuals, to several thousand animals. Domestic animal poisonings have been linked to blooms in seven states. In the vicinity of Geist reservoir exist livestock, domestic pets, and three species of special concern. Most of the endangered and threatened species are birds, though some plants and mollusks are on the list as well. The only species that is both on the federal and Indiana state endangered species lists is the clubshell mollusk. In terms of livestock, the area contains 4766 acres of pastureland. See table (insert number) for a list of endangered and threatened species in the area.

4.3.2 Overview

4.3.2.1 Fish Microcystins are toxic to fish at concentrations as low as a few micrograms perliter, and possibly even fractional. Fish will either ingest cyanobacteria or prey hat have fed on cyanobacteria. They can absorb the toxins directly from the water as well. Microcystins disrupt normal cellular activity in the livers of fish, leading to widespread cell death and loss of liver structure. Fish have a limited capacity to detoxify microcystins, and thus are succumb easily to their toxic effects. Mature fish do appear to be less sensitive to acute microcystin toxicity, and have a higher tolerance for oral exposure than injection (to the point of there being no oral LD50 values found in carp). Developing fish appear to be sensitive to chronic exposures to microcystins. Sublethal effects over a period of time can add up to inhibiting the ability of fish to continue functioning effectively. See table 4.2.

Fish	Total Dose (ug MC/kg)	Sublethal Effect
Carp (adult)	40	Widespread liver damage
Carp (adult)	1400	Severe liver damage
Carp (juvenile)	400	Severe liver and kidney damage
Trout	4400	Severe liver damage
Perch	9200	Severe liver damage
Tilapia	25200	Significant oxidative stress in liver

Table 4.2: Sublethal Effects for Fish

4.3.2.2 Birds Bird deaths have been linked to cyanobacterial blooms in Canada and the United States since the early 1900s. The severity of bird kills from these events have ranged from a few individuals to several thousand birds. There is regrettably very little experimental work done on birds, but what has been done demonstrated a lower LD50 than what has been demonstrated for mice. In the following analysis, a toxicological study on quail will be used to extrapolate toxicity data to endangered birds (Takahashi and Kaya, 1993).

4.3.2.3 Mammals Some mammals appear to be attracted to cyanobacteria in water, and livestock and dogs have been observed to drink infested water while clean water was plainly accessible, avidly consuming crust and mats. Mice have shown clear preference for Microcystis aeruginosa scum over clean drinking water. This increases the likelihood of oral exposure. Animals that go swimming in the lake during a bloom, such as dogs, will risk damage from dermal contact as while as drinking the water.

4.3.2.4 Mollusks No experimental data is presently available on mollusks. As several species of endangered mollusk do live in the Geist reservoir, it should be assumed that they could be impacted.

4.3.3 Toxicity risk for wildlife

Extrapolation across species was done mainly using body-weight. Body weight ranges are included below.

Type	Common Name	State Status	Average Adult Weight Range
Bird	Loggerhead Shrike	Endangered	35-50g
Bird	Least Bittern	Endangered	51-102g
Bird	Red-Shouldered Hawk	Species of Special Concern	486-774g
Bird	Osprey	Endangered	1400-2000g
Bird	Black-crowned Night Heron	Endangered	727-1014g
Bird	King Rail	Endangered	305-370 g
Bird	Cerulean Warbler	Endangered	8-10g
Bird	Upland Sandpiper	Endangered	97-226g
Mammal	American Badger	Species of Special Concern	4 to 12kg
Mammal	Bobcat	Species of Special Concern	4 to 18.3kg
Mammal	Least Weasel	Species of Special Concern	680 to 3968.93g

Table 4.3: Wildlife Mass Ranges

5 Simulation

This analysis section describes the calculations that were used to build a large set of simulated outcomes. Note that if more complex information is desired, the code included in the appendix is heavily commented.

5.1 Symbology

5.1.1 Units Table

Variable	Units	Symbol
Phosphorus	$\frac{mg}{L}$	P
Cyanobacteria Population	$\frac{cells}{mL}$	N
Microcystin Concentration	$\frac{\mu g}{L}$	C
Time	POSIX, days	t
Average Daily Wind-speed	tenths of $\frac{m}{s}$	W
Daily Temperature Midpoint	tenths of C°	T
Dose/Human Dose Equivalent	Acute: $\frac{\mu g}{kg * event}$; Chronic: $\frac{\mu g}{kg * day}$	D
Exposure	Acute:mL , $\frac{mL}{hour}$; Chronic: $\frac{mL}{day}$	E
Exposure Time	Acute: hours (if exposure is per hour)	E_t
Body Mass	kg	M
Doubling time	$\frac{1}{days}$	d
Instantaneous growth rate	$\frac{cells}{mL}$	r
Carrying capacity	$\frac{cells}{mL}$	K

Table 5.1: Variables

5.1.2 Suffixes/Subscripts

The numbering system below keeps track of stages of data processing. Period (0-1) is the period in which we estimate model parameters. In period (1-2), the stepwise evolution of cyanobacteria populations in simulated. In period (2-3), the step-wise evolution of microcystin levels is simulated.

Subscript/Suffix	Meaning
0	Value at start of time period
0.5, no number	Averaged value over time period
1	Value at end of time period
1.5	Projected averaged value within the next time period
2	Projected value at the end of the next time period
2.5	Projected averaged value within the third time period
3	Projected averaged value at the end of the third time period
s	Variable subtracted from its mean (standardized)
L	The log of a variable
sd	The standard deviation of a variable

Table 5.2: Conventions

5.2 Cyanobacteria Growth Model

5.2.1 Alternatives

Non-linear estimation was attempted in several configurations. The failure of results to converge in most cases, combined with the potential for multiple equilibria, ruled this out as an option. Attempts to combine coefficients from multiple models into compound linear coefficients also proved impossible.

5.2.2 Data Description

Data comes from two sources.

Data from Geist was given to us by Melissa Clark and gathered by Citizen's Energy, the utility company which provides drinking water to Indianapolis. Samples are taken approximately every 14 days over the summer months of three years. In 2010, data is from April to October. In 2011, data is from June to August. In 2012, data is from June to October.

Climate data is taken from NOAA's NCDC network. Data is queried for every day between the first date and the last date in the Geist data. Then, variables are averaged over the periods in the Geist data, and finally, the two data-sets are merged together.

5.2.3 Predict Cyanobacteria Populations

5.2.3.1 Bootstrap The first step we need to do is bootstrap the Geist Data. Each of our Monte-Carlo simulations will be associated with its own bootstrap of the Geist Data. This will allow us to automatically build uncertainty into all calculations involving this set of data.

5.2.3.2 Logistic Growth We will use the basic assumption of logistic growth, that is,

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right)$$

A quick transformation to logs simplifies analysis.

$$\frac{1}{N}\frac{dN}{dt} = r\left(1 - \frac{N}{K}\right)$$

$$\frac{d\left(log(N)\right)}{dt} = r\left(1 - \frac{N}{K}\right)$$

$$\frac{dN_L}{dt} = r\left(1 - \frac{N}{K}\right)$$

This equation modifies a simple proportional growth rate by penalizing populations close to a carrying capacity. At equilibrium, N will always equal K.

5.2.3.2.1 Find Logistic Growth Rates Assume also that a value of r is a constant regardless of the species of bacteria and environmental variables. r represents the maximum rate of growth, and can be viewed as a biological constraint. This assumption has been used elsewhere. According to literature, cyanobacteria can achieve maximum doubling times of between 0.3 to 1.4 per day (Kaushik, 1987). We will assume a uniform distribution.

At maximum growth rates:

$$N = N_0 * 2^{d*t} = N_0 * e^{ln(2)*t/d}$$

$$max\left(\frac{dN}{dt}\right) = \frac{ln(2)}{d} * N$$

Compare to the logistic equation

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right)$$

Which reaches its maximum when $K = 2N$, or

$$max\left(\frac{dN}{dt}\right) = rN\left(1 - \frac{N}{2N}\right) = \frac{rN}{2}$$

We can set these equal:

$$\frac{ln(2)}{d} * N = \frac{rN}{2}$$

$$r = 2 * \frac{ln(2)}{d}$$

This allows us to back-solve for the logistic rate of growth given a doubling time.

5.2.3.2.2 Find Carrying Capacity To back-solve for K, it is necessary to distribute and discretize the differential equation. This discretization is problematic due to varying conditions over 2 week time steps, but nevertheless necessary unless data on a smaller time-scale becomes available.

$$\frac{\Delta N_L}{\Delta t} = r - \frac{rN_0}{K}$$

Then solve for K.

$$\frac{rN_0}{K} = r - \frac{\Delta N_L}{\Delta t}$$

$$K = \frac{rN_o}{r - \frac{\Delta N_L}{\Delta t}}$$

We will also need, for continuity, to truncate K at 1: that is, the carrying capacity cannot go below $\frac{1\ cell}{mL}$.

This estimate of K can be calculated for every consecutive pair of data points in the data separated by a reasonably small time. We assume that K, carrying capacity, would be a multiplicative function of environmental variables. The value of these environmental variables should represent conditions throughout that time interval in order to adequately explain flows.

5.2.3.3 Model Carrying Capacity The model used is below.

$$K_L = \beta_0 + \beta_1 P_{Ls} + \beta_2 P_{Ls}^2 + \beta_5 W_{Ls} + \beta_6 W_{Ls}^2 + \beta_3 T_s + \beta_4 {T_s}^2 + \epsilon$$

(see subscript table above)

The environmental variables we chose are P (phosphorus concentrations), T (temperature), and W (windspeed). These parameters were available in the data and also are suspected of importance to cyanobacteria growth in the literature (see above). The values of these variables should represent an aggregate of information about the variables within the time step. Therefore, the mean was taken of all values for these variables available within each time step.

We decided to use a multiplicative function to estimate carrying capacity. Note, we are using the log of all variables except for temperature. This is because, at least within the given range of values, temperature does not have a true zero. We wanted to include the squared terms for the following reasons. We suspect that there is an optimum temperature level, as well as an optimum wind speed, at which each species of cyanobacteria will thrive. An optimum wind speed will allow mixing of nutrients in the lake while still being slow enough for stable cyanobacteria mats. Squared terms were necessary to incorporate this. Subtraction from the mean allows for a better estimation of squared terms.

Note that models were estimated for each bootstrap and each clade of cyanobacteria. We also included estimates of error term standard deviation, and incorporated a randomly generated error term in the model bellow.

5.2.3.4 Simulate Carrying Capacity and Cyanobacteria Populations

Note, this will be considered period (1-2).

5.2.3.4.1 Specification tests

The initial values for cyanobacteria population and microcystin concentration (needed due to the stepwise models), and values of three environmental variables, wind-speed, phosphorus, and temperature, will all be simulated. Thus, we need to test the distribution of these variables, and as well, we will test whether they are correlated with one another. Note that variables which will be logged (everything but temperature) are assumed to have a log normal distribution, and thus plots include the log of most variables.

First, plot all variables but population (which needs to separated by clade).

```
`Cyanobacteria Specification Graph`
```

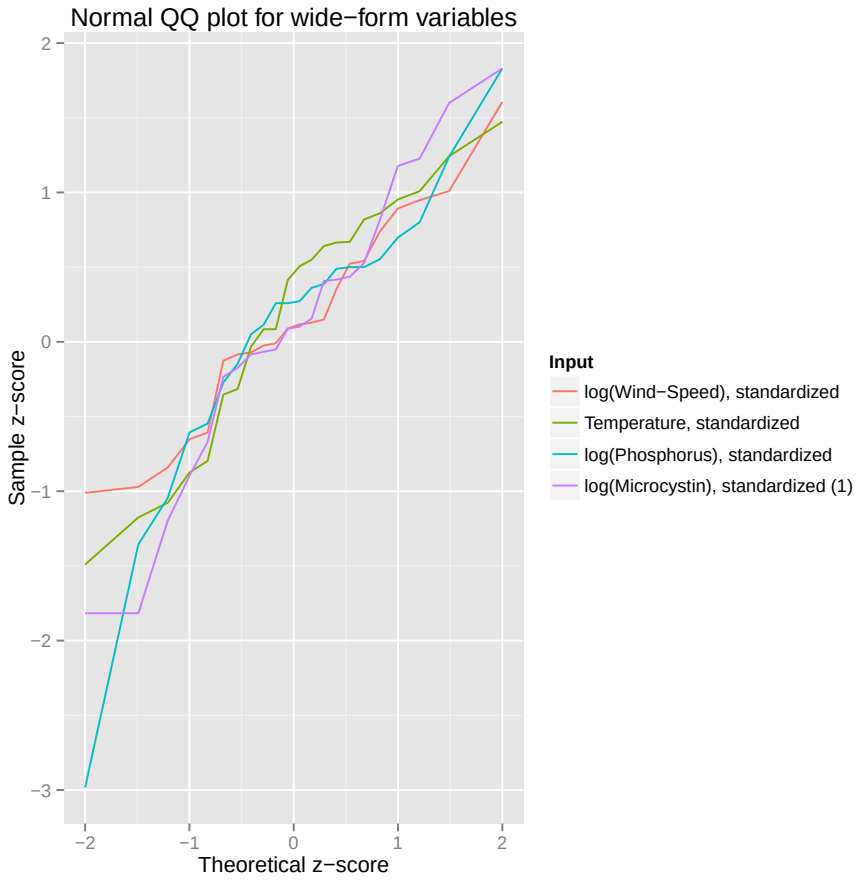

Figure 5.1: Normal QQ plot for several variables. See suffix table for variable interpretation. Values close to aligning with the y=x line indicate a normal distribution.

The wide-variable set seems decently normal, minus a few outliers on the far left side.

Now plot populations separated by clade.

```
`Cyanobacteria Clade Specification Plot`
```

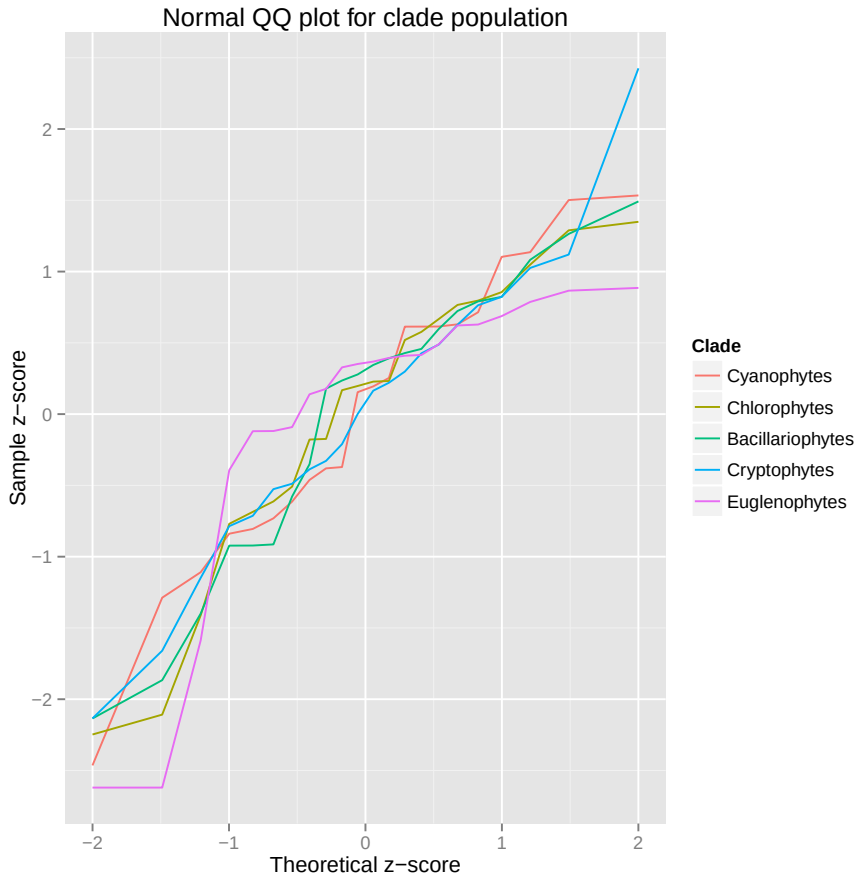

Figure 5.2: Normal QQ plot for clade populations. Values close to aligning with the y=x line indicate a normal distribution.

The clade population distributions also seem close to a log-normal distribution. The flat tail on the left for Euglenophytes was due to a short series of values of 0 in the last few weeks of data. This is anomalous but not a large concern.

Covariance was calculated between each of these predictor variables. However, none of the correlations passed a 5% significance threshold, and thus, covariance was ignored (i.e., predictors are assumed to be independent). Had we more time, we would like have liked to incorporate covariance estimates into the model via bootstrapping.

5.2.3.4.2 Simulate inputs For each bootstrap, we need to calculate distribution parameters. Since we are assuming normal/log normal distributions, we can thus need to calculate means and standard deviations first. Then, we will use these parameters to randomly generate feasible values of the input values above.

5.2.3.4.3 Incorporate STEPL data We need to take all predicted phosphorus values and reduce them proportionally by the percent reduction given in STEPL. We assume that runoff phosphorus is proportional to the phosphorus levels in Geist. There are three scenarios, no BMPs, recommended BMPs, and 100% BMPs. We cannot take into account uncertainty in STEPL because the practicality of the complex model.

5.2.3.4.4 Simulate The modeling in this cyanobacteria section will work as follows. First, select a starting population value. Then, project out one interval, using environmental variables and the initial value of population. This will be considered interval 2. Predict carrying capacity using the carrying capacity model, then predict the population growth rates using the logistic equation. The interval lasts 14 days, which aligns with the time periods in the Geist data.

5.2.4 Predict Microcystin

5.2.4.1 Estimate Model We will predict the ratio of cyanotoxin concentrations based on a multiplicative function of the concentration of different clades of cyanobacteria:

$$\frac{C_1}{C_0} = \left(N_1^{\beta_1} N_2^{\beta_2} ... \, e^{\epsilon}\right)^{\Delta t}$$

Taking logs:

$$\frac{log\,(C_1) - log\,(C_0)}{\Delta t} = \beta_{N_1} log\,(N_1) + \beta_{N_2} log\,(N_2) ... + \epsilon$$

or, instantized,
$\frac{dC_L}{dt} = \frac{1}{C}\frac{dC}{dt} = \beta_{N_1} log\,(N_1) + \beta_{N_2} log\,(N_1) ...$
Thus, the rate of microcystin growth can be estimated by cyanobacteria populations at the start of the period. We would have liked also to have included environmental variables in this model, but there does not seem to be a strong indication in the literature of which to use, and none were statistically significant.

5.2.4.2 Simulate Microcystin Note, this will be considered period (2-3)

5.2.4.2.1 Simulate prediction variables Calculate the mean and standard deviation for each clade of bacteria, and then randomly generate a value from the normal distribution for each value.

5.2.4.2.2 Simulate Use the simulated initial microcystin concentration, and simulated populations of cyanobacteria from the end of the previous step, predict the growth rate of the log of microcystin over the simulated period. This will be considered period 2-3.

5.3 Exposure

A somewhat novel methodology was used for this section.

5.3.0.3 Body Mass Body mass for each species and each Monte-Carlo simulation was simulated assuming a log normal distribution. This was relatively easy to do given specific quantiles in the form of intervals. Human mass distribution data was taken from (*Exposure factors handbook, 2011 edition* 2011). A range was used that included all ages of humans.

5.3.0.4 Drinking Water Next, drinking water requirements were predicted using mass. This methodology is explained in (*Animal weights and their food and water requirements* 2001). Two models were used, one for birds, and one for mammals. Generally, drinking water requirements $\left(in\ \frac{mL}{day}\right)$ is a multiplicative function of weight in kg. Unfortunately, it was impossible simulate the error in these predictions.

$$daily\ drinking\ water = a * mass^b$$

5.3.0.5 Exposure Rates Recreational exposures were randomly simulated assuming a log normal distribution using the confidence intervals found previously. Drinking water exposure rates, of course, were calculated above.

5.3.1 Dose

Dose proved relatively simply to calculated. Note that for chronic acute exposure, an extra multiplication step is required for time exposed.

$$Dose = \frac{Concentration * Exposure * (Exposure\ Time,\ for\ acute)}{Mass}$$

For acute

$$\frac{\mu g}{kg} = \frac{\frac{\mu g}{L} * \frac{mL}{hour} * hours * \frac{1\ L}{1000\ mL}}{kg}\ or\ \frac{\frac{\mu g}{L} * mL * \frac{1\ L}{1000\ mL}}{kg}$$

For chronic,

$$\frac{\mu g}{kg * day} = \frac{\frac{\mu g}{L} * \frac{mL}{day} * \frac{1\ L}{1000\ mL}}{kg}$$

5.3.1.1 Species to Species Conversion Species to species dose conversion was also conducted using body mass. This methodology is explained in (*Guidance for Industry: Estimating the maximum safe starting dose in initial clinical trials for therapeutics in adult healthy volunteers* 2005). The general equation is as follows.

$$model\ animal\ equivalent\ dose = end{-}point\ animal\ dose \left(\frac{end - point\ animal\ mass}{model\ animal\ mass}\right)^{0.25}$$

This non-intuitive equation has been empirically validated (*Guidance for Industry: Estimating the maximum safe starting dose in initial clinical trials for therapeutics in adult healthy volunteers* 2005). Unfortunately, again, it was impossible simulate the error in these predictions. Note that the model animal for all mammals was mice, and the model animal for all birds was quails.

5.4 Toxicology

We will using the general model underling benchmark doses learned in class. We will simulate a logit (not the very similar probit) model for convenience. Our basic model will be:

$$odds = \frac{Pr}{1 - Pr}$$

$$log(odds) = \beta_0 + \beta_1 log(dose + 1)$$

Note that the + 1 unfortunately is required to prevent logging 0. We will again bootstrap the toxicology the toxic data, so that each Monte-Carlo simulation is associated with a bootstrap of the toxicological data. This allows us to take into account the uncertainty in the data. After the model is built for each boot-strap, it will be used to simulate log odds for each scenario.

6 Risk Evaluation

Because the tables for this section would not fit in the report, they are attached separately.

6.1 Expected Value Table

The attached "Casualty Table" lists, for every combination of scenarios, an expected value for the number of people who will develop various negative health outcomes. The numbers are very high. More than anything, this should be cause for concern.

6.2 Interval Table

The attached "Interval Table" lists 2.5% and 97.5% boundaries for the log of the odds ratio of any given individual developing a negative health outcome. Note, however, that this variation includes alleatoric uncertainty as well as natural variation between individuals. The boundaries are so wide that they call into question nearly every quantitative result in this report. However, in a certain sense, understanding such huge variation is highly important. Policy makers need to understand that because of this uncertainty, catastrophic outcomes are quite possible unless other information suggests otherwise.

6.3 Program Evaluation Table

This table displays 2.5% and 97.5% boundaries for the percent decrease in the odds of a suite of negative outcomes for individuals following BMP plan implementation. The boundaries include 0, which suggests that we cannot be certain of the efficacy the BMP plans. However, changes in either direction are very small, suggesting against using BMP plans at all.

7 Discussion

7.1 Economic Implications

Along with human health and ecological impacts, a cyanotoxin release into Geist Reservoir would also have economic implications for the community. If the lake is monitored, these economic implications will be more likely to occur than severe human health effects. The toxin release would affect both recreational activities as well as the drinking water supply.

7.2 Recreational Activities

If a toxin release occurs in Geist Reservoir, there will be a tradeoff between reduced recreation opportunities or increased health risks from people continuing to use Geist for recreation. Other incidences of toxic releases into lakes used for recreation have led to huge losses in revenues generated from recreational activities (Hudnell, 2008). For example, a 1992 study in New South Wales considered the costs associated with tourism, and long-term effects and found that with 9 reservoirs affected, the total cost was AU$ 1.2 million (Walker et al., 1996). The costs associated with a bloom in Geist Reservoir would come from reduced fishing, boating, swimming, and secondary costs associated with tourism. One incident would only incur these costs once, recurring incidents would have longer term effects such as an overall decline in tourism to the area (Hudnell, 2008).

If a toxin release occurs in Geist, there will be significant costs associated with the contaminated drinking water supply. There will be a tradeoff between treating the water or completely shutting down the reservoir and getting a drinking water supply from elsewhere (Cheung, Liang, and Lee, 2013). It is

difficult to put a value on the loss of drinking water for an area, but the most accurate method is to use the costs of treatment of the water. Treatment options range from artificial mixing of the reservoir to algicides, and can be very costly (Hudnell, 2008). An Australian study found that to treat a reservoir with algicides costs from AU$20,000 to AU$50,000 per treatment. Artificial mixing can cost up to AU$100,000 to install and run a system (Hudnell, 2008). The other option is to pull in water supplies from alternate sources if a toxin release occurs. This is feasible in Indianapolis because the water supply comes from three different reservoirs and so the costs would be low. In some areas, this can be a very costly option; if no other options exist then water would need to be shipped in at a high cost (Hudnell, 2008).

7.3 Risk Management

7.3.1 Future Steps

To better understand how to predict and manage the risk associated with algal blooms, we recommend the following three strategic steps:

7.3.1.1 Additional Data Collection We recommend a fairly substantial amount of additional data collection related to various parts of our analysis. Most significantly, we recommend a study to understand the water movement and cycling within Geist Reservoir. This will allow us to model the transport of nutrients and toxins, and may allow us to better predict where cyanobacteria blooms will occur within the reservoir. If this is coupled with more updated and specific information on how the reservoir is used recreationally, and which parts of the reservoir are typically used for which activities, we will be able to better model the risk of exposure to toxins within the lake.

We also recommend more thorough and regular monitoring at Geist Reservoir to create a more substantial dataset regarding water quality parameters. This would allow us to establish trends more easily, and determine if there are relationships between environmental factors and inputs, with algal blooms and toxin release.

7.3.1.2 BMP Implementation Secondly, we recommend additional research into where BMPs are possible in the watershed. There is a distinction to be made here between politically possible, and ecologically or technically possible. What we are proposing to do is assess where BMPs are technically possible, to establish a realistic "best" outcome when running STEPL. This involves understanding what BMPs can be applied in specific areas and instances. While most BMPs are utilized in agricultural settings, they can also be used in urban and forest landscapes. As discussed earlier, the goal of BMPs are to reduce the runoff of nutrients into water bodies of concern. It is important to assess what pollutants specifically need to be reduced in choosing the BMP, as some are more effective than others at removing different pollutants. Below is

a table highlighting the results of a literature review that show the reductions associated with specific BMPs.

Table 11. Reduction in Water Quality Metrics Given Individual Conservation Practice Implementation

	Total Nitrogen	Organic Phosphorus	Soluble Phosphorus	Total Phosphorus	Total Sediment
Brush Management[a]	-1 – 37%	-	-	-8 - 42%	-40 - 64%
Contour Terrace[a]	-56 – 59%	-	-	-60 - 65%	-84 - 86%
Cover Crop (25%, 50%, 75%)[b]	-10%; -21%; -27%	-	-	-11%; -21%; -26%	-8%; -19% -24%
Critical Area Planting[a]	-90 – 96%	-	-	-82 – 95%	-98 – 99%
Crop Residue Management[a]	-14 – 36%	-	-	-12 - 25%	-29 - 41%
Fertilizer Reduction (25%, 50%, 75%)[b]	-0.70%; -1.50%; -1.90%	-	-1.40%; -1.70%; -0.70%	-	0%; -0.90%; -1.1%
Filter Strip (25%, 50%, 75%)[b]	-22%; -34%; -43%	-	-	-23%; -36%; -44%	-20%; -35%; -43%
Filter Strip + No-Till (25%, 50%, 75%)[b]	-23%; -40%; -61%	-	-	-24%; -40%; -62%	-23%; -40%; -65%
Forage Harvest Management[a]	-4 – 23%	-	-	-1 - 11%	-21 - 76%
Grade Stabilization Structure[a]	-95 – 98%	-	-	-93 – 97%	-98 – 99%
Grass and Riparian Filters[c]	-	-	-50%	-60%	-55 - 82%
High-P Application with Composting[d]	-	-83%	-86%	-86%	-
N-Based Application with Composting[d]	-	-78%	-83%	-81%	-
No-Till (25%, 50%, 75%)[b]	-11%; -25%; -41%	-	-	-11%; -25%; -42%	-13%; -33%; -50%
Nutrient Management[e] **(Plan)**[a]	(-36%)[e] (-77 - 93%)[a]	-	-	(-59%)[e] (-53 - 78%)[a]	-
Range Seeding[a]	-89 - 92%	-	-	-77 – 88%	-97 - 98%
Riparian Buffer (200 feet)[f]	-1.7%	-	-	-11%	-12%
Roadside Tree Buffer (10 feet)[f]	-26%	-	-	-11%	-13%
Roadside + Riparian Buffer[f]	-26%	-	-	-20%	-25%

a = (Santhi, et al., 2006); **b** = (Kieser & Associates, 2008); **c** = (Daniels & Gilliam, 1996); **d** = (Osei, et al., 2000); **e** = (Agriculture and Agri-Food Canada, 2012); **f** = (Matteo, Randhir & Bloniarz, 2006)

- Not Measured

Table 7.1: BMPs table

This research can help inform which BMPs may be most beneficial if more than one are ecologically possible in an area. We also recommend working on outreach initiatives to encourage landowners and municipalities to implement BMPs. Most people's concern and hesitation in pursuing BMPs is associated with the cost. Fortunately, there are programs where funding can be acquired to implement BMPs, particularly in agricultural land. One example is the United

States Department of Agriculture's (USDA) Natural Resource Conservation Service's (NRCS) National Water Quality Initiative (NWQI). The NWQI provides funding to states, who then solicit applications for farmers in specific watersheds. They help the farmer evaluate which BMPs can be used and would be effective, and provide funding to implement the BMPs. This is one option of many that can be used to help reduce the worry associated with cost in implementing the BMPs. Additionally, the argument can be made, and supported by literature, that BMPs help reduce costs for municipalities and states by mitigating the risks associated with blooms and losing a source of drinking water or recreational income. BMPs can also be used to improve water quality as mandated by the Clean Water Act, and Total Maximum Daily Loads (TMDLs). All of these benefits are associated with economic benefits and savings, and should be highlighted to encourage further BMP implementation.

7.3.1.3 Collaboration with Subwatershed Communities Finally, to accomplish the previous two mentioned recommendations, we recommend facilitating and encouraging collaboration between the subwatershed communities and municipalities. This will both reduce costs associated with accomplishing these recommendations, as well as increase efficiency and effectiveness by reducing duplication of efforts and increasing resources available to everyone. This will also help the entire watershed work as a whole to focus on significant sources of nutrient input, and target those areas for reduction efforts and BMP implementation.

7.3.2 Emergency Management

Emergency plans should be put into place in the event a bloom does occur. As mentioned earlier in the report, should toxins actually be released, Geist would need to be isolated from the public and animals until the toxins break down. The Environmental Protection Agency reviewed and suggested a series of management methods in the event of a bloom and subsequent toxin release. EPA recommendations can be broken down into physical, chemical, and biological controls. The tables attached tables were generated by the EPA to guide managers in choosing a plan. Note that policy makers have several viable options even after a cyanobacteria bloom event.

See attached: "waterbody management plans.csv", "cyanobacteria treatment processes.csv".

8 Works Cited

References

1990 United States Census, summary (1993). Tech. rep. Government of the United States.

2013 drinking water report (2013). Tech. rep. Indianapolis, IN: Citizen's Energy Group. URL: http://www.citizensenergygroup.com/pdf/2013WaterReport.pdf (visited on 05/08/2015).

Adler, Jonathan H. (2002). "Fables of the Cuyahoga: Reconstructing a history of environmental protection". In: *Fordham Environmental Law Journal* 14, p. 89. URL: http://heinonline.org/HOL/Page?handle=hein.journals/frdmev14&id=97&div=&collection=.

Alonso Fernández, J.R. et al. (2013). "Forecasting the cyanotoxins presence in fresh waters: A new model based on genetic algorithms combined with the MARS technique". In: *Ecological Engineering* 53, pp. 68–78. ISSN: 0925-8574. DOI: 10.1016/j.ecoleng.2012.12.015. URL: http://www.sciencedirect.com/science/article/pii/S0925857412003692.

Animal weights and their food and water requirements (2001). Tech. rep. British Columbia Mininstry of Environment. URL: http://www.env.gov.bc.ca/wat/wq/reference/foodandwater.html (visited on 05/08/2015).

Bell, S. G. and G. A. Codd (1994). "Cyanobacterial toxins and human health." In: *Reviews in Medical Microbiology* 5.4. ISSN: 0954-139X. URL: http://journals.lww.com/revmedmicrobiol/Fulltext/1994/10000/Cyanobacterial_toxins_and_human_health_.5.aspx.

Carmichael, W.W. (1992). "Cyanobacteria secondary metabolites—the cyanotoxins". In: *Journal of Applied Bacteriology* 72.6, pp. 445–459. ISSN: 1365-2672. DOI: 10.1111/j.1365-2672.1992.tb01858.x. URL: http://dx.doi.org/10.1111/j.1365-2672.1992.tb01858.x.

Carvalho, Laurence et al. (2011). "Cyanobacterial blooms: statistical models describing risk factors for national-scale lake assessment and lake management". In: *Science of the Total Environment* 409.24, pp. 5353–5358. URL: http://www.sciencedirect.com/science/article/pii/S004896971101031X (visited on 03/17/2015).

Cheung, MelissaY., Song Liang, and Jiyoung Lee (2013). "Toxin-producing cyanobacteria in freshwater: A review of the problems, impact on drinking water safety, and efforts for protecting public health". English. In: *Journal of Microbiology* 51.1, pp. 1–10. ISSN: 1225-8873. DOI: 10.1007/s12275-013-2549-3. URL: http://dx.doi.org/10.1007/s12275-013-2549-3.

Codd, GA and KA Beattie (1991). "Cyanobacteria (blue-green algae) and their toxins: awareness and action in the United Kingdom". In: *PHLS Microbiology Digest* 8.3, pp. 82–86.

Codd, Geoffrry A et al. (1999). "Cyanobacterial toxins, exposure routes and human health". In: *European Journal of Phycology* 34.04, pp. 405–415. ISSN: 1469-4433. URL: href=%22http://dx.doi.org/10.1017/S0967026299002255.

Cyanobacteria/cyanotoxins, US EPA (2015). URL: http://www2.epa.gov/nutrient-policy-data/cyanobacteriacyanotoxins (visited on 05/08/2015).

Exposure factors handbook, 2011 edition (2011). Tech. rep. US EPA. URL: http://www.epa.gov/ncea/efh/pdfs/efh-complete.pdf (visited on 05/08/2015).

Fawell, J K et al. (1999). "The toxicity of cyanobacterial toxins in the mouse: I Microcystin-LR". In: *Human & Experimental Toxicology* 18.3, pp. 162–167.

DOI: 10.1177/096032719901800305. URL: http://het.sagepub.com/content/18/3/162.abstract.

Garcia Nieto, P.J. et al. (2011). "Study of cyanotoxins presence from experimental cyanobacteria concentrations using a new data mining methodology based on multivariate adaptive regression splines in Trasona reservoir (Northern Spain)". In: *Journal of Hazardous Materials* 195, pp. 414–421. ISSN: 0304-3894. DOI: 10.1016/j.jhazmat.2011.08.061. URL: http://www.sciencedirect.com/science/article/pii/S030438941101079X.

Geist Reservoir/Upper Fall Creek Watershed management plan (2011). Tech. rep. Geist Watershed Alliance. URL: http://www.in.gov/dnr/fishwild/files/fw-Geist_Reservoir_Upper_Fall_Creek_Wtrshd_Mgmt_Plan_Del_Ham_Han_Hen_Mad_Mar_Counties_Feb_2011.pdf (visited on 05/08/2015).

Guidance for Industry: Estimating the maximum safe starting dose in initial clinical trials for therapeutics in adult healthy volunteers (2005). Tech. rep. US Food and Drug Administration. URL: http://www.fda.gov/downloads/Drugs/Guidances/UCM078932.pdf (visited on 05/08/2015).

Hooser, S. B. et al. (1989). "Toxicity of microcystin LR, a cyclic heptapeptide hepatotoxin from Microcystis aeruginosa, to rats and mice". In: *Veterinary Pathology Online* 26.3, pp. 246–252. DOI: 10.1177/030098588902600309. URL: http://vet.sagepub.com/content/26/3/246.abstract.

Hudnell, H Kenneth (2008). *Cyanobacterial harmful algal blooms: State of the science and research needs*. Springer Science & Business Media. ISBN: 0387758658.

Hunter (1998). "Cyanobacterial toxins and human health". In: *Journal of Applied Microbiology* 84.S1, 35S–40S. ISSN: 1365-2672. DOI: 10.1046/j.1365-2672.1998.0840s135S.x. URL: http://dx.doi.org/10.1046/j.1365-2672.1998.0840s135S.x.

Kaushik, B D (1987). *Laboratory methods for blue-green algae*. New Delhi: Pub. Assoc. Publishing Co.

Onderka, Milan (2007). "Correlations between several environmental factors affecting the bloom events of cyanobacteria in Liptovska Mara reservoir (Slovakia)—A simple regression model". In: *Ecological Modelling* 209.2, pp. 412–416. URL: http://www.sciencedirect.com/science/article/pii/S0304380007004048 (visited on 03/17/2015).

Persaud, Anurani D. et al. (2015). "Forecasting cyanobacteria dominance in Canadian temperate lakes". In: *Journal of environmental management* 151, pp. 343–352. URL: http://www.sciencedirect.com/science/article/pii/S0301479715000171 (visited on 03/17/2015).

Pilotto, Louis S. et al. (1997). "Health effects of exposure to cyanobacteria (blue–green algae) during recreational water–related activities". In: *Australian and New Zealand Journal of Public Health* 21.6, pp. 562–566. ISSN: 1753-6405. DOI: 10.1111/j.1467-842X.1997.tb01755.x. URL: http://dx.doi.org/10.1111/j.1467-842X.1997.tb01755.x.

Takahashi, Shinji and Kunimitsu Kaya (1993). "Quail spleen is enlarged by microcystin RR as a blue-green algal hepatotoxin". In: *Natural Toxins* 1.5, pp. 283–285. ISSN: 1522-7189. DOI: 10.1002/nt.2620010505. URL: http://dx.doi.org/10.1002/nt.2620010505.

Tao, Min et al. (2012). "Use of a generalized additive model to investigate key abiotic factors affecting microcystin cellular quotas in heavy bloom areas of Lake Taihu". In: *PloS one* 7.2, e32020. URL: http://dx.plos.org/10.1371/journal.pone.0032020 (visited on 03/17/2015).

Vilán Vilán, J.A. et al. (2013). "Support Vector Machines and Multilayer Perceptron Networks Used to Evaluate the Cyanotoxins Presence from Experimental Cyanobacteria Concentrations in the Trasona Reservoir (Northern Spain)". English. In: *Water Resources Management* 27.9, pp. 3457–3476. ISSN: 0920-4741. DOI: 10.1007/s11269-013-0358-4. URL: http://dx.doi.org/10.1007/s11269-013-0358-4.

Walker, Craig. et al. (1996). *The economic costs associated with lost recreation benefits due to blue-green algae in New South Wales : three case studies*. English. [Parramatta, N.S.W.]: Dept. of Water Resources, Technical Services Division.

Water quality recreational project, Geist Reservoir, Indianapolis, Indiana (1965). Tech. rep. US Department of Health, Education, and Welfare.

9 Appendix: List of attached tables

File	Type of Dataset	Description
Geist	Input	Information about Geist Reservoir
Tox	Input	Toxicology Data
Endpoints	Input	Information about end-point organisms
Models	Input	Information about model organisms
Exposure	Input	Information about exposed individuals
Expected Value Table	Output	A table of the expected values of individuals with negative health outcomes
Interval Table	Output	Interval estimates for the log odds of negative health outcomes
Program Evaluation	Output	The reduction of odds of negative health outcomes after BMP implementation
Waterbody Management Methods	Supplemental	Information about managing cyanobacteria blooms
Cyanobacteria Treatment Processes	Supplemental	Information about managing cyanobacteria blooms

Table 9.1: List of attachments

10 Appendix: Code

```
setwd("~/Dropbox/Risk/Project/Math")
```

```
Replications = 10000

library(pipeR)
library(lubridate)
library(rnoaa)
library(tidyr)
library(magrittr)
library(pipeR)
library(zoo)
library(ggplot2)
library(lazyeval)
library(drc)
library(bmd)
library(knitr)
library(stringi)
library(dplyr)

`lag of` = function(vector) {
  vector = c(vector[1], vector[-length(vector)])
  vector[1] = NA
  vector}

# Input geist data from Melissa Clark
`Geist Data` = read.csv("geist.csv", check.names = FALSE)

# Select the variables we are interested in.
`Geist Data Selection` =
  `Geist Data` %>%
  transmute(
    `Time (1)` = mdy(Date),
    `Phosphorus (1)` = `TP (mg/L)`,
    `Microcystin (1)` = `Microcystin (ug/L)`,
    Cyanophytes, Chlorophytes, Bacillariophytes,
    Cryptophytes, Euglenophytes) %>%

  # Filter out rows with missing data and create an ID variable
  filter(complete.cases(.)) %>%
  mutate(`Geist ID` = 1:n())

# For completeness, the code to download data from NOAA is included below.
# Data was downloaded for every day between the first and last day in the Geist data.

# token = "VMeBWzQcEkpqpTdiMqgPWbFATWMMPAGP"
#
# `Long Climate Data` = data.frame(
```

```
#   time_post = seq(
#     min(`Geist Data Selection`$`Time (1)`),
#     max(`Geist Data Selection`$`Time (1)`),
#     by = "day")) %>%
#   group_by(`Time (1)`) %>%
#   do(ncdc(
#     token = token,
#     datasetid='GHCND',
#     stationid= "GHCND:USW00093819",
#     startdate = as.character(first(`Geist Data Selection`$`Time (1)`)),
#     enddate = as.character(first(`Geist Data Selection`$`Time (1)`))) %>%
#       use_series(data)) %>%
#     ungroup %>%
#     rename(`Data Type` = datatype,
#             Value = value)

# Instead of redownloading the data, we are going to use a version from memory.
load("longClimateData.RData")

# The data needs to be converted into wide form by datatype
`Wide Climate Data` =
  `Long Climate Data` %>%
  select(`Time (1)`, `Data Type`, Value) %>%
  # Spread various data types into seperate columns
  spread(`Data Type`, Value, fill = 0)

# Find a temperature midpoint between the maximum and minimum daily temperature.
`Climate Data Selection` =
  `Wide Climate Data` %>%
  transmute(
    `Time (1)`,
    Temperature = (TMIN + TMAX)/2,
    `Wind-Speed` = AWND)

# Average temperature and windspeed data over the intervals in the Geist Data.
# Windspeed is logged first in preparation for log-log model below.
`Climate Data Summarized` =
  `Climate Data Selection` %>%
  # A tricky way to match dates into ( , ] intervals from Geist.
  left_join(`Geist Data Selection`) %>%
  mutate(`Geist ID` = na.locf(`Geist ID`, fromLast = TRUE)) %>%
  # Summarize over bins.
  group_by(`Geist ID`) %>%
  summarize(
    Temperature = mean(Temperature),
```

```
    `log(Wind-Speed)` = mean(log(`Wind-Speed`)))

# Finally, merge the Geist data and the climate data together.
`Wide Cyanobacteria Data` =
  full_join(`Geist Data Selection`, `Climate Data Summarized`)

# We are going to calculate logs, lags, averages, deltas, and derivative estimates.
`Wide Cyanobacteria Data`  %<>% mutate(
  # logs
  `log(Phosphorus) (1)` = log(`Phosphorus (1)`),
  `log(Microcystin) (1)` = log(`Microcystin (1)`),

  # lags
  `Time (0)` = `lag of`(`Time (1)`),
  `log(Phosphorus) (0)` = `lag of`(`log(Phosphorus) (1)`),
  `log(Microcystin) (0)` = `lag of`(`log(Microcystin) (1)`),

  # averages
  `log(Phosphorus)` =
    (`log(Phosphorus) (1)` + `log(Phosphorus) (0)`)/2,

  # standardizations
  `log(Phosphorus), standardized` =
    scale(`log(Phosphorus)`) %>% as.vector(),
  `log(Wind-Speed), standardized` =
    scale(`log(Wind-Speed)`) %>% as.vector(),
  `Temperature, standardized` =
    scale(Temperature) %>% as.vector(),
  `log(Microcystin), standardized (1)` =
    scale(`log(Microcystin) (1)`) %>% as.vector(),

  # squares
  `log(Phosphorus), standardized ^ 2` =
    `log(Phosphorus), standardized` ^ 2,
  `log(Wind-Speed), standardized ^ 2` =
    `log(Wind-Speed), standardized` ^ 2,
  `Temperature, standardized ^ 2` =
    `Temperature, standardized` ^ 2,

  # deltas
  `log(Microcystin), difference` =
    `log(Microcystin) (1)` - `log(Microcystin) (0)`,
  `Time, difference` =
    difftime(`Time (1)`, `Time (0)`, units = "days") %>%
    as.numeric(),
```

```
  # derivatives
  `log(Microcystin), growth rate` =
    `log(Microcystin), difference` / `Time, difference`)

# In order to simultaneously conduct operation on cyanobacteria populations,
# we need to convert to long form by clade,
# i.e. gather different clade columns into one column

`Long Cyanobacteria Data` =
  `Wide Cyanobacteria Data` %>%
  gather(
    Clade, `Cyanobacteria (1)`,
    Cyanophytes, Chlorophytes, Bacillariophytes, Cryptophytes, Euglenophytes) %>%
  # group by clade to avoid lagging into non-matching clades
  group_by(Clade)

# We will calculate a similar suite of lags, logs, averages, deltas, and derivatives,
# this time based on cyanobacteria populations.
# Note +1 within log expressions to avoid logging 0.
`Long Cyanobacteria Data` %<>% mutate(
  # log
  `log(Cyanobacteria) (1)` = log(`Cyanobacteria (1)` + 1),
  # lag
  `log(Cyanobacteria) (0)` = lag(`log(Cyanobacteria) (1)`),
  # average
  `log(Cyanobacteria)` =
    (`log(Cyanobacteria) (0)` + `log(Cyanobacteria) (1)`)/2,
  # delta
  `log(Cyanobacteria), difference` =
    `log(Cyanobacteria) (1)` - `log(Cyanobacteria) (0)`,
  # standardization
  `log(Cyanobacteria), standardized (1)` =
    scale(`log(Cyanobacteria) (1)`) %>% as.vector(),

  # derivatives
  `log(Cyanobacteria), growth rate` =
    `log(Cyanobacteria), difference` / `Time, difference`)

# Finally, filter the data to remove invalid time intervals
# That is, the interval between the two years of data.
`Wide Cyanobacteria Data` %<>%
  filter(`Time, difference` < 30 &
           !is.na(`Time, difference`))
`Long Cyanobacteria Data` %<>%
  filter(`Time, difference` < 30 &
```

```
           !is.na(`Time, difference`))

# Set up a basic frame for Monte Carlo simulation.
`Monte-Carlo ID` = 1:Replications
`Monte-Carlo Replications` = data_frame(`Monte-Carlo ID`)

# Build a frame of Geist ID's
`Geist ID Frame` =
  `Wide Cyanobacteria Data` %>%
  select(`Geist ID`)

# For each Monte Carlo simulation, randomly resample from the Geist ID's.
`Bootstrap ID's` =
  `Monte-Carlo Replications` %>%
  group_by(`Monte-Carlo ID`) %>%
  do(sample_frac(`Geist ID Frame`, replace = TRUE))

# Merge in Geist data in long form using Geist ID's.
`Long Cyanobacteria Bootstraps` =
  `Bootstrap ID's` %>%
  right_join(`Long Cyanobacteria Data`)

# We can simulate doubling time values randomly from a uniform distribution.
# Then convert to logistic growth rates.
`Long Cyanobacteria Bootstraps` %<>%
  group_by(`Monte-Carlo ID`, Clade) %>%
  mutate(
    `Doubling Time` = runif(1, min = 0.3, max = 1.2),
    `Rate Constant` = 2 * log(2) / `Doubling Time`)

# Calculate K using the equation above.
# Log K to prepare for log-log model below
`Long Cyanobacteria Bootstraps` %<>% mutate(
  `Carrying Capacity` =
    `Rate Constant` * exp(`log(Cyanobacteria) (0)`) /
    (`Rate Constant` - `log(Cyanobacteria), growth rate`),
  `Carrying Capacity` = ifelse(
    `Carrying Capacity` < 1 | is.na(`Carrying Capacity`),
    1,
    `Carrying Capacity`),
  `log(Carrying Capacity)` = log(`Carrying Capacity`))

# A function to convert regression results into a dataframe
# which contains model paremeters
```

```
`linear regression of` = function(Dataframe, Model) {
  # Run the model
  Results = lm(Model, Dataframe)
  # Select coefficients and convert to a dataframe
  `Coefficeints` =
    summary(Results)$coefficients[, "Estimate"] %>%
    t %>% as.data.frame
  # Also select out sigma, an estimate of the standard deviation of the error term
  `Coefficeints`$`Error Standard Deviation` =
    summary(Results)$sigma
  `Coefficeints` %<>% rename(`Intercept` = `(Intercept)`)
  # Remove backticks
  names(`Coefficeints`) %<>%
    stri_replace_all_fixed("`", "")
  # Add the word parameter to names
  names(`Coefficeints`) = paste(names(`Coefficeints`), "parameter")
  `Coefficeints`}

# This is our model for capacity (see above).
`log(Carrying Capacity) Model` =
  `log(Carrying Capacity)` ~
    `log(Wind-Speed), standardized` +
    `log(Wind-Speed), standardized ^ 2` +
    `Temperature, standardized` +
    `Temperature, standardized ^ 2` +
    `log(Phosphorus), standardized`

# Run the model for each bootstrap and each cyanobacteria clade.
`log(Carrying Capacity) Model Results` =
  `Long Cyanobacteria Bootstraps` %>%
  group_by(`Monte-Carlo ID`, `Clade`) %>%
  do(`linear regression of`(., `log(Carrying Capacity) Model`))

# First, collect the variables we are interested in and gather them into one column
`Wide Cyanobacteria Inputs` =
  `Wide Cyanobacteria Data` %>% select(
    `log(Wind-Speed), standardized`,
    `Temperature, standardized`,
    `log(Phosphorus), standardized`,
    `log(Microcystin), standardized (1)`)

`Long Cyanobacteria Inputs` =
  `Wide Cyanobacteria Inputs` %>%
  gather(Input, Value)
```

```r
# Next, plot normal quantile-quantile plots.
`Cyanobacteria Specification Graph` =
  ggplot(`Long Cyanobacteria Inputs`) +
  aes(sample = Value, color = Input) +
  stat_qq(geom = "line") +
  labs(
    title = "Normal QQ plot for wide-form variables",
    x = "Theoretical z-score",
    y = "Sample z-score")

# Note that we are going to create a normalized version of log Cyanobacteria,
# which is required for the QQ plot function.
`Cyanobacteria Clade Specification` =
  `Long Cyanobacteria Data` %>%
  group_by(Clade) %>%
  mutate(
    `log(Cyanobacteria), standardized` =
      scale(`log(Cyanobacteria)`) %>%
      as.vector)

`Cyanobacteria Clade Specification Plot` =
  ggplot(`Cyanobacteria Clade Specification`) +
  aes(sample = `log(Cyanobacteria), standardized`, color = Clade) +
  stat_qq(geom = "line")  +
  labs(
    title = "Normal QQ plot for clade population",
    x = "Theoretical z-score",
    y = "Sample z-score")

# Generate pairwise correlations between variables
# Note: covariance between clades is not an issue,
# because all were boot-strapped correspondingly.

Correlations =
  `Wide Cyanobacteria Inputs` %>%
  filter(complete.cases(.)) %>%
  cor


# After noting a particularly high correlation between wind and temperature,
# check the p-value to see if it is significant.
with(`Wide Cyanobacteria Data`,
     cor.test(`log(Wind-Speed), standardized`, `Temperature, standardized`))

# First, find the mean and standard deviation of our four predictor variables
```

```
# for each bootstrap.
`Predictor Distributions` =
  `Long Cyanobacteria Bootstraps` %>%
  ungroup %>%
  # Bump up the timeframe for the simulation in period 1-2.
  rename(`log(Cyanobacteria) (1), simulated` =
           `log(Cyanobacteria) (1)`,
         `log(Wind-Speed), standardized (1.5)` =
           `log(Wind-Speed), standardized`,
         `log(Phosphorus), standardized (1.5)` =
           `log(Phosphorus), standardized`,
         `Temperature, standardized (1.5)` =
           `Temperature, standardized`) %>%

  # Gather predictor variables into one column
  gather(Predictor, Value,
         `log(Wind-Speed), standardized (1.5)`,
         `Temperature, standardized (1.5)`,
         `log(Phosphorus), standardized (1.5)`,
         `log(Cyanobacteria) (1), simulated`) %>%
  # Find the mean and standard deviation for each bootstrap, clade, and predictor
  group_by(`Monte-Carlo ID`, Clade, Predictor) %>%
  summarize(
    Mean = mean(Value, na.rm = TRUE),
    `Standard Deviation` = sd(Value, na.rm = TRUE))

# Simulate the values for each boot-strap
# We need to separate population from the other variables.
# Population is special because random effects should not be constant between clades.
# Environmental effects, however, should be constant across clades within a simulation.
`Cyanobacteria Simulations` =
  `Predictor Distributions` %>%
  filter(Predictor == "log(Cyanobacteria)") %>%
  # for each Monte-Carlo simulation and each clade
  group_by(`Monte-Carlo ID`, Clade) %>%
  mutate(Simulation = rnorm(1, Mean, `Standard Deviation`))

`Other Simulations` =
  `Predictor Distributions` %>%
  filter(Predictor != "log(Cyanobacteria)") %>%
  group_by("Monte-Carlo ID") %>%
  mutate(Simulation = rnorm(1, Mean, `Standard Deviation`))

# Bind the two back together to finish processing
`Simulations` =
```

```
  bind_rows(
    `Cyanobacteria Simulations`,
    `Other Simulations`) %>%
  select(Predictor, Simulation, `Monte-Carlo ID`, Clade) %>%
  # Back to wide form, with each predictor in a seperate column
  spread(Predictor, Simulation)

# Read in STEPL
`STEPL Data` = read.csv("STEPL.csv", check.names = FALSE) %>% mutate(
  # and calculate proportions
  `Phosphorus Proportion After Reduction` =
    `Total Phosphorus`/max(`Total Phosphorus`))

# Recollect rate constant data for each simulation from the previous analysis.
`Rate Constant Data`=
  `Long Cyanobacteria Bootstraps` %>%
  ungroup %>%
  select(`Rate Constant`, `Monte-Carlo ID`) %>%
  group_by(`Monte-Carlo ID`) %>%
  summarize(
    `Rate Constant` = first(`Rate Constant`))

# Bring together all the data-sets we will need for predictions
# This includes model results, distributions, and rate constant data.
`Cyanobacteria Prediction Data` =
  `Simulations` %>%
  left_join(`log(Carrying Capacity) Model Results`) %>%
  left_join(`Rate Constant Data`) %>%
  merge(`STEPL Data`)

# Finally, scale down phosphorus values using STEPL data.
# Note, use of logarithms converts multiplication to addition.
`Cyanobacteria Prediction Data` %<>%
  mutate(`log(Phosphorus), standardized (1.5)` =
           `log(Phosphorus), standardized (1.5)` +
           log(`Phosphorus Proportion After Reduction`))

`Cyanobacteria Prediction Data` %<>%
  group_by(`Monte-Carlo ID`, `Clade`) %>%
  mutate(
  # First, run the model to predict K.
  # Note the use of a random error term.
  `log(Carrying Capacity), (1.5)` =

    `Intercept parameter` +
```

```
    `log(Wind-Speed), standardized parameter` *
      `log(Wind-Speed), standardized (1.5)` +

    `log(Wind-Speed), standardized ^ 2 parameter` *
      `log(Wind-Speed), standardized (1.5)`^2 +

    `Temperature, standardized parameter` *
      `Temperature, standardized (1.5)` +

    `Temperature, standardized ^ 2 parameter` *
      `Temperature, standardized (1.5)`^2  +

    `log(Phosphorus), standardized parameter` *
      `log(Phosphorus), standardized (1.5)` +

    rnorm(1) * `Error Standard Deviation parameter`,

  # Estimate the growth rate in the coming interval using logistic growth
  `log(Cyanobacteria), growth rate (1.5)` =
    `Rate Constant` *
    (1 -
      exp(`log(Cyanobacteria) (1), simulated`) /
      exp(`log(Carrying Capacity), (1.5)`)),

  # Use this growth rate to estimate population values at the end of next period.
  # Use intervals of 14 days.
  `log(Cyanobacteria) (2)` =
    `log(Cyanobacteria) (1), simulated` +
    14 * `log(Cyanobacteria), growth rate (1.5)`)

# Separate intial clade data back into different columns
`Revised Wide Cyanobacteria Data` =
  `Long Cyanobacteria Data` %>%
  ungroup %>%
  mutate(Clade = sprintf("log(%s) (0)", Clade)) %>%
  select(
    Clade,
    `log(Cyanobacteria) (0)`,
    `Geist ID`,
    `log(Microcystin), growth rate`,
    `log(Microcystin) (1)`) %>%
  # Each clade gets it's own column
  spread(Clade, `log(Cyanobacteria) (0)`)

# Use the original bootstrap ID shuffle to create bootstrapped wide data.
```

```
`Wide Cyanobacteria Bootstraps` =
  `Bootstrap ID's`  %>%
  left_join(`Revised Wide Cyanobacteria Data`)

# Our model for microcystin concentrations.
`Microcystin Model` =
  `log(Microcystin), growth rate` ~
    `log(Cyanophytes) (0)` +
    `log(Chlorophytes) (0)` +
    `log(Bacillariophytes) (0)` +
    `log(Cryptophytes) (0)` +
    `log(Euglenophytes) (0)`

# Run the model for each bootstrap.
`Microcystin Model Results` =
  `Wide Cyanobacteria Bootstraps` %>%
  group_by(`Monte-Carlo ID`) %>%
  do(`linear regression of`(., `Microcystin Model`))

`Microcystin Simulations` =
  `Wide Cyanobacteria Bootstraps` %>%
  group_by(`Monte-Carlo ID`) %>%
  # Find distribution parameters
  summarize(
    Mean = mean(`log(Microcystin) (1)`, na.rm = TRUE),
    `Standard Deviation` = sd(`log(Microcystin) (1)`, na.rm = TRUE)) %>%
  # Simulate randomly from a log normal distribution
  mutate(`log(Microcystin) simulated (2)`=
           rnorm(1, Mean, `Standard Deviation`))

# Separate prediction clade data back into different columns
`Cyanobacteria Predictions`  =
  `Cyanobacteria Prediction Data`  %>%
  ungroup %>%
  # Indicate that clade data  (will be) the log value after a second projected period
  mutate(Clade = sprintf("log(%s) (2)", Clade)) %>%
  select(`Monte-Carlo ID`,
         Clade,
         `Implementation Plan`,
         `log(Cyanobacteria) (2)`) %>%
  # Each clade gets its own column
  spread(Clade, `log(Cyanobacteria) (2)`)

# Merge distributions, model results, and predictions all together.
`Microcystin Predictions` =
```

```
  `Microcystin Simulations` %>%
  left_join(`Microcystin Model Results`) %>%
  left_join(`Cyanobacteria Predictions`) %>%
  group_by(`Monte-Carlo ID`) %>%
  mutate(
    #Estimate the derivative.
    `log(Microcystin), growth rate (2.5)` =
      `Intercept parameter` +
      `log(Cyanophytes) (0) parameter` *
        `log(Cyanophytes) (2)` +
      `log(Chlorophytes) (0) parameter` *
        `log(Chlorophytes) (2)` +
      `log(Cryptophytes) (0) parameter` *
        `log(Cryptophytes) (2)` +
      `log(Euglenophytes) (0) parameter` *
        `log(Euglenophytes) (2)` +
      `log(Bacillariophytes) (0) parameter` *
        `log(Bacillariophytes) (2)` +
      rnorm(1) * `Error Standard Deviation parameter`,
    # Use this growth rate to estimate concentration values at the end of next period.
    # Use intervals of 14 days.
    `log(Microcystin) (3)` =
      `log(Microcystin) simulated (2)` +
      14 * `log(Microcystin), growth rate (2.5)`)

`Just Microcystin Predictions` =
  `Microcystin Predictions` %>%
  ungroup %>%
  select(`Monte-Carlo ID`,
         `Implementation Plan`,
         `log(Microcystin) (3)`)

# A function to estimate log-normal distributions from confidence intervals.
`log Normal Parameters of` =
  function(confidence, `exp(lower limit)`, `exp(mean)`, `exp(upper limit)`) {
    # logs
    `lower limit` = log(`exp(lower limit)`)
    `mean` = log(`exp(mean)`)
    `upper limit` = log(`exp(upper limit)`)

    # calculate confidence
    `z-score` = qnorm(1 - (1 - `confidence`)/2)

    # fill in missing information using symmetry
    # if else statements only replace data if it is missing
```

```
    `lower limit` = ifelse(
      is.na(`lower limit`),
      2*mean - `upper limit`,
      `lower limit`)

    mean = ifelse(
      is.na(mean),
      (`lower limit` + `upper limit`) /2 ,
      mean)

    `upper limit` = ifelse(
      is.na(`upper limit`),
      2*mean - `lower limit`,
      `upper limit`)

    # back-calculate standard deviation
    `standard deviation` = (`upper limit` - `lower limit`) / 2 / `z-score`

    data_frame(`mean`, `standard deviation`)}

# Input exposure data

`Exposure Data` = read.csv("exposure.csv", check.names = FALSE) %>%
  mutate(
    # If per_person exposure is missing, calculate it from total exposure and
    # number of people.
    `Per Person Exposure Time` = ifelse(
      # if
      is.na(`Per Person Exposure Time`),
      # then
      `Total Exposure Time`/`Number Exposed`,
      # else
      `Per Person Exposure Time`)) %>%
  rowwise %>%
  mutate(
    #estimate distribution results
    `Exposure Parameters` = list(`log Normal Parameters of`(
      `Exposure Confidence`,
      `Exposure lower limit`,
      `Exposure mean`,
      `Exposure upper limit`)),

    `log(Exposure), mean` = `Exposure Parameters`[[1]],
    `log(Exposure), standard deviation` = `Exposure Parameters`[[2]])
```

```
# input endpoint organism data
`Endpoint Organism Data` =
  read.csv("endpoints.csv", check.names = FALSE) %>%
  rowwise %>% mutate(
    `Mass Parameters`  = list( `log Normal Parameters of`(
      `Mass confidence`,
      `Mass lower limit`,
      `Mass mean`,
      `Mass upper limit`)),

    `log(Mass), mean` = `Mass Parameters`[[1]],
    `log(Mass), standard deviation` = `Mass Parameters`[[2]])

# input model organism data
`Model Organism Data` =
  read.csv("models.csv", check.names = FALSE)

# Merge exposure data with predictions above, model organism data,
# and endpoint organism data.
# Note, this will replicate data for each exposure scenario
# and each species
`Dose Data` =
  `Just Microcystin Predictions` %>%
  merge(`Exposure Data`) %>%
  merge(`Endpoint Organism Data`) %>%
  left_join(`Model Organism Data`)

# Simulate mass because we need it for the exposure calculations
`Dose Data` %<>%
  group_by(Species,
           `Monte-Carlo ID`,
           `Exposure Scenario`,
           `Time Frame`) %>%
  mutate(
  # Randomly generate a bodyweight
  `log(Mass), simulated` =
    rnorm(1, `log(Mass), mean`, `log(Mass), standard deviation`))

#Separate drinking and non-drinking data
`Non-Drinking Data` = `Dose Data` %>%
  ungroup %>%
  filter(`Exposure Scenario` != "Drinking") %>%
  group_by(Species,
           `Monte-Carlo ID`,
           `Exposure Scenario`,
```

```
           `Time Frame`) %>%
  mutate(
    # Simulate exposure
    `log(Exposure), simulated` =
      rnorm(1,
            `log(Exposure), mean`,
            `log(Exposure), standard deviation`)) %>%
  ungroup

#Just for drinking data, calculate exposure from body weight (see above).
`Drinking Data` = `Dose Data` %>%
  filter(`Exposure Scenario` == "Drinking") %>%
  mutate(
  `log(Exposure), simulated` =
    `Zero Water Consumption` +
    `Water Consumption Elasticity` * `log(Mass), simulated`) %>%
  ungroup

`Dose Data Merge` =
  bind_rows(`Drinking Data`, `Non-Drinking Data`) %>%
  mutate(
    # Add/subtract logs to multiple/divide and calculate dose.
    `log(Dose), simulated` =
      `log(Microcystin) (3)` +
      `log(Exposure), simulated` +
      log(`Per Person Exposure Time`) -
      `log(Mass), simulated` -
      log(1000))

# Convert doses to model doses to match toxicity data below.
`Dose Data Merge` %<>% mutate(
  `log(Dose), simulated` =
    `log(Dose), simulated` +
    0.25 * `log(Mass), simulated` -
    0.25 * log(`Average Model Mass`))

# Read in data
`Toxicology Data` = read.csv("tox.csv", check.names = FALSE) %>%
  # Remove congestion; it won't converge
  filter(Response != "Congestion")

# Function to convert tox data to long form, i.e.,
# a 0 or 1 for each animal
`Convert Toxicology Data to Long Form` = function(Dataframe) {
  Zeros = rep(0, Dataframe$`Total Number` - Dataframe$`Number With Response`)
```

```
  Ones = rep(1, Dataframe$`Number With Response`)
  data_frame(Exhibited = c(Zeros, Ones))}

# Apply the function
`Long Toxicology Data` =
  `Toxicology Data` %>%
  group_by(Response, `Time Frame`, `Model Species`, Dose) %>%
  do(`Convert Toxicology Data to Long Form`(.)) %>%
  group_by(Response, `Time Frame`, `Model Species`)

# Bootstrap the data
# For each Monte Carlo simulation, randomly resample from the Geist ID's.
`Toxicology Data Bootstraps` =
  `Monte-Carlo Replications` %>%
  group_by(`Monte-Carlo ID`) %>%
  do(sample_frac(`Long Toxicology Data`, replace = TRUE))

# Logit function
`logistic regression of` = function(Dataframe) {
  # Run logit
  Results = glm(
    Exhibited ~ log(Dose + 1),
    Dataframe,
    family = "binomial")

  # Format results
  Results$coefficients %>%
    t %>% as.data.frame %>%
    rename(
      `Intercept parameter` = `(Intercept)`,
      `log(Dose) parameter` = `log(Dose + 1)`)}

# Run logits for each simulation, response, time frame, and model species
`Toxicology Results` =
  `Toxicology Data Bootstraps` %>%
  group_by(`Monte-Carlo ID`, Response, `Time Frame`, `Model Species`) %>%
  do(`logistic regression of`(.))

# Convert endpoint doses to model doses
# This allows us to plug in endpoint doses

# Merge our toxicology data into the final results
`Final Odds` =
  `Dose Data Merge` %>%
  left_join(`Toxicology Results`)
```

```
# Build odds ratios
`Final Odds` %<>% mutate(
  #Convert dose from animals to humans
  #Small fix to account for +1 adjustment
  #`log(Dose), simulated` = log(exp(`log(Dose), simulated`) + 1),
  #Calculate log odds
  `log(Odds Ratio)` = `Intercept parameter` +
    `log(Dose) parameter` * `log(Dose), simulated`)

# Make a sample graph for boating exposure

`Response Filtered Data` = `Final Odds` %>%
  filter(`Implementation Plan` == "No BMP's" &
           `Exposure Scenario` == "Boating" &
           Response == "Death") %>%
  ungroup %>%
  # Get rid of tails, 2.5% on each side.
  mutate(
    `Lower Bound` = quantile(`log(Odds Ratio)`, 0.015, na.rm = TRUE),
    `Upper Bound` = quantile(`log(Odds Ratio)`, 0.975, na.rm = TRUE)) %>%
  filter(`Lower Bound` < `log(Odds Ratio)` & `log(Odds Ratio)` < `Upper Bound`)

# Graph a smoothed density plot
`Response Graph` =
  ggplot(`Response Filtered Data`) +
  aes(x = `log(Odds Ratio)`) +
  geom_density(trim = FALSE) +
  labs(x = sprintf("Log Odds Ratio for %s after %s",
                   "Death", "Boating"))

# Calculate a table of casualties
`Casualty Table` =
  `Final Odds` %>%
  ungroup %>%
  # We need to know the number exposed for this calculation
  # We don't have this for the ecological stuff
  filter(!is.na(`Number Exposed`)) %>%
  select(
    Species,
    `Monte-Carlo ID`,
    `Implementation Plan`,
    `Exposure Scenario`,
    `Time Frame`,
    Response,
    `Number Exposed`,
```

```
    `log(Odds Ratio)`) %>%
  group_by(
    Species,
    `Exposure Scenario`,
    `Time Frame`,
    `Implementation Plan`,
    `Monte-Carlo ID`,
    Response) %>%
  # For each scenario defined by the groups above,
  # Randomly assign each individual a stamina to determine whether their ability
  # To withstand the change of a negative health outcome
  mutate(
    `Stamina` = runif(1, 0, 1),
    `Stamina Log Odds Ratio` = log(`Stamina`/(1-`Stamina`))) %>%
  # Regroup to summarize over Monte-Carlo replications
  group_by(
    Species,
    `Exposure Scenario`,
    `Time Frame`,
    `Implementation Plan`,
    Response) %>%
  # Calculate the expected value of the proportion of people who
  # will express a negative health outcome
  summarize(
    `Number Exposed` = first(`Number Exposed`),
    `Expected Number with Symptoms` =
      sum(`log(Odds Ratio)` > `Stamina Log Odds Ratio`)/n() * `Number Exposed`)

write.csv(`Casualty Table`, "Expected Value Table.csv")

# Calculate the difference in log(Odds Ratio) for the two BMP plans
`Wide Odds` =
  `Final Odds`  %>%
  select(
    Species,
    `Exposure Scenario`,
    `Time Frame`,
    `Monte-Carlo ID`,
    `Implementation Plan`,
    Response,
    `log(Odds Ratio)`) %>%
  # Put the data for each implementation plan in a separate column
  spread(`Implementation Plan`, `log(Odds Ratio)`) %>%
  # Now gather the two treatment plans into one column
  gather(`Implementation Plan`, `Post-Policy log(Odds Ratio)`,
```

```
          `Recommended BMP's`, `Full BMP's`) %>%
  # Compare them against the control
  mutate(`Percent Decrease in Odds Ratio` =
           (1 - exp(`No BMP's` - `Post-Policy log(Odds Ratio)`)) * 100) %>%
  group_by(Species,
           `Exposure Scenario`,
           `Time Frame`,
           `Implementation Plan`,
           Response) %>%
  # Calculate confidence intervals
  summarize(
    `2.5% Percent Decrease in Odds Ratio` =
      quantile(`Percent Decrease in Odds Ratio`, 0.25, na.rm = TRUE),
    `97.5% Percent Decrease in Odds Ratio` =
      quantile(`Percent Decrease in Odds Ratio`, 0.975, na.rm = TRUE))

write.csv(`Wide Odds`, "Program Evaluation.csv")

# Calculate confidence intervals for the log of the odds ratios for individuals.
`Interval Table` =
  `Final Odds`  %>%
  group_by(
    Species,
    `Exposure Scenario`,
    `Time Frame`,
    `Implementation Plan`,
    Response) %>%
  # Find confidence intervals
  summarize(
    `2.5% log(Odds Ratio)` = quantile(`log(Odds Ratio)`, 0.25, na.rm = TRUE),
    `97.5% log(Odds Ratio)` = quantile(`log(Odds Ratio)`, 0.975, na.rm = TRUE))

write.csv(`Interval Table`, "Interval Table.csv")
```