Estimating Uncertainty in Fishing Effort Estimates Using Bootstrapping with a Two-Stage Model

A Senior Project
Presented to
The Faculty of the Statistics Department
California Polytechnic State University, San Luis Obispo

In Partial Fulfillment
Of the Requirements for the Degree
Bachelor of Science, Statistics

By:
Samantha Dellinger
June 2014

Table of Contents

I. Introduction

A. Background

In 2007, in hopes of protecting the two species of fish rockfish and lingcod, the government implemented marine protected areas (or mpa's), in the South Central Coast. In these protected areas shown in Figure 1,[1] the capabilities of the captains that lead the party boat fishing trips were limited, which raised the question of how the fishing has changed since the marine protected areas were put in place. It is a common hypothesis that the fisherman will fish on the edge of the specified marine protected areas, in hopes that fishing extremely close to the protected areas will lead to them catching bigger and better fish.

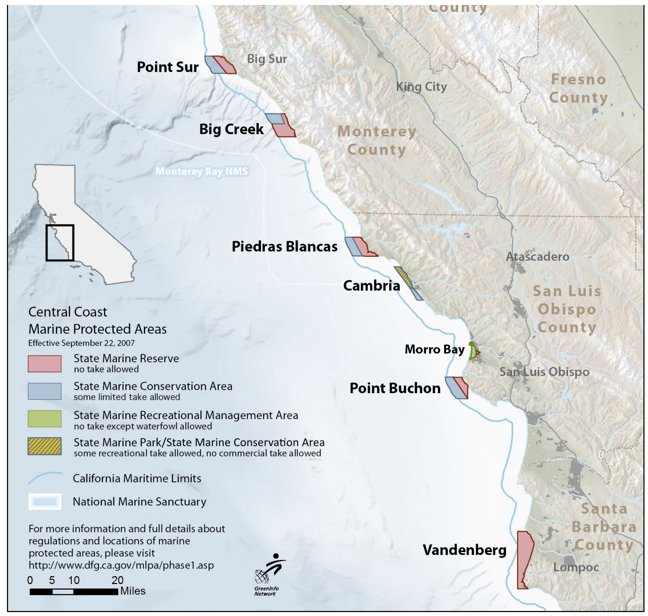


Figure 1: Map of the Central Coast

Party boat fishing consists of a captain, who owns a commercial passenger fishing vessel, being paid to take others on an agreed upon number of trips, or until all of the people on the vessel have reached their bag limit (the legal maximum number fish an individual can catch). For the South Central Coast, these fishing trips come out of Morro Bay and Port San Luis, and are primarily booked through Virg's Sport Fishing, Patriot Sport Fishing, and Central Coast Sport Fishing.

B. The Data

When Dr. Andrew Schaffner and I met with Biology grad student Morgan Ivens-Duran for the first time, she came to us with a data set that included information pertaining to party boat fishing on the South Central Coast of California that covered eleven past years. This data collection began in 2003, when Morgan's lab, the Center for Coastal Marine Sciences, became interested in where fisherman on the South Central Coast were fishing, and how the rockfish and lingcod cohabitate. As natural in a sample, this data set included

[1] California Marine Protected Areas, *Central Coast MPA Region*, http://www.californiampas.org/pages/regions/centralcoast.html (January 2014).

information on a small proportion of the boat trips that were made each year, only about 6 to 22 percent. For each trip that was sampled, we have information on three variables: the latitude and longitude of every drop that was made, and the effort that was made at each drop (cumulative time per drop that all fishing lines are in the water).

In 2007, the marine protected areas were put in place, and as a result, the data is split into pre-mpa data from 2003-2006, and post-mpa data from 2008-2012. For the purpose of our analyses, we will leave out the year 2007 because it was the year that the marine protected areas were put in place, and thus has mixed pre and most mpa data. Below, Figure 2 shows a bar graph of the total number of trips that were made per year, showing the divide between pre-mpa years (green) and post-mpa year (blue) with 2007 as a buffer year in gray.

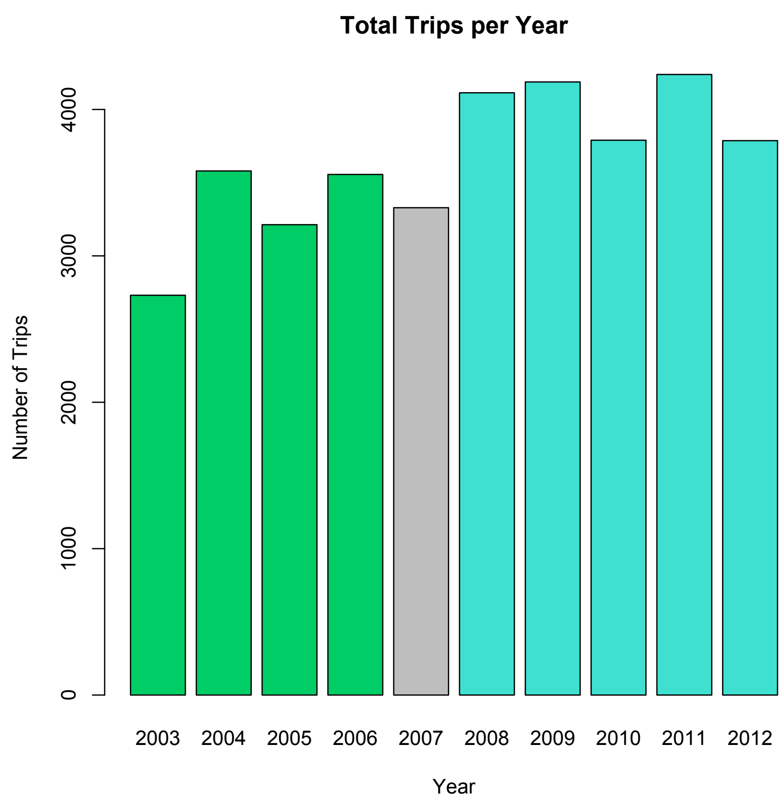


Figure 2: Bar Graph of Total Trips Made per Year (Appendix lines 285-295)

Pre and Post MPA data can be compared to investigate how the spatial pattern and intensity of fishing effort had changed in the South Central Coast region over the 10 year time period, and specifically how it has changed as a result of the implementation of the marine protected areas.

C. Analytics

In order to assist Morgan in exploring her research question, we created estimates of the fishing effort along the Central Coast for each year, allowing her to get a clear picture of where the most effort was located. Furthermore, we provide estimates of the uncertainty in our effort estimates along the Central Coast. To do this we applied several estimation techniques to the data including Kernel Density Estimation and Splines to estimate effort, and Bootstrapping to estimate the uncertainty in our estimates.

The Kernel Density estimation will allow us to estimate the spatial distribution (density) of drops to gain a better understanding of where the fisherman are fishing and how often. In addition to where and how often these fisherman were fishing, we needed to also include the amount of effort that was being put forth each time fishing lines were dropped at each location. We estimated the effort using two-dimensional splines. Finally, to estimate the variability (uncertainty) of these estimates we used bootstrapping. This process was repeated separately for each of the 10 years of pre-mpa and post-mpa data to create a clear picture could be drawn about how the fishing location effort changes from year to year.

III. Tools for Analysis

A. Kernel Density Estimation to Estimate Drop Densities

Kernel Density Estimation is a nonparametric way to estimate the probability density of a random variable. In the simplest terms, Kernel Density Estimation is a density smoother calculated using a weighted moving average. This concept applies really well to a simple histogram: for a set value of x, we compute the (scaled by bandwidth) proportion of the data that falls within a specific neighborhood (determined by the bandwidth) of x. To illustrate this idea, we can first look at the definition of the pdf, $f(x)$, of a random variable X,

$$P(x-b<X<x+b)=\int_{x-b}^{x+b} f(t)dt \approx 2b\, f(x),$$

and thus,

$$f(x) \approx \frac{1}{2b} P(x-b<X<x+b)$$

where *b* represents the bandwidth parameter[2]. This probability can also be estimated by a frequency in the sample, so

$$\hat{f}(x) = \frac{1}{2b} \frac{\text{number of observations in } (x-b, x+b)}{n}.$$ [3]

The selection of the bandwidth, *b*, will have a significant effect on the kernel density estimations; choosing a *b* too large will result in an overly smooth and non descriptive curve, whereas choosing a *b* that is too small will result in a highly irregular and overly specific smoothing curve. The following histograms in Figure 3 are of a sample of 20 longitude values modeled with different values of *b*, and a possible kernel density estimate curve using the bandwidth *b*=0.1.

---

[2] Walter Zucchini, *Part I: Kernel Density Estimation,* Applied Smoothing Techniques, http://isc.temple.edu/economics/Econ616/Kernel/ast_part1.pdf

[3] Walter Zucchini, *Part 1: Kernel Density Estimation*, http://isc.temple.edu/economics/Econ616/Kernel/ast_part1.pdf

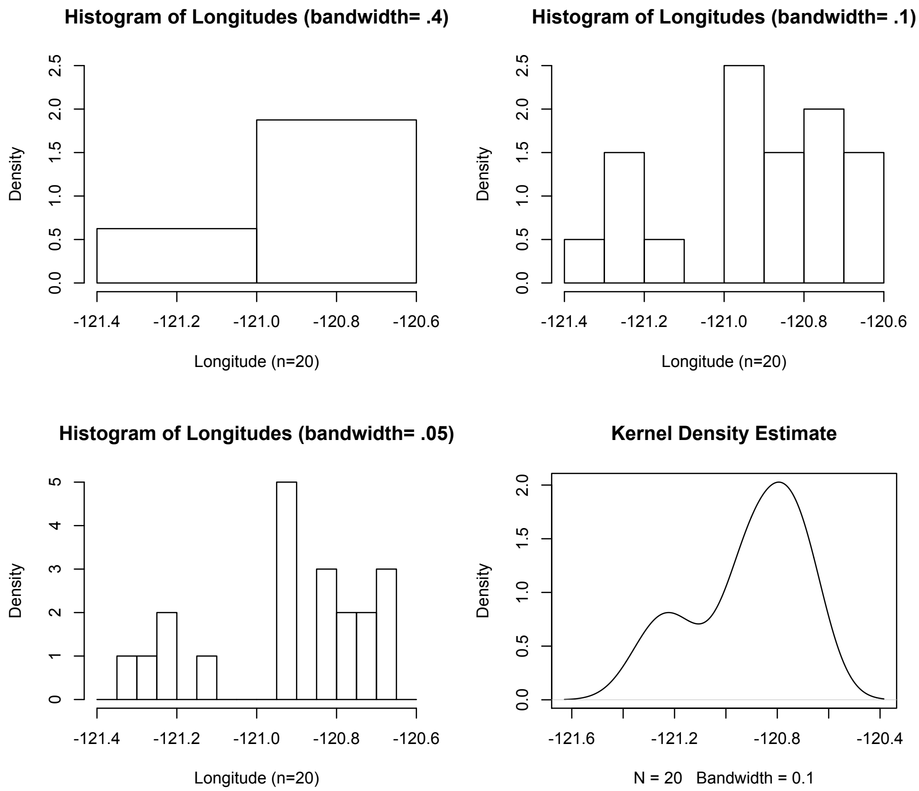


Figure 3: Sample of Longitudes plotted with varying bandwidths (Appendix lines 38-64)

You can visualize the kernel density estimate by smoothing a histogram with bins of width 2b. Imagine that you have a rectangle, with height $\frac{1}{2b}$ and width $2b$, that is placed over each point in the sample on the x-axis. The estimate of $f(x)$ at a given point is $\frac{1}{n}$ times the sum of the heights of all of the rectangles that include that point. This method would be using the rectangular "weighting" function, in which all of points in the rectangle are given an equal weight when calculating the estimate of $f(x)$ at that given point. Figure 4 shows an example of the effect that changing *b* (denoted bw in the image) has when using a rectangular kernel estimate.

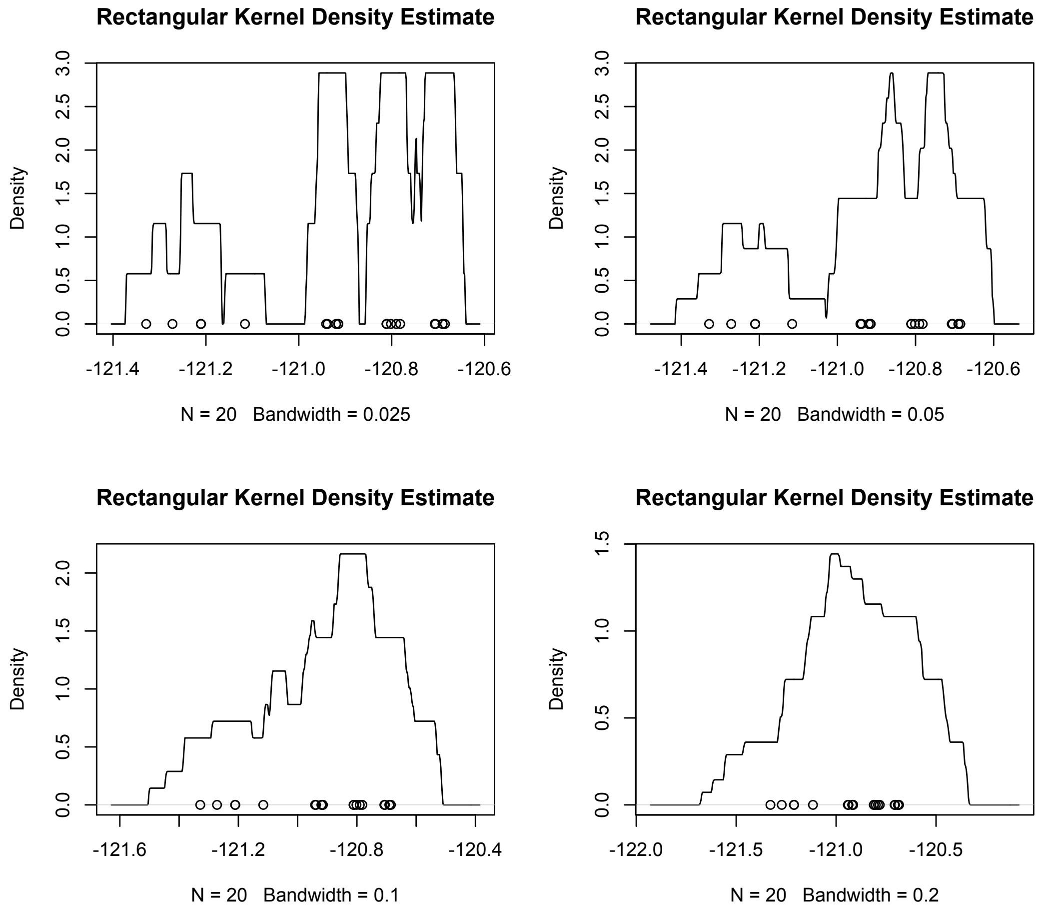


Figure 4: Rectangular Kernel Density Estimates of Sample of Longitudes with varying bandwidths (Appendix lines 69-117)

This series of images (Figure 4) shows the importance of choosing a value for *b*, which controls the degree of smoothing that is appropriate for the data. You can see that a value of *b*=0.025 results in very scattered peaks that do not do much smoothing of the data at all, where as a value of *b*=0.2 creates more smoothing in the data, but is slightly too large due to the flat and non-descript nature of the smoothing curve.

In the context of estimating the density of fishing locations (lat/long), we chose an appropriate bandwidth (b=.0155) using a cluster analysis of the latitudes and longitudes that were fished to identify clusters that would represent fishing spots. The median size of the fishing spots was used to choose the bandwidth.

As important as it is to choose an appropriate bandwidth for your data, it is also important to consider the type of kernel that you want to use. In the example above the type of kernel is rectangular, while other common types of kernels include triangular, Epanechnikov, and Gaussian. The important idea to note about these kernels is that they are all functions that weight the data and that follow the form,

$$w(t, b) = \frac{1}{h} K\left(\frac{t}{b}\right)$$

where $K\left(\frac{t}{b}\right)$ is a function of the kernel[4]; the kernel density will determine the shape of your smoothing function. For the purposes of our analyses, we applied a Gaussian kernel, which will put more weight on the points in the middle of our kernels and less weight on the points in the tails of the kernel, and will resemble a normal curve.

Because the weights are applied to the points that are encapsulated in each kernel, the effect of a bandwidth that is too small is amplified and the curve will result in having too many peaks. On the other end of the spectrum, a bandwidth that is too large will result in a very shallow and normal-looking curve that does not illustrate any patterns in the data. The data that was modeled using rectangular kernels above is now illustrated using Gaussian kernels, in Figure 5.

---

[4] Walter Zucchini, *Part 1: Kernel Density Estimation*, http://isc.temple.edu/economics/Econ616/Kernel/ast_part1.pdf

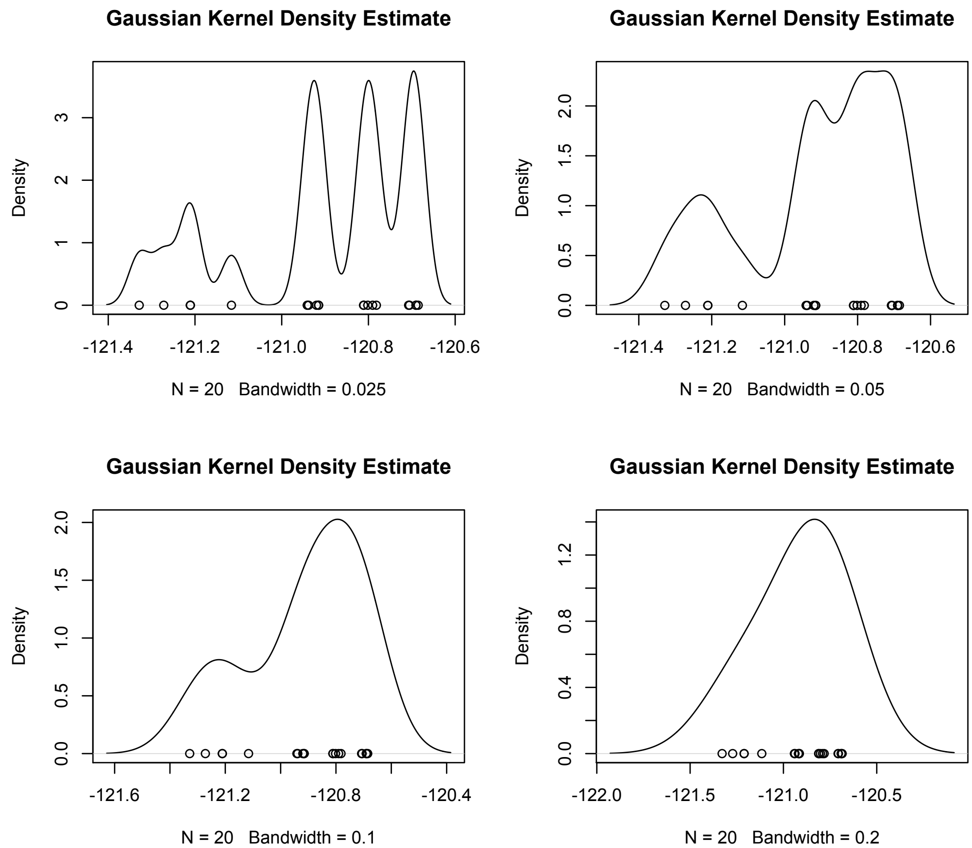


Figure 5: Gaussian Kernel Density Estimation of Sample of Longitudes with varying bandwidths (Appendix lines 122-167)

For the purposes of our data, we use kernel density estimation to obtain a density estimate of the drop locations (latitude and longitude), so we need to generalize the one dimensional kernel density estimation to be multidimensional.

Consider the form of the one-dimensional estimation, where again *b* is the bandwidth around x_0, and *K* is the kernel that controls the weight given to the point x_i based on its proximity to x_0:

$$\hat{f}\,(x_0) = \frac{1}{nb}\sum_i K\left(\frac{x_i - x_0}{b}\right).$$ [5]

This form is the estimation can be generalized to the multidimensional situation, in which different bandwidths are allowed for each direction, where

[5] Walter Zucchini, *Part 1: Kernel Density Estimation*, http://isc.temple.edu/economics/Econ616/Kernel/ast_part1.pdf

$$\widehat{f}(x) = \frac{1}{n}\sum_i \prod_{j=1}^{p} \frac{1}{b_j} K\left(\frac{x_{ij}-x_{0j}}{b_j}\right).$$ [6]

We used built-in functions in the software package R to obtain the kernel density estimates of the drop densities. The R code used for these drop location density estimates is located on pages 33 and 34 of the Appendix, from line 461 to line 473.

B. Generalized Additive Model to Smooth Effort Estimations

In order to predict effort values at latitude and longitude values that were not sampled during data collection, we will be using the generalized additive model (gam) function in R. Within the gam function we will be using splines as our smoothing functions to better predict our values for effort. In order to better explain the Generalized Additive Models, or GAM, we will first look at the General Linear Model and the Generalized Linear Model. To begin, consider the task of exploring the association between effort, and latitude and longitude values.

The general linear model, or least squares regression model refers to a situation in which we have a response variable (e.g. effort), that we believe to be some function of other variables (e.g. latitude and longitude). In this standard model, our explanatory variable is assumed to be normally distributed with mean μ and variance σ^2, where the X's are our predictor variables. These predictor variables are scaled by some coefficient β_i (or b_i in the context of a sample of data) and are summed, giving us the linear predictor that provides the estimated fitted y value according to the given X values. Symbolically, for a sample of data, this relationship looks like the following:

$$y = b_0 + b_1X_1 + b_2X_2.$$

This linear regression model often is too simplified and limited to capture what is really going on with the data, which is why we will next look at a more extended version, which is a generalized linear model.

In exploring this association with the general linear model we are assuming that a function of effort is some linear combination of latitude and longitude. In the generalized linear model, the link function of your variable which relates your predictor variables to a function of your explanatory variable(s), is directly related to the linear combination of your predictors. This link function can also be expressed as the estimated fitted values of your variable, which in the

---

[6] Patrick Breheny, *Kernel density estimation*, Slide 22, http://web.as.uky.edu/statistics/users/pbreheny/621/F12/notes/10-18.pdf

context of our example would be the estimated fitted values of effort. The linear combination of the predictor variables refers to, in our case, the values of latitude and longitude that are scaled by some coefficient b_i that is determined by software when the regression is run. This linear relationship can be expressed generally as,

$$g(\mu) = b_0 + b_1X_1 + b_2X_2.$$

One of the biggest differences between the general linear model and the generalized linear model is that distributions other than Gaussian can be applied as the link functions. Thus, the important idea to note here is that the generalized linear model is just an expansion of the standard general linear model that most are familiar with.

Taking the generalized linear model one step further results in the generalized additive model, which is the model that we are using in this analysis. The generalized additive model uses smooth functions of the predictor variables, which can take any number of forms. This model symbolically takes the form of,

$$g(\mu) = b_0 + f(x_1, x_2).$$

The incorporation of the link functions of the linear predictor variables is the key difference between the generalized linear model and the generalized additive model. The addition of the smoothing aspect of this equation allows for more accurate predictions of our effort values, based on a predictive function of latitude and of longitude. We will be using a spline smoother within the GAM function in R, to smooth and predict effort values based on the latitude and longitude.

The idea behind the GAM smoother is extremely similar to that of the kernel density estimators; for every x value, x_0, we choose a neighborhood around it and fit some type of model, for example a linear regression on the data points captured in that neighborhood. Using the fitted model for the specified neighborhood, you end up with a fitted value corresponding to that specific x_0, and if we repeat this process for every x_0 in our sample we would end up with fitted values based on a rolling neighborhood that is relative to each x value.[7] Instead of fitting a linear regression, we can fit a polynomial regression or some other type of model that gives weight to the x values in a given neighborhood based off of their relative distances from x_0. In the context of our analysis, the function we will use to weight our values is a spline smoother. Within these spline smoothers there are two categories, natural splines and b-splines; R uses b-splines for its estimations, so that is the function that will be the basis of our resulting analyses.

---

[7] Michael Clark, *Generalized Additive* Models, Center for Social Research University of Notre Dame, Page 7, http://www3.nd.edu/~mclark19/learn/GAMS.pdf

Splines are more complex smoothing functions because a spline curve is actually a piecewise polynomial curve that joins together two or more curves, or “basis functions”, at locations called “knots”. A spline is defined as being a piecewise m-1 degree polynomial that is continuous up to its first m-2 derivatives; the continuity requirements allows for the curve to be as smooth as possible. On the following page, Figure 6 shows an example of two curves joined at a knot at x=10. This example is for illustrative purposes only, because this piecewise curve is not continuously differentiable, and this cannot be a true spline. More flexible curves can be obtained by increasing the degree of the spline and/or by increasing the number of knots.[8]

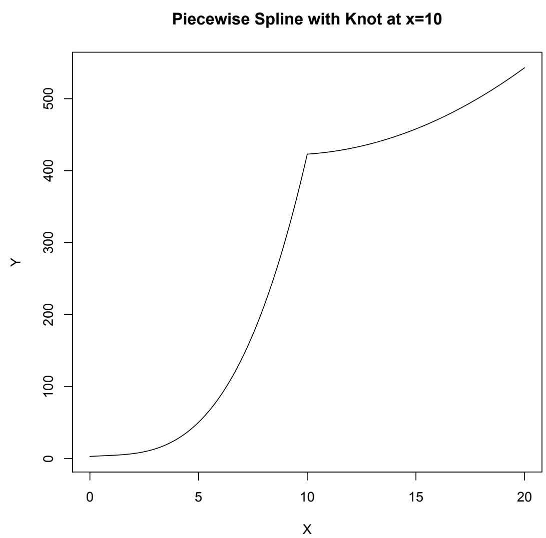


Figure 6: Piecewise Curve with a “Knot” (Appendix lines 178-188)

However, as with the kernel density estimation, there are tradeoffs for increasing or decreasing the number of knots used: having too few knots results in the functions being too restrictive and not fitting the data well while having too many knots leads to the risk of over fitting your data.

As with any estimation, it is important to also consider the error of these estimations, and is always helpful to construct confidence intervals illustrating the accuracy of your predictions. As stated previously, we will be working with the highly complicated but more stable and efficient spline in our R work and analyses, including the example the follows.

To illustrate the workings of spline smoothing functions, we will look at a simplified example of what we are doing, by looking at predicting effort from just a single predictor, longitude. The equation that goes alone with the single predictor equation is

$$\hat{y} = f(x),$$

---

[8] Patrick Breheny, *Kernel density estimation*, http://web.as.uky.edu/statistics/users/pbreheny/621/F12/notes/10-18.pdf

where f is a smooth function, specifically a smooth function of longitude for the following example.

The following images and analyses were based on the same sample of 20 longitudes, and their corresponding 20 efforts values. Figure 7 shows a scatterplot of the sample of twenty longitudes and their corresponding efforts that are being used for this example.

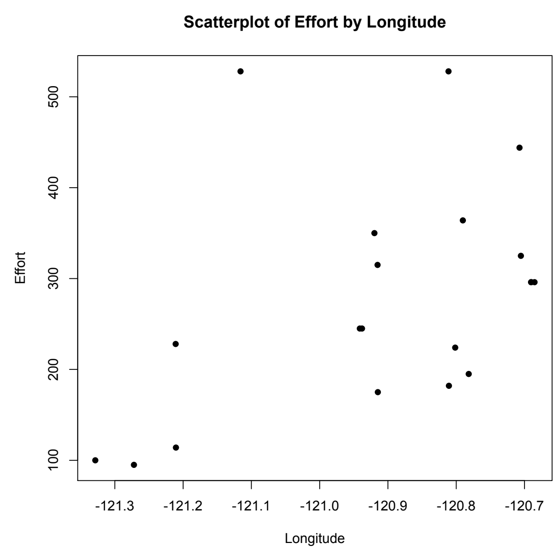


Figure 7: Scatterplot of Sample of Longitudes with Corresponding Effort Values (Appendix lines 192-211)

In Figure 8, the 20 sample longitude points are plotted against their relative predicted efforts (calculated with the spline function in R), as well as the spline curve that predicts the effort at all points between -121.4 and -120.6. Because these are predicted, or estimated, effort values it is important to show what the variability is around your estimations. The variability in the estimations is represented in this plot as the dotted lines that fall on either side of the prediction curve. These lines are the 95% confidence interval lines for the predictions, and you can see that the confidence lines show the most variability (are the farthest away from the line) in the places where there are little or no recorded effort values to base the predictions off of. Good illustrations of this occur at longitude values of about -121.2 and also around -120.6. On this graph there are also tick marks along the longitude axis that represent where the sample data points fall, and we can again look at the space where there are little or no tick marks and see that at those values the variability around the estimates is much higher.

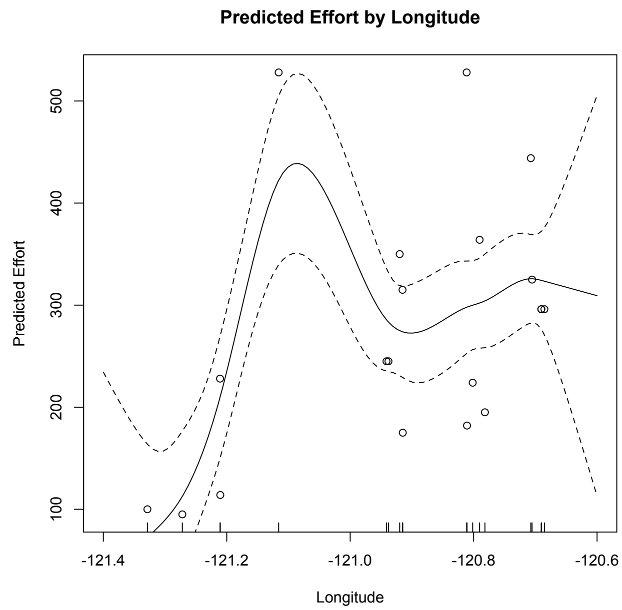


Figure 8: Spline Curve to Predict Effort from Longitude with 95% Confidence Bands (Appendix lines 216-225)

For our actual analyses, we will be carrying out the same basic process, however we will be predicting effort using an interactive spline model that includes a spline smoother that will be incorporating both longitude and latitude values, and the interaction between the two variables. This model is represented by,

$$\hat{y} = f(x_1, x_2),$$

where x_1 is the latitude and x_2 is the longitude. This model will allow us to predict where and how often effort is being exerted while fishing for the Central Coast of California, both where we do and do not have existing data.

As previously mentioned, it is extremely important to be able to include how much error is associated with your predictions. In order to predict this error, we will use a two-stage model with bootstrapping.

C. Two-Stage Bootstrapping to Estimate Fishing Effort and Variability of Estimates

The purpose of using bootstrapping is primarily to estimate the variability or uncertainty in the effort predictions. The bootstrap process is shown in Figure 9 and is described as follows. First, start with the original sample of data, of size

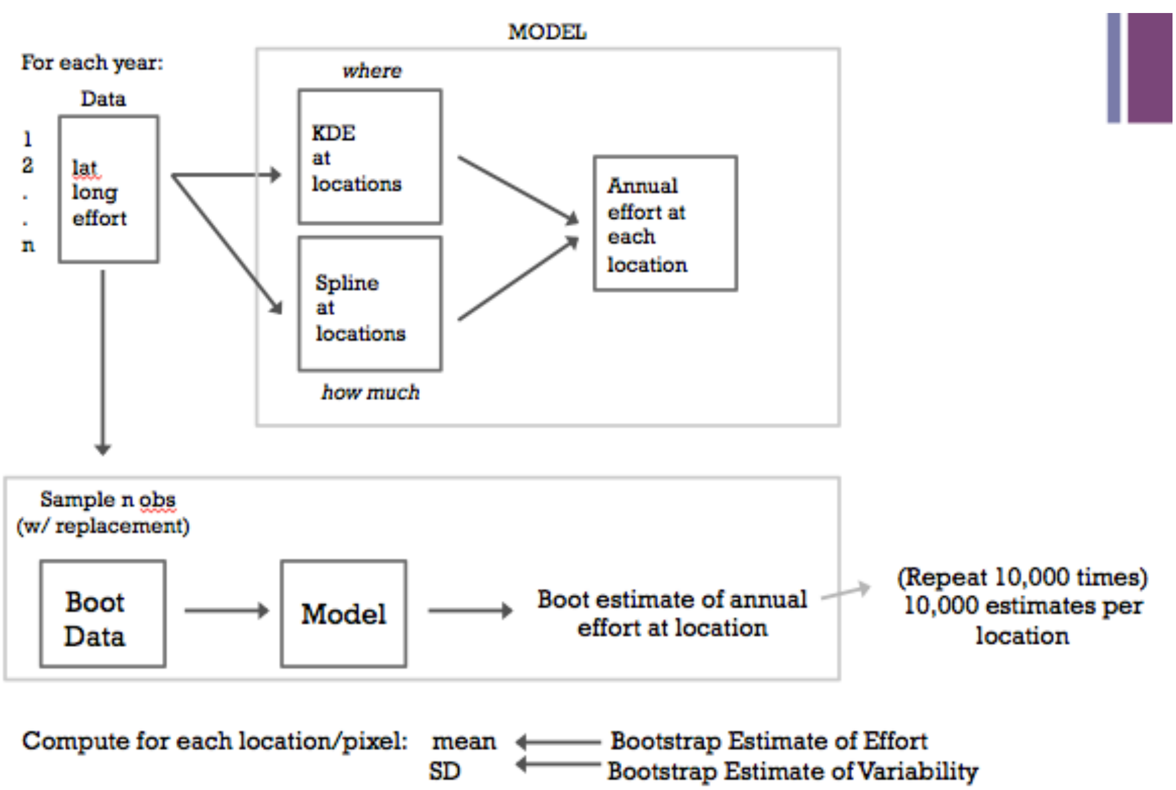
n. Then, sample n observations from the original data, with replacement, and call this the “boot data”. The original sample size and thus the size of the boot data will change depending on the year that the analysis is being performed on. This boot data is now fit using the two-stage model as previously explained- KDE and splines. The KDE allows us to determine where the fisherman are fishing, the spline allows us to determine how often the fisherman are fishing in certain location, and together they allow us to determine the bootstrap estimate of annual effort at each location on our map. This annual estimate is calculated by

$$total\ \#\ drops * prob\ of\ drop\ (at\ each\ location) *\ effort\ (at\ each\ location).$$

In this equation, the probability of a drop at each location, or pixel, is determined by the KDE and the effort at each location/pixel, is determines by the spline.

We will repeat this process 5,000 times, which will result in 5,000 estimates for annual effort at each location. To be able to create a single graph depicting the bootstrapped effort estimates and variability estimates, the final step is to take these 5,000 estimates and take the mean, which will result in the bootstrap estimate of effort, and take the standard deviation, which will result in the bootstrap estimate of variability.

Figure 9: Map of Bootstrap Process with a Two-Stage Model

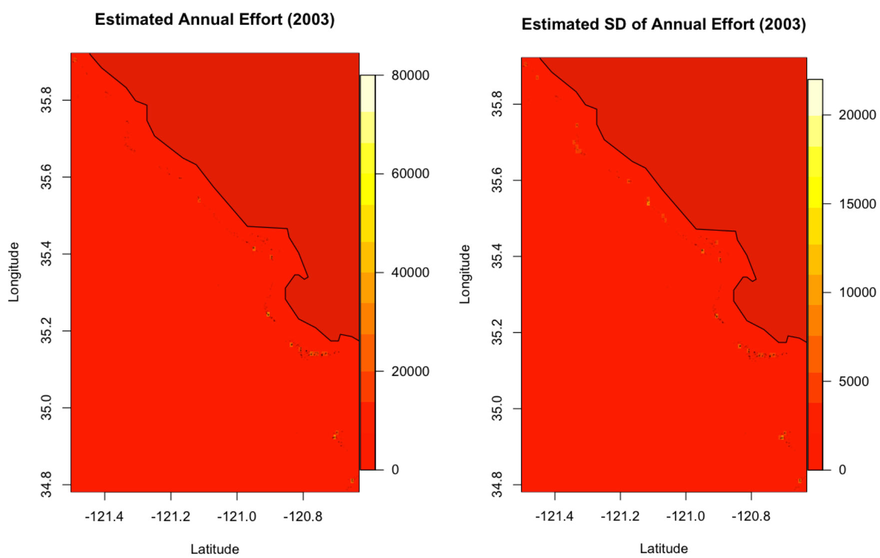


IV. Results

A. Estimation of Effort (KDE and Splines)

Using the bootstrapping as well as a combination of the kernel density estimation to estimate where fisherman were fishing, and splines to estimate how much effort was being put in at each location, the resulting images give a good picture of the fishing patterns along the central coast, from year to year. Figure 10 shows an example of the images that will be looked at in the following sections; these two images show the estimated effort and estimated variability in effort for the year 2003.

Figure 10: 2003 Estimated Annual Effort and Estimated Variability



For all 9 years of data, an image of the estimated annual effort has been created with the estimated effort at each pixel being measured on a heat scale from 0 to 80,000. In Figure 11 below, the estimated annual effort for 2003 for the whole Central Coast is placed next to a zoomed-in portion of the coast. This smaller portion of the Central Coast allows us to better see how these estimation are mapped and how the effort in this section is distributed; the enhanced image is also on a smaller heat scale (0 to 30,000) to better see how much effort is estimated at each pixel.

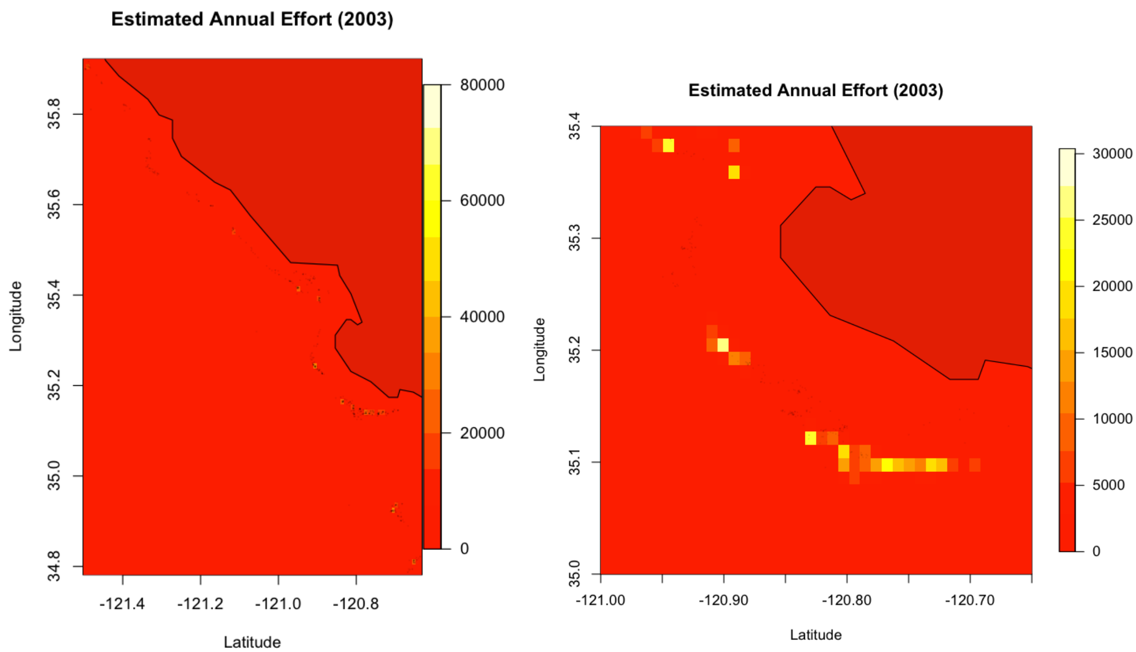


Figure 11: 2003 Estimated Annual Effort with Enhanced Portion to Illustrate Pixel-Wise Estimation of Effort (Appendix lines 566-576, 596-604)

To investigate whether or not the fishing patterns have changed since the mpa's were put in to place, we can compare the images in Figure 12, which depict the fishing patterns for the pre-mpa years, to the images in Figure 13, which depict the fishing patterns for the post-mpa years.

Figure 12: Pre MPA (2003-2006) Estimated Annual Effort
(Appendix lines 568-576 repeated for each year)

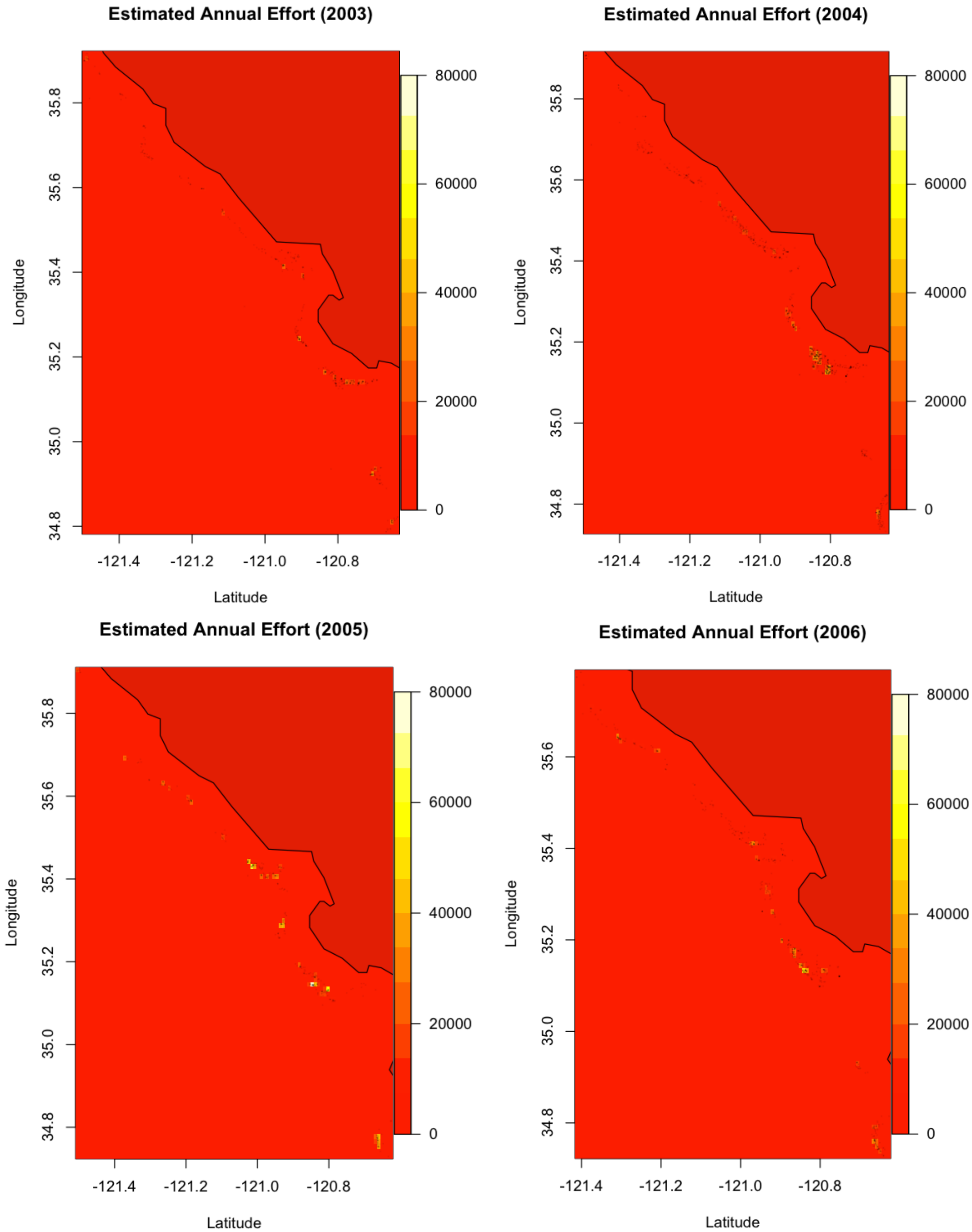
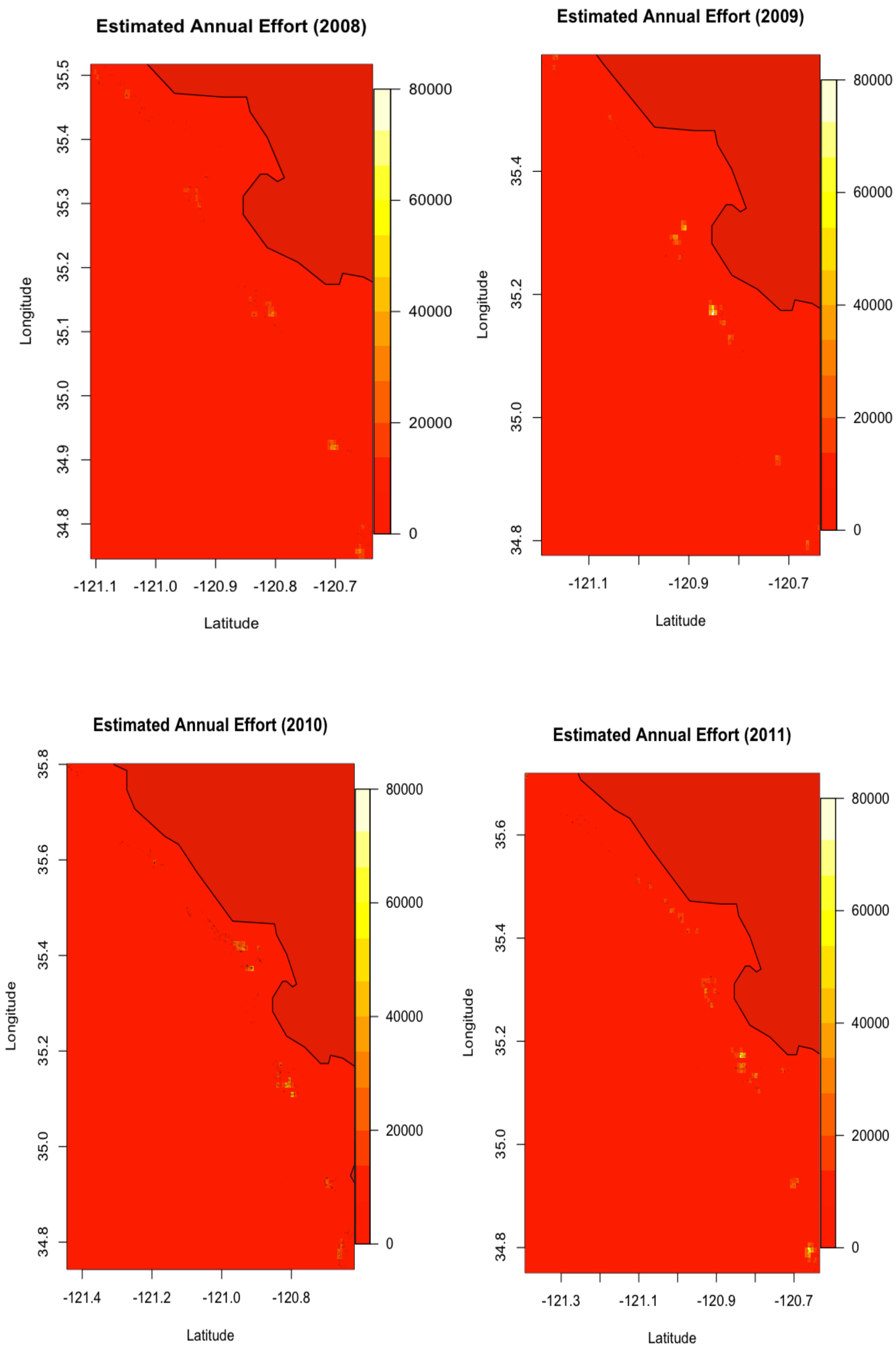
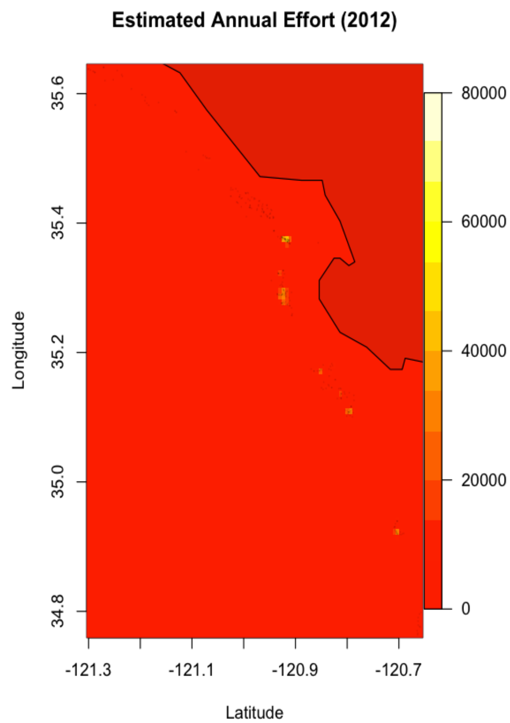


Figure 13: Post MPA (2008-2012) Estimated Annual Effort,
Continues on page 20
(Appendix lines 568-576 repeated for each year)





From looking at the nine images of annual effort, it is not perfectly clear whether or not the pattern of effort changed substantially after the marine protected areas were put in place. However, there are interesting patterns to note such as the effort generally getting larger from 2003 to 2005, which we can see based on the pixels becoming brighter. Also, in the post-mpa years the fishing effort is more condensed into specific locations, rather than being spread along the coast which could potentially be attributed to the areas where fishing was no longer allowed.

B. Estimation of Variability (Bootstrapping)

Bootstrapping our data through the two-stage model of KDE and Splines allows for the estimation of the variability that surrounds the effort estimations at each location; the estimate of this variability for each year is shown based on a heat scale ranging from 0 to 22,000. In Figure 14, the estimated variability in our data for 2003 is shown next to a zoomed in portion of the Central Coast (the same portion that was provided in Figure11). By looking at the selected portion of the coast in Figure 14, we can see that the variability around these estimates is moderately high; the fairly high standard deviations could be attributed to the small amount of data that we have, and that we are using to try to predict effort along the whole Central Coast.

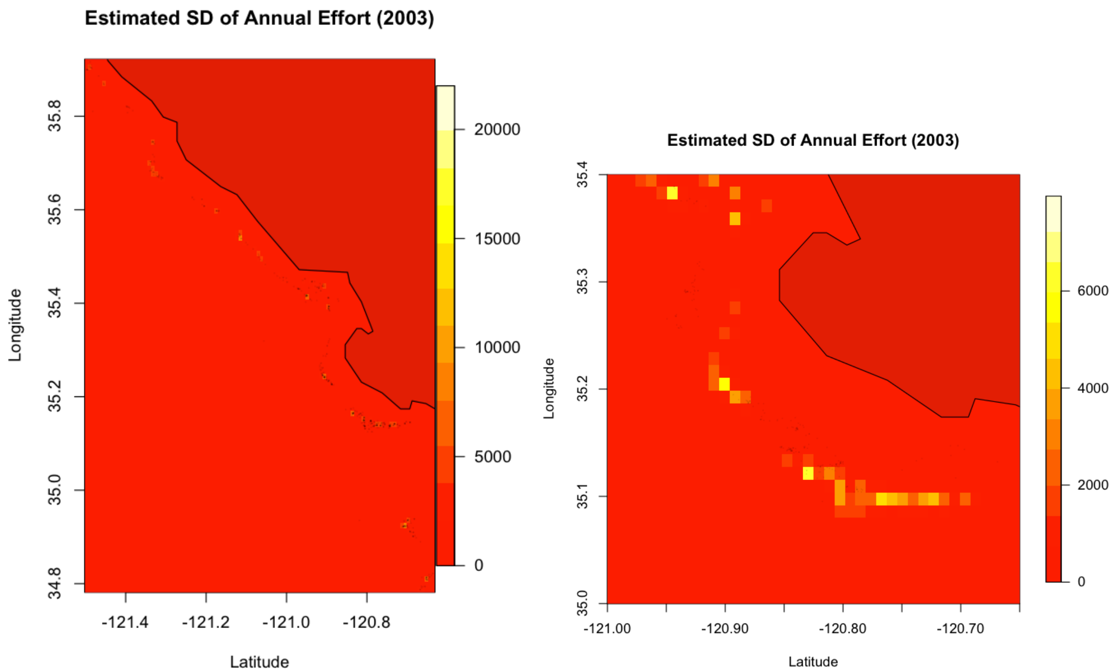


Figure 14: 2003 Estimated Annual Effort with Enhanced Portion to Illustrate Pixel-Wise Estimation of Effort (Appendix lines 579-587, 607-615)

In addition to looking at the patterns in estimated effort from year to year, looking at the estimated variability from year to year can tell us a lot about our data; to do this we will look at Figures 15 and 16 on the following pages. The years with the brightest pixels represent the years with the highest variability, and the ones that stand out are the years 2005 and 2009. These are also the years that were found to have higher estimated annual efforts. The trend appears to follow that the higher the estimated annual effort is, the higher the estimated variability will be also; this is an interesting observation to note because this is telling us that we can generally be less certain about the accuracy of our estimations when we are estimating higher values of effort. There does not appear to be any significant differences in the variability between the pre-mpa years and the post-mpa years, because the variability varies greatly within the pre-mpa and post-mpa groups individually.

Figure 15: 2003 Estimated Standard Deviation of Annual Effort with Enhanced Portion to Illustrate Pixel-Wise Estimation of Effort (Appendix lines 579-587 repeated for each year)

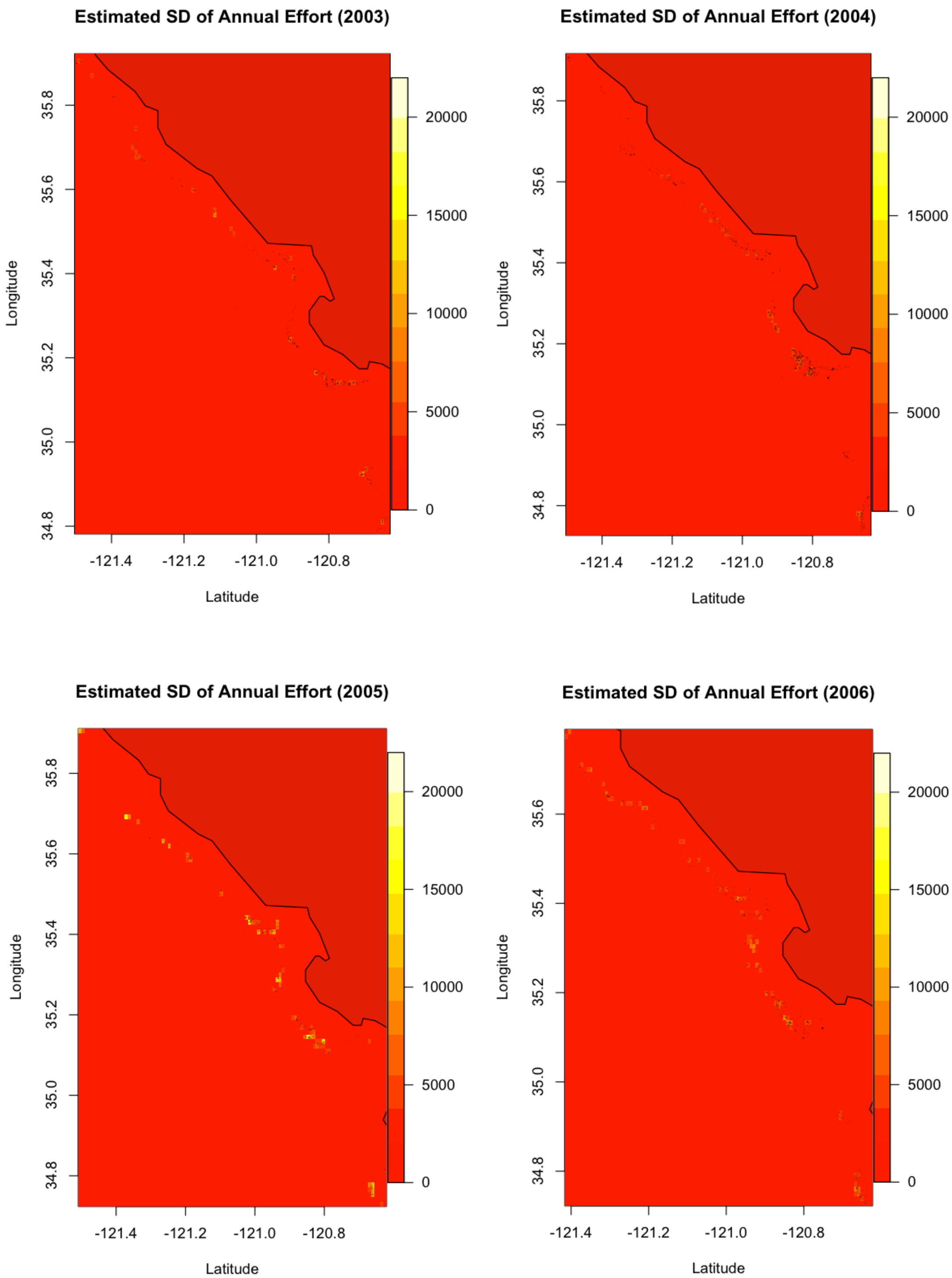
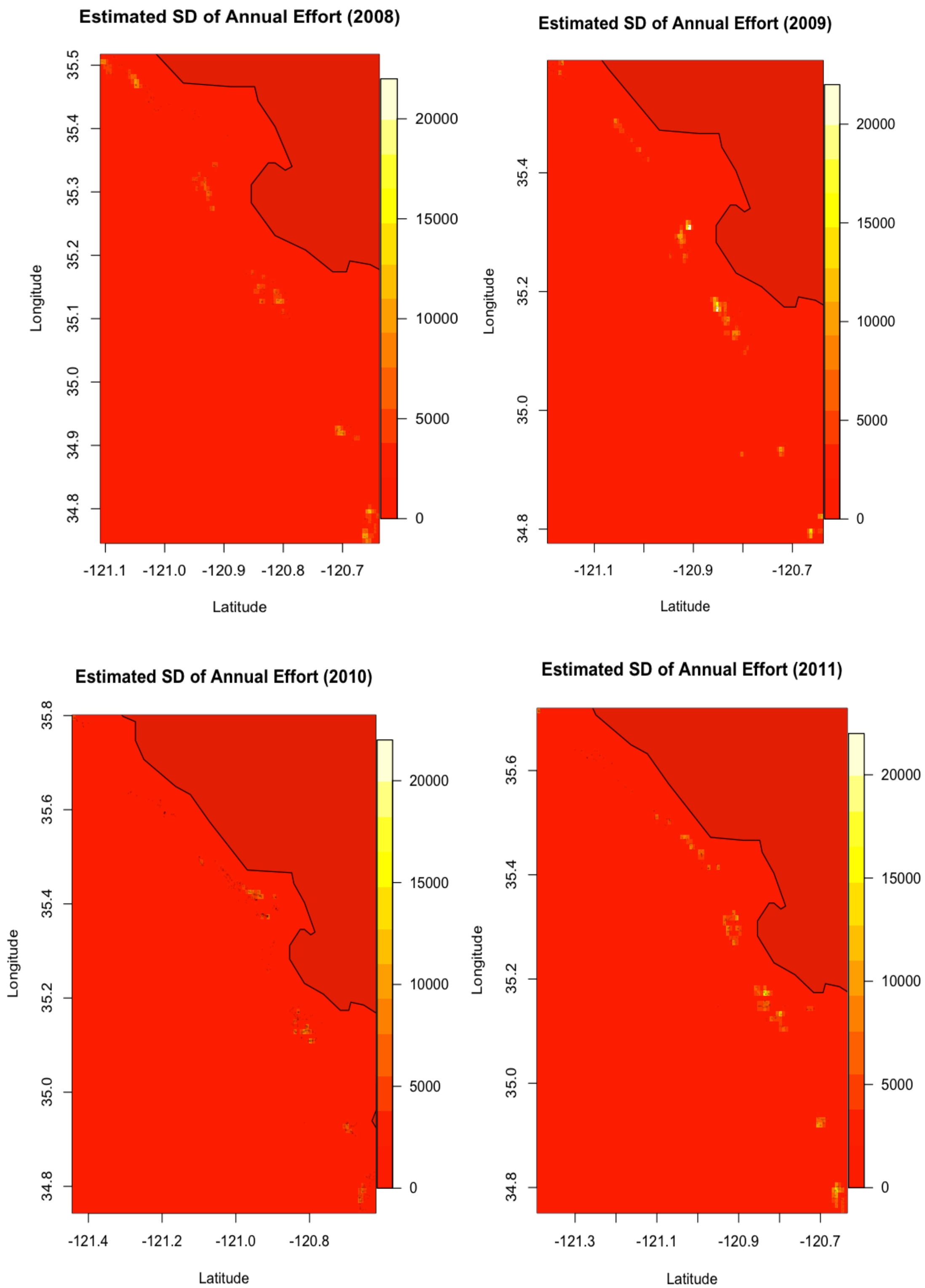
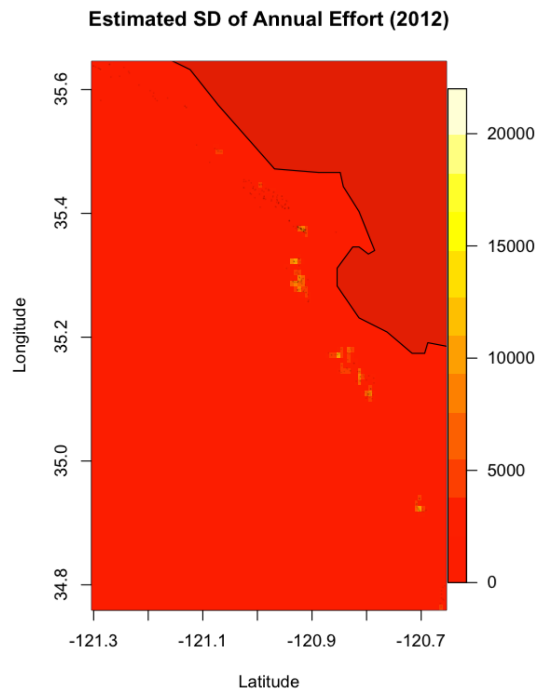


Figure 16: Post MPA (2008-2012) Estimated Standard Deviation of Annual Effort, Continues on page 24 (Appendix lines 579-587 repeated for each year)





C. Annual Effort Pre and Post MPA

One of the driving forces for this project was the question of whether or not fishing effort has changed due to the implementation of the marine protected areas along the Central Coast. Another way to investigate this question is to look at the difference in average annual effort for the pre-mpa years and the post-mpa years. To do this we will compute a 95% confidence interval for the difference in average annual effort before and after the mpa implementation, using the following values:

t_i = total annual effort averaged over all locations, for i=2003-6, 2008-12

$\hat{d}$ = - $\frac{t2003+t2004+t2005+t2006}{4}$ + $\frac{2008+2009+t2010+t2011+t2012}{5}$, the difference in average annual effort pre and post mpa

$\hat{S}^2_{ti}$ = (standard deviation of t_i)^2 , the variance of the bootstrapped annual effort

$\hat{S}_d{}^2$=$\left(\frac{\hat{S}t2003+\hat{S}t2004+\hat{S}t2005+\hat{S}t2006}{4}+\frac{\hat{S}t2008+\hat{S}t2009+\hat{S}t2010+\hat{S}t2011+\hat{S}t2012}{5}\right)^2$ the variance of the average difference in annual effort pre and post mpa.

Finally, our 95% confidence interval will take the following form:

$$\hat{d} \text{ +/- } 1.96 * \hat{S}_d.$$

Using the code provided in the Appendix, from lines 621 to 670, the resulting values are $\hat{d} = 188080.2$, and $\hat{S}_d$ = 45254.87. Using these numbers to plug in to the equation above, the resulting confidence interval is:

188080.2 +/- 1.96 * 45254.87 = (99380.68, 276779.8).

Based on this interval, we are 95% confidence that the average bootstrapped estimated annual effort after the mpa's were put in place is between 99380.68 and 276779.8 higher than the average bootstrapped estimated annual effort before the mpa's were implemented. This information tells us that, regardless of where the fishermen are fishing, they have been fishing more since the mpa's were implemented. The scope of our project does not allow us to determine why this is the case, however it could be speculated that this could be due to there being less fish outside of the mpa's, which causes the fisherman to have to fish for longer (put in more effort) in order for everyone to catch the desired number of fish.

D. Limitations

The major limitations that were come across during this project were the small proportion of total trips per year that we had data on, and the inability to calculate the average difference in annual fishing effort pre and post mpa, per pixel. The small amount of data did not pose an issue, other than the fact that it caused out estimated of effort for each location to be less accurate/have more variability than desired. The larger problem we had was when trying to calculate the difference in average effort between the two groups of years.

When writing the code, we did not yet know that we would have enough time to be able to look at the difference between the groups of years, so the code was not written to be compatible with that procedure. In order to compute the difference pixel-wise, we would have had to specify a set range for latitudes and longitudes to be used within the bootstrap, which would ensure that the limits of all 9 years would match-up. Since we did not include this specified range, the latitude and longitude limits on the 9 years are non consistent, which keeps us form being able to compute the pixel by pixel difference in average annual effort.

V. Appendix

A. R Code

```
##################################
#                                #
# KDE  examples                  #
#                                #
##################################


# load required packages
library(maps)
library(MASS)
library(mgcv)
library(fields)
library(akima)


# sample of latitudes and longitudes n=20 and efforts
lat = c(35.125, 34.91, 35.13667, 35.14867, 35.545, 34.93462,
 34.92433, 34.91, 34.91933, 35.42633, 35.427, 35.40567, 35.4315,
 35.4085, 35.1231, 35.12617, 35.6127, 35.61125, 35.63847,
 35.6975)

long = c(-120.8113, -120.6903, -120.7817, -120.7903, -121.1159, -
 120.7073, -120.7053, -120.6903, -120.6853, -120.9152, -120.9148,
 -120.938, -120.9198, -120.9412, -120.8107, -120.8015, -121.2105,
 -121.2109, -121.272, -121.3286)

effort = c(528, 296, 195, 364, 528, 444, 325, 296, 296, 315, 175,
 245, 350, 245, 182, 224, 114, 228, 95, 100)
#fishing hours, total hours all lines combined



### Examples of Kernel Density Estimation to Estimate Drop
 Densities


##EXAMPLE 1: Differences between bandwidths

#create 2x2 grid for images
par(mfrow=c(2,2))

#examples with different bandwidths and one smooth line
#bin width = .4
hist(long, prob=TRUE, main="Histogram of Longitudes (bandwidth=
 .4)",xlab="Longitude (n=20)", ylab="Density", breaks=seq(-
```

```
 121.4,-120.6, by=.4), ylim=c(0,2.5))


#bin width = .1
hist(long, prob=TRUE, main="Histogram of Longitudes (bandwidth=
 .1)",xlab="Longitude (n=20)", ylab="Density", breaks=seq(-
 121.4,-120.6, by=.1), ylim=c(0,2.5))


#bin width = .025
hist(long, prob=TRUE, main="Histogram of Longitudes (bandwidth=
 .05)",xlab="Longitude (n=20)", ylab="Density", breaks=seq(-
 121.4,-120.6, by=.05), ylim=c(0,5))


#Example of kernel density estimate
k1= density(long, bw=.1)
plot(k1,type="l",main="Kernel Density Estimate")




##EXAMPLE 2: Differences between bandwidths with rectangular
 kernels

#create 2x2 grid for images
par(mfrow=c(2,2))

#rectangular with bw=.025
k2a= density(long, bw=.025, kernel="rectangular")
plot(k2a,type="l",main="Rectangular Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Rectangular Kernel (bw=.025)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3),
 border="grey")
#lines(k2a, lwd=2)


#rectangular with bw=.05
k2b= density(long, bw=.05, kernel="rectangular")
plot(k2b,main="Rectangular Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Rectangular Kernel (bw=.05)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3),
 border="grey")
#lines(k2b)
#plot(k2b,type="l",main="Kernel Density Estimate")


#rectangular with bw=.1
```

```
k2c= density(long, bw=.1, kernel="rectangular")
plot(k2c, main="Rectangular Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Rectangular Kernel (bw=.1)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3),
 border="grey")
#lines(k2c)


#rectangular with bw=.2
k2d= density(long, bw=.2, kernel="rectangular")
plot(k2d,main="Rectangular Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Rectangular Kernel (bw=.2)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3),
 border="grey")
#lines(k2d)




##EXAMPLE 3: Differences in between bandwidths with Gaussian
 kernels

#create 2x2 grid for images
par(mfrow=c(2,2))


#gaussian with bw=.025
k3a= density(long, bw=.025, kernel="gaussian")
plot(k3a, main="Gaussian Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Gaussian Kernel (bw=.025)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3.6))
#lines(k3a)


#gaussian with bw=.05
k3b= density(long, bw=.05, kernel="gaussian")
plot(k3b, main="Gaussian Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Gaussian Kernel (bw=.05)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3.6))
#lines(k3b)


#gaussian with bw=.1
k3c= density(long, bw=.1, kernel="gaussian")
```

```
plot(k3c, main="Gaussian Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Gaussian Kernel (bw=.1)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3.6))
#lines(k3c)


#gaussian with bw=.2
k3d= density(long, bw=.2, kernel="gaussian")
plot(k3d, main="Gaussian Kernel Density Estimate")
points(long, rep(0, length(long)))

#hist(long, prob=TRUE, main="Gaussian Kernel (bw=.2)",
 xlab="Longitude (n=20)", ylab="Density", ylim=c(0,3.6),
 border="light grey", col="grey")
#lines(3d)




#################################
#                               #
# Spline examples               #
#                               #
#################################

### Examples for spline estimation of effort

#Graph #1: generic example of knots in spline
x1=seq(0,10,length=100)
x2=seq(10,20,length=100)
y1=3 + 2*x1 - x1^2 + .5*x1^3
y2=423 + 2*x1 + x1^2


plot(c(x1,x2), c(y1,y2), xlab="X", ylab="Y", main="Piecewise Spline
 with Knot at x=10", type="l")



#sample of latitudes and longitudes n=20 and efforts

lat = c(35.125, 34.91, 35.13667, 35.14867, 35.545, 34.93462,
 34.92433, 34.91, 34.91933, 35.42633, 35.427, 35.40567, 35.4315,
 35.4085, 35.1231, 35.12617, 35.6127, 35.61125, 35.63847, 35.6975)

long = c(-120.8113, -120.6903, -120.7817, -120.7903, -121.1159, -
 120.7073, -120.7053, -120.6903, -120.6853, -120.9152, -120.9148, -
 120.938, -120.9198, -120.9412, -120.8107, -120.8015, -121.2105, -
 121.2109, -121.272, -121.3286)

effort = c(528, 296, 195, 364, 528, 444, 325, 296, 296, 315, 175,
```

```
 245, 350, 245, 182, 224, 114, 228, 95, 100)
#fishing hours, total hours all lines combined



#Graph #2: Scatterplot of sample n=20 of data
plot(long, effort, xlab="Longitude", ylab="Effort",
 main="Scatterplot of Effort by Longitude", pch=16)




#Graph #3: Predicted effort by longitude

y.pred = predict(effort.gam, data.frame(long=x.long), se=TRUE)
plot(x.long, y.pred$fit, type="l", ylim=range(effort),
 xlab="Longitude", ylab="Predicted Effort", main="Predicted Effort
 by Longitude")
lines(x.long, y.pred$fit-y.pred$se, lty=2)
lines(x.long, y.pred$fit+y.pred$se, lty=2)
points(long, effort)
axis(1, at=long, labels=NA, tcl=.5)
y.pred



#################################
#                               #
#    Bar Graph                  #
#                               #
#################################

#bar graph of total trips per year

year=c("2003","2004","2005","2006","2007","2008","2009","2010","2011
","2012")
TotalTrips=c(2731, 3580, 3213, 3556, 3329, 4114, 4188, 3790, 4239,
 3787)

barplot(height=TotalTrips, names.arg=year, xlab="Year", ylab="Number
 of Trips", main="Total Trips per Year", col=c("springgreen3",
 "springgreen3", "springgreen3","springgreen3", "gray", "turquoise",
 "turquoise", "turquoise", "turquoise", "turquoise"))




###########################################
#                                         #
# Part Boat Fishing                R Code #
#                                         #
###########################################
```

```r
#load required packages
library(maps)
library(MASS)
library(mgcv)
library(fields)
library(akima)

#Read in data
#setwd("/Volumes/USB/Senior Project")
#setwd("F:/Senior Project")
setwd("/Users/samdellinger/Documents/Cal Poly 2013-14/Senior Project")
fishing.data<- read.table('FishingFullDataSet.csv', header= T, sep=",")

##Run through all of analyses individually for each year

# subset data into years; run invididually
#remove outlier in 2005 subset of data
fishing.data = subset(fishing.data, subset = Year == 2003)
#fishing.data = subset(fishing.data, subset = Year == 2004)
#fishing.data = subset(fishing.data, subset = Year == 2005&LongDD>-250)
#fishing.data = subset(fishing.data, subset = Year == 2006)
#fishing.data = subset(fishing.data, subset = Year == 2008)
#fishing.data = subset(fishing.data, subset = Year == 2009)
#fishing.data = subset(fishing.data, subset = Year == 2010)
#fishing.data = subset(fishing.data, subset = Year == 2011)
#fishing.data = subset(fishing.data, subset = Year == 2012)


#Plot data with scatterplot
plot(fishing.data$LongDD, fishing.data$LatDD, main='Scatterplot of Lat vs. Long', xlab='LatDD', ylab='LongDD')


#Useful base map for properly scaled plots
# run this map without any graphics windows first, then
# all subsequent graphics will be have aspect ratios
# that are scaled properly


# Ran alone creates simple image of CA coastlines
map(database = "state", regions = "CA", xlim = range(fishing.data$LongDD)+c(-.1,.1), ylim = range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col = 'darkgreen')
# add drops to above plot
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col = rgb(0,0,0,.1), cex = .2)


# Store map info
map.info =   map(database = "state", regions = "CA", xlim =
```

```
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 'darkgreen')
map.plot = cbind(x = map.info$x, y = map.info$y)

# Plot of effort with circles proportional to effort
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 'darkgreen')
symbols(fishing.data$LongDD, fishing.data$LatDD, circles =
 fishing.data$Effort, fg = rgb(0,0,0,.05), inches = .5, add = TRUE)




########################################
#                                      #
#     KDE to estimate drop denisty     #
#                                      #
########################################

#bandwidth previously determined using clutser analysis; median
 fishing spot size

kde = with(fishing.data, kde2d(LongDD, LatDD, n=100, h=.0155))
image.plot(kde, col = heat.colors(50))
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 'darkgreen', add = TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)

#change years
title("KDE (2003)")

#total drops
#different number of trips per year

N = 2731 #2003
#N = 3580 #2004
#N = 2313 #2005
#N = 3556 #2006
#N = 4114 #2008
#N = 4188 #2009
#N = 3790 #2010
#N = 4239 #2011
#N = 3787 #2012

pixel.ar = diff(kde$x[1:2]) * diff(kde$y[1:2])
tot.est.drops = N * kde$z/sum(kde$z * pixel.ar) * pixel.ar
image.plot(kde$x, kde$y, tot.est.drops, col = heat.colors(50), xlim
 = range(fishing.data$LongDD)+c(-.1,.1), ylim =
```

```
 range(fishing.data$LatDD)+c(-.1, .1))
map(database = "state", regions = "CA", fill = TRUE, col =
 'darkgreen', add = TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Total Estimated number of Drops in pixel")
sum(tot.est.drops)
# evidence of some bias, so divide kde$z by sum(kde$z) in calcs




########################################
#                                      #
#Spline (within GAM) to estimate effort#
#                                      #
########################################

gam.fit = gam(Effort ~ s(LongDD, LatDD), data = fishing.data)
summary(gam.fit)
vis.gam(gam.fit, plot.type = "contour", main = NULL)
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 rgb(0,100/256,0,.1), add = TRUE)
symbols(fishing.data$LongDD, fishing.data$LatDD, circles =
 fishing.data$Effort, fg = rgb(0,0,0,.05), inches = .5, add = TRUE)
title("Effort")


#predicted values
pred.vals = data.frame(expand.grid(LongDD = kde$x, LatDD = kde$y))
effort.pred = matrix(predict(gam.fit, pred.vals), ncol = 100)
image.plot(kde$x, kde$y, effort.pred, col = heat.colors(50),
 xlab="Latitude", ylab="Longitude")
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 rgb(0,100/256,0,.1), add = TRUE)
symbols(fishing.data$LongDD, fishing.data$LatDD, circles =
 fishing.data$Effort, fg = rgb(0,0,0,.1), inches = .5, add = TRUE)
title("Effort")





####################
#                  #
#   TOTAL EFFORT   #
#                  #
####################

#different number of trips per year
```

```
N = 2731 #2003
#N = 3580 #2004
#N = 2313 #2005
#N = 3556 #2006
#N = 4114 #2008
#N = 4188 #2009
#N = 3790 #2010
#N = 4239 #2011
#N = 3787 #2012

pixel.ar = diff(kde$x[1:2]) * diff(kde$y[1:2])
#change year in prods
prods2003 = N * kde$z/sum(kde$z * pixel.ar) * pixel.ar * effort.pred

#change year in titles and withing image.plot
#par(mfrow=c(1,2))

#need heat scale the same on all years
image.plot(kde$x, kde$y, prods2003, col = heat.colors(50),
 xlab="Latitude", ylab="Longitude", zlim=c(0,80000))
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 rgb(0,100/256,0,.1), add = TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Annual Effort (2003)")

###zoomed in portion of the graph for better visualization
#image.plot(kde$x, kde$y, prods2003, col = heat.colors(50), xlim=c(-
 121.0, -120.65), ylim=c(35, 35.4), xlab="Latitude",
 ylab="Longitude")
#map(database = "state", regions = "CA", xlim = c(-121.0,-120.5),
 ylim = c(35,35.4), fill = TRUE, col = rgb(0,100/256,0,.1), add =
 TRUE)
#points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
#title("Annual Effort (2003)")



#####################
#                   #
#     Bootstrap     #
#                   #
#####################

set.seed(226)
B = 5000 # number of bootstraps
# total number of drops made by fishing company (monitored + not)

N = 2731 #2003
#N = 3580 #2004
```

```
#N = 2313 #2005
#N = 3556 #2006
#N = 4114 #2008
#N = 4188 #2009
#N = 3790 #2010
#N = 4239 #2011
#N = 3787 #2012

n = nrow(fishing.data)
kde.n = 100 # resolution of model
# fit kde just to get pixels and base pixels set
kde = with(fishing.data, kde2d(LongDD, LatDD, n=kde.n, h=.0155))
kde.lims = c(range(fishing.data$LongDD), range(fishing.data$LatDD))
image.plot(kde, col = heat.colors(12))
pixel.ar = diff(kde$x[1:2]) * diff(kde$y[1:2])


#change year in prods.array and within bootstrap loop
prods.array2012 = array(NA,c(kde.n,kde.n,B))
for(B.i in 1:B){
 bootdata = fishing.data[sample(1:n, replace = TRUE),]
 kde.b = with(bootdata, kde2d(LongDD, LatDD, n=kde.n, lims =
 kde.lims, h=.0155))
 gam.b = gam(Effort ~ s(LongDD, LatDD), data = bootdata)
 pred.b = data.frame(expand.grid(LongDD = kde$x, LatDD = kde$y))
 effort.b = matrix(predict(gam.b, pred.b), ncol = kde.n)
 prods.array2012[,,B.i] = N * kde.b$z/sum(kde.b$z * pixel.ar) *
 pixel.ar * effort.b
 }



#change years
effort.mean2012 = apply(prods.array2012, c(1,2), mean)
effort.sd2012 = sqrt(apply(prods.array2012, c(1,2), var))


#save objects as .Rdata to avoid re-running bootstraps
#change years
save(prods.array2012, effort.mean2012, effort.sd2012,
 file="2012Boot5000Data.Rdata")


##determine max range for mean and sd to set limits for image.plot

#2003-2010
load(file="/Volumes/USB/Senior Project/2005Boot5000Data.Rdata")
#2011-2012
load(file="/Users/samdellinger/Documents/Cal Poly 2013-14/Senior
 Project/2012Boot5000Data.Rdata")


range(effort.mean2003)
```

```
#2003 mean range= 0, 30385.87
#2004 mean range= 0, 33821.26
#2005 mean range= 0, 74563.47
#2006 mean range= 0, 49182.44
#2008 mean range= 0, 39058.74
#2009 mean range= 0, 79894.28
#2010 mean range= 0, 40188.94
#2011 mean range= 0, 65443.65
#2012 mean range= 0, 43507.75

##max mean range = (0, 79894.28) so set zlim=c(0,80000)


range(effort.sd2003)

#2003 sd range= 0, 7955.95
#2004 sd range= 0, 5882.436
#2005 sd range= 0, 16477.52
#2006 sd range= 0, 8720.045
#2008 sd range= 0, 12063.52
#2009 sd range= 0, 21678.58
#2010 sd range= 0, 8773.712
#2011 sd range= 0, 1502.64
#2012 sd range= 0, 9763.391

##max sd range = (0, 21678.58) so set zlim=c(0,22000)



#################################
#                               #
# Images of Bootstrap Estiamtes #
# of Effort and Variability     #
#                               #
#################################


#change years in effort.mean and effort.sd
#par(mfrow=c(1,2))

image.plot(kde$x, kde$y, effort.mean2005, col = heat.colors(12),
 xlab="Latitude", ylab="Longitude", zlim=c(0, 80000))
map(database = "state", regions = "CA", xlim =
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 rgb(0,100/256,0,.1), add = TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Estimated Annual Effort (2005)")


image.plot(kde$x, kde$y, effort.sd2005, col = heat.colors(12),
 xlab="Latitude", ylab="Longitude",  zlim=c(0, 22000))
map(database = "state", regions = "CA", xlim =
```

```
 range(fishing.data$LongDD)+c(-.1,.1), ylim =
 range(fishing.data$LatDD)+c(-.1, .1), fill = TRUE, col =
 rgb(0,100/256,0,.1), add = TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Estimated SD of Annual Effort (2005)")

##zoomed in portions of bootstrapped graphs
#change years in effort.mean and effort.sd
#par(mfrow=c(1,2))

image.plot(kde$x, kde$y, effort.mean2003, col = heat.colors(12),
 xlim = c(-121.0,-120.65), ylim = c(35,35.4), xlab="Latitude",
 ylab="Longitude")
map(database = "state", regions = "CA", xlim = c(-121.0,-120.65),
 ylim = c(35,35.4), fill = TRUE, col = rgb(0,100/256,0,.1), add =
 TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Estimated Annual Effort (2003)")

image.plot(kde$x, kde$y, effort.sd2003, col = heat.colors(12), xlim
 = c(-121.0,-120.65), ylim = c(35,35.4), xlab="Latitude",
 ylab="Longitude")
map(database = "state", regions = "CA", xlim = c(-121.0,-120.65),
 ylim = c(35,35.4), fill = TRUE, col = rgb(0,100/256,0,.1), add =
 TRUE)
points(fishing.data$LongDD, fishing.data$LatDD, pch = 16, col =
 rgb(0,0,0,.1), cex = .2)
title("Estimated SD of Annual Effort (2003)")

###DIFFERENCE IN (average) ANNUAL EFFORT pre-mpa and post-mpa

#total effort per year
t2003 = mean(apply(prods.array2003, 3, sum))
t2004 = mean(apply(prods.array2004, 3, sum))
t2005 = mean(apply(prods.array2005, 3, sum))
t2006 = mean(apply(prods.array2006, 3, sum))

t2008 = mean(apply(prods.array2008, 3, sum))
t2009 = mean(apply(prods.array2009, 3, sum))
t2010 = mean(apply(prods.array2010, 3, sum))
t2011 = mean(apply(prods.array2011, 3, sum))
t2012 = mean(apply(prods.array2012, 3, sum))
```

```
#difference in total effort pre and post mpas
d.hat = .2 * (t2008+t2009+t2010+t2011+t2012) - .25 *
 (t2003+t2004+t2005+t2006)


#estimated variances of total effort
s.hat.2003 = var(apply(prods.array2003, 3, sum))
s.hat.2004 = var(apply(prods.array2004, 3, sum))
s.hat.2005 = var(apply(prods.array2005, 3, sum))
s.hat.2006 = var(apply(prods.array2006, 3, sum))

s.hat.2008 = var(apply(prods.array2008, 3, sum))
s.hat.2009 = var(apply(prods.array2009, 3, sum))
s.hat.2010 = var(apply(prods.array2010, 3, sum))
s.hat.2011 = var(apply(prods.array2011, 3, sum))
s.hat.2012 = var(apply(prods.array2012, 3, sum))


#estimated variance for the difference in total effort
#(1/5)^2 = .04   (1/4)^2 = .0625
s.hat.d = sqrt(.04*s.hat.2003 + .04*s.hat.2004+ .04*s.hat.2005+
 .04*s.hat.2006+ .0625*s.hat.2008+ .0625*s.hat.2009+
 .0625*s.hat.2010+ .0625*s.hat.2011+ .0625*s.hat.2012)


#create confidence interval
d.ll = d.hat - (1.96 * s.hat.d)
d.ul = d.hat + (1.96 * s.hat.d)


#summary of parts to confidence interval
d.hat
s.hat.d
d.ll
d.ul
```

B. Record of Hours

Date	Start Time	End Time	Hours
9/30/13	10:30 AM	11:30 AM	1
10/4/13	10:00 AM	12:00 PM	2
10/9/13	11:00 AM	12:00 PM	1
10/24/13	2:00 PM	3:00 PM	1.5
10/25/13	10:00 AM	12:00 PM	2
10/26/13	1:30 PM	2:30 PM	1
10/28/13	11:30 AM	12:00 PM	0.5
11/1/13	10:00 AM	11:00 AM	1
11/20/13	11:00 AM	1:00 PM	2
11/23/13	9:00 AM	11:00 AM	2

12/1/13	7:00 PM	7:30 PM	0.5
12/4/13	11:00 AM	12:00 PM	1
12/7/13	12:00 PM	2:00 PM	2
12/11/13	11:00 AM	12:00 PM	1
1/8/14	11:00 AM	11:00 AM	1
1/9/14	12:30 PM	2:00 PM	1.5
1/14/14	1:00 PM	2:00 PM	1
2/4/14	1:00 PM	2:00 PM	1
2/5/14	9:00 AM	10:00 AM	1
2/6/14	10:00 AM	11:00 AM	1
2/7/14	11:00 AM	12:00 PM	1
2/20/14	9:00 AM	10:00 AM	1
	7:30 PM	9:30 PM	2
2/21/14	6:30 PM	9:30 PM	3
3/4/14	9:30 AM	10:30 AM	1
3/6/14	12:00 PM	3:30 PM	3.5
3/13/14	9:00 AM	10:00 AM	1
3/17/14	6:00 PM	10:00 PM	4
3/18/14	9:00 AM	10:00 AM	1
4/3/14	11:30 AM	1:00 PM	1.5
4/7/14	3:00 PM	5:00 PM	2
4/10/14	9:00 AM	10:00 AM	1
4/16/14	1:00 PM	4:00 PM	3
4/21/14	9:00 AM	12:00 PM	3
4/24/14	10:00 AM	11:00 AM	1
	12:00 PM	2:00 PM	2
4/28/14	3:00 PM	7:00 PM	4
5/9/14	10:00 AM	2:30 PM	4.5
5/13/14	10:00 AM	12:30 PM	2.5
5/15/14	10:00 AM	11:00 AM	1
5/20/14	11:30 AM	11:00 PM	1.5
5/22/14	10:00 AM	11:00 AM	1
5/25/14	9:00 AM	12:00 PM	3
5/26/14	10:00 AM	1:30 PM	3.5
	5:00 PM	10:30 PM	5.5
5/27/14	12:00 PM	1:30 PM	1.5
28-May	7:00 PM	12:00 PM	5
5/29/14	10:00 AM	12:00 PM	2
	7:30 PM	10:00 PM	2.5
5/30/14	11:00 AM	12:30 PM	1.5
6/2/14	7:00 PM	9:30 PM	2.5

6/3/14	11:00 AM	4:00 PM	5
6/4/14	12:30 PM	2:00 PM	1.5
6/7/14	9:30 AM	11:00 PM	1.5
6/8/14	4:30 PM	8:00 PM	3.5
		Total:	109

VI. Works Cited

Breheny, Patrick. *Kernel density estimation*, http://web.as.uky.edu/statistics/users/pbreheny/621/F12/notes/10-18.pdf

California Marine Protected Areas, *Central Coast MPA Region*. http://www.californiampas.org/pages/regions/centralcoast.html (January 2014).

Clark, Michael,.*Generalized Additive* Models, Center for Social Research University of Notre Dame, Page 7, http://www3.nd.edu/~mclark19/learn/GAMS.pdf

Zucchini, Walter,.*Part I: Kernel Density Estimation,* Applied Smoothing Techniques, http://isc.temple.edu/economics/Econ616/Kernel/ast_part1.pdf