Using Survival Analysis Methods to Study Santa Barbara County Divorces

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

Bachelors of Science

By

Joel Vazquez

June, 2011

Contents

Introduction

My introduction to survival analysis methods came during the winter 2010 quarter when I took Stat 417, Survival Analysis, with Dr. Sklar. Here, I learned how to draw inference on time-to-event data. We analyzed various time-to-event data sets, learned what left/right censoring meant, along with its correct use in a models with survival analysis techniques. Also, in the course our grade depended on three items: a midterm, a final, and an out-of-class project. For the out of class project I was paired up with two other individuals and came up with the interesting idea.

For our project, we used a data set that involved how many children a couple had, when they married, when they separated, and their corresponding finalization of their divorce. One complication that arose was that not all divorces were completed, that is some couples hadn't finalized the divorce. Here came the first application of what I learned in Survival Analysis. Before any of the analysis could be run we had to create two variables, one being the length of marriage (date of separation – date of marriage) and the other being time to judgment (date if their divorce finalization – date of separation). Due to the fact that not all of the divorces were finalized, I had to create a right censoring indicator variable and set the date of divorce finalization for each of these events to the date of when the data was collected, 06/11/03.

We had a fair amount of work in hand when dealing with data before it was all done but I drew a great interest from this form of data analyses, so much so that I chose to do something similar for my senior project. With the aid of Dr. Frame, I have been able to research a similar data set that was collected in Santa Barbara County instead of San Luis Obispo County. The complete data set used can be found on a pdf file online.

San Luis Obispo Data Collection Process

During my third year (Winter 2010) at Cal Poly- San Luis Obispo I chose to take survival analysis, Stat 417. Towards the end of the quarter each student was assigned to a group of three to four students and each group was given the task of finding a data set which we could implement survival analysis methods on. At first, this seemed a simple task but as time went by we found this process to be rather hard. Search engines such as Bing, Yahoo, and Google could not find sufficient data to fit the need of our survival analysis project. After restless hours of searching for data sets online we decided to make a data set of our own. One of the project members acquired this idea from a recent project one of the Statistic Department members had recently finished. This data set concerned divorces in Santa Barbara County. The data set originally used had four variables: a variable that indicated whether the couple had a child or not, the date the couple was married, the date of the couples separation, and the date their divorce was finalized. We could have easily used this data set but due to the fact we had to personally collect the data this wouldn't apply.

We chose to collect a similar data set but instead of it concerning Santa Barbara County divorces concerned San Luis Obispo County divorces. Even though this process of which I will talk about was extremely time consuming it was possible due to the fact that divorce files are public records. The first thing you must do is find the San Luis Obispo Courthouse Annex address. The address for this location is 1035 Palm Street, San Luis Obispo, CA, 93401. This entrance is about 100 feet southwest from the Santa Rosa and Palm Street intersection. Once you have arrived at this location, you must pass through metal detector. This process usually doesn't take long but one must leave sharp objects at home. Once you've passed the metal detector you should enter room 385 which is the first door on your left. You will then approach

an open window and ask for the Microfiche files that hold the divorce information. There are two microfiche projectors on the far right corner of the room that allow you to view the files. The files vary from about four slides to some that are fifty slides. The variable values for each observation are at times hard to find and therefore about three minutes should be expected in finding all of the variables for each case. What our group chose to use was a Microsoft Excel spreadsheet to write down all of the observation values but any sort of spreadsheet should do.

Santa Barbara County Divorce Data

As stated in the background section, the data set used includes only Santa Barbara County divorces. The data set originally consisted of over 300 observations, but due to some observations having missing entries, the data set we actually used only consisted of 287 observations. From the raw dataset found online, I was able to collect four variables: date of marriage, date of separation, date of judgment, and a variable indicating whether the couple had children or not. From these four variables, I was able to produce two more variables. One indicating how long the couple was married (in years) and the other indicating how long it took for the divorce to be finalized (in months). I was then able to create a marriage duration categorical variable. The categorical variable has four levels, very short, short, medium, and long. The very short level accounts for the marriages that last for less than one year, the very short level accounts for marriages that span one year to five years, the medium level accounts for marriages that span five to ten years, and the long level accounts for marriages that last longer than ten years.

A necessary measure had to be taken when developing the amount of time until the divorce was finalized. Since some divorces hadn't been finalized, I had to specify the date of collection as the data of finalization. This measure was taken due to the fact we only know of

each time-to-event observation up to this day of its collection. Whether the divorce was ever actually finalized is outside of what we can figure out from the data set. For each of these observations, 06/11/03 was set as the date of judgment. To account for variability that comes from these observations, I created a censor indicator variable. This allows for each of the 287 observations to have a time to judgment response and thus be included in the making of each of our statistical models.

Methods

When analyzing a quantitative response variable by other quantitative or indicator variables, a model that usually applies is a regression model. A regression model leads to a functional relationship between a response and a set of explanatory variables. A regression model indicates which explanatory variables have an effect on the response variable, in this case time to judgment. A regression model allows us to ask "what if" type of questions. In the context of my data one can ask what if a couple had a child and what if they were married for an extended period of time rather than short. A regression model allows us to estimate the mean time to judgment for different circumstances.

The coefficient for each explanatory variable level will be assessed at the five percent significance level. For any conclusion to be valid we must check the assumptions that are necessary for a regression model. The assumptions necessary for a regression model are normality and constant variance. To assess that the assumption of normality of the error terms is met we usually first look at a Normal Probability Plot. If a Normal Probability Plot shows departure from the straight diagonal line, which represents normality, there is reason to believe that this assumption is violated. If this assumption is violated then many problems can arise, the most problematic in this situation is that tests used to assess the significance of terms in our models are possibly compromised. To assess the constant variance assumption of the error terms we look at a Residual versus Fitted Values Plot, a plot of the error terms against the values that the model predicts for that value. If the plot has a rectangular (random scatter) look to it then this is evidence that this assumption is not violated. Violations of this assumption make estimating the precision that we are able to have when estimating parts of our model we wish to estimate.

To assess which model best fits our response variable, time to judgment, it is appropriate to use a partial f test and use the leave one out cross-validation technique. In using a partial f test to compare models we are testing whether or not certain predictor variables are necessary when predicting our response, time to judgment. This type of model selection technique allows us to compare a simplified regression models versus fuller models. For a partial f test to be appropriate, all variables used in the simplified version of the regression model must be in the full regression model. In this case an example of a null and alternative hypothesis is as follows:

H_0: $y = \beta_0 + \beta_1 + \beta_2$ (smaller model)

H_1: $y = \beta_0 + \beta_1 + \beta_2 + \beta_3 + \beta_4 + \beta_5$ (larger model)

**If the null hypothesis is rejected, at least one of the β terms added in the larger model is said to be different from 0.*

The partial f test should lead us to the best fitting model for our given data but a cross-validation technique can be used to evaluate our model's predictive capability. The cross-validation technique used is called the leave one out method. The leave one out technique allows us simulate a "new" observation by removing data points, fitting the model without the data, using the estimated model to predict the response values, and consider the sum of squared prediction errors. The smaller the value for the error term, the better our models predictive capability is.

The only problem with analyzing our data with a regression models is that it doesn't completely account for the fact that there are right censored data points. In a way, we can account for this affect by having a censor indicator variable that specifies whether or not a divorce has been finalized. To fit a survival regression models to our complete data set I will use

the survreg function found in R. When using the survreg function in R, we identify the possibility of some observations being right-censored. One interesting aspect about this form of regression models is that a distribution must be fit to our response variable, since that is all the survreg function allows. R allows us to choose from six distributions: Weibull, Exponential, Gaussian, Logistic, Lognormal, or Log-logistic. The output coming from this model will allow me to conclude which explanatory variables are significant predictors in predicting time to judgment.

Results

Prior to running any of the previously explained statistical methods, I chose to first take a look at descriptive statistics and their corresponding graphs. The first set of descriptive statistics I looked at concerned looking at our response variable, time to judgment, by the four marriage duration level.

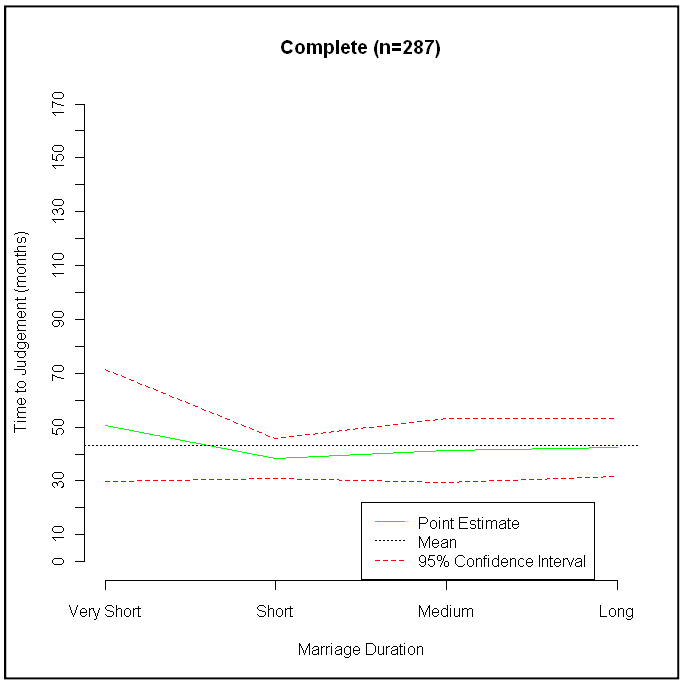

Figure 1: Time to Judgment by Marriage Duration (Complete Data Set)

The first graph I looked at is that of the complete data set with 287 observations, Figure 1. Here, comparisons can be made between the four marriage duration groups. The black dotted line corresponds to the overall time to judgment mean, the green line corresponds to the mean time to judgment by marriage duration category, and the red line represents the 95% confidence interval for time to judgment at each of the four marriage duration categories. From Figure 1 we can see that the overall means for the four categories yield different time to judgment means. The couples that correspond to the very short category, which match up to couples that have been married for less than one year, on average take the longest time to finalize their divorce.

The couples that correspond to the short marriage, on average, took the shortest time to finalize their divorce. Even though there are small differences between groups, it is worth noting that this graph, and the next two, don't necessarily account for whether the couple has children or if their divorce is finalized. It is simply looking at the complete data set without accounting for any other variables.

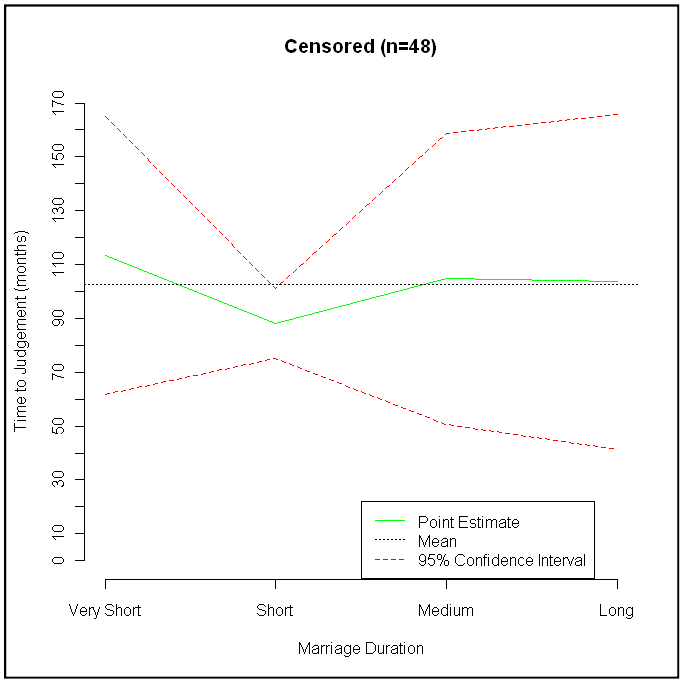

Figure 2: Time to Judgment by Marriage Duration (Censored Data Set)

If we solely look at the observations which are right censored, there is an apparent amount of variability shown within each of the four marriage groups, as seen in Figure 2. This is shown by the wide estimates for the 95% confidence intervals. There is also an apparent change in each time to judgment point estimate. The values of time to judgment that before had an overall mean of about 42 months now have a mean length of about 102 months. The mean that corresponds to the very short marriage group still yields the highest average time to judgment, while the mean that corresponds to the short marriage group yields the lowest average for time to judgment. As previously stated, the variability for each of our four variables is very large. This

may be due to a couple of things: the sample size for each of these groups is 6, 11, 12, and 13 respectively and due to the fact that all of these cases were never finalized the censor date may be far from separation.

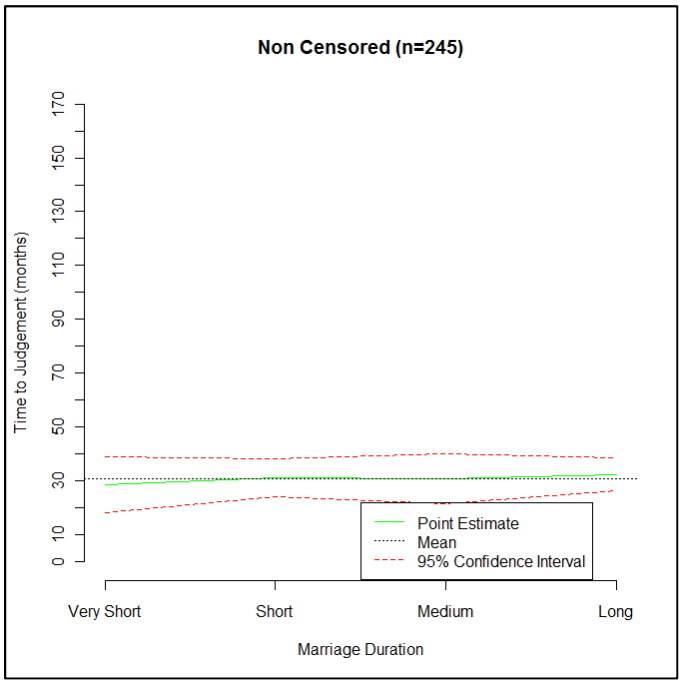

Figure 3: Time to Judgment by Marriage Duration (Non-Censored Data Set)

When looking only at the cases where all divorce cases are finalized, we can see that on average, the four marriage duration groups have similar time to judgment means. Another difference between Figure 2 and Figure 3 is the large difference in overall mean. This difference is averaged out in the graph corresponding to the complete data set, Figure 1. The group that yields the largest average are couples who have been married for at least ten years, the long marriage duration group. The variability corresponding to each of these four subgroups is also relatively small compared to the following group that amounts only of censored observations. This is mainly due to the fact that most data points which greatly deviate from the mean are removed, and their means are smaller along with the decrease in magnitude of the 95% confidence intervals.

To further investigate the variability within each data set knowledge of central moments may be used, in this case kurtosis. In Tables 1, 2, and 3, confidence intervals for time to judgment by marriage duration levels are listed with their corresponding kurtosis values. The 95% confidence intervals found on these tables are displayed on the previous three graphs (Figures 1, 2, and 3). What cannot be explained by the previous three simple graphs is the amount of variation due to extreme observations; this is where kurtosis can be used.

Table 1: Complete 95% Confidence Intervals (n=287)

Marriage Length	Lower Bound	Upper Bound
Duration = Very Short	29.97	71.40
Duration = Short	30.82	45.88
Duration = Medium	29.55	53.13
Duration = Long	31.79	53.19

Table 2: Censored 95% Confidence Intervals (n=42)

Marriage Length	Lower Bound	Upper Bound
Duration = Very Short	61.78	165.08
Duration = Short	75.02	101.02
Duration = Medium	50.66	158.79
Duration = Long	41.28	165.85

Table 3: Non-Censored 95% Confidence Intervals (n=245)

Marriage Length	Lower Bound	Upper Bound
Duration = Very Short	18.23	38.83
Duration = Short	24.12	38.38
Duration = Medium	21.56	39.98
Duration = Long	26.28	38.61

For the confidence interval values found on Tables 1, 2, and 3 the time to judgment by marriage length level that yields the largest kurtosis value is the long marriage length category found in the complete data set, 40.31. In the same data set, the time to judgment by marriage duration subgroup that yields the smallest kurtosis value is the short duration group, 3.33.

When looking at the data set, its seems as though the large kurtosis value is mainly due to one or two extreme observations. In the Complete/Long histogram, observation 258 of our complete data set has a time to judgment response value of 442.63. This sole extreme observation makes it so that the kurtosis value for the subgroup to be large and thus wider confidence interval. In attempt to grasp the effect of this sole observation, I chose to omit the observation from our data set and re-evaluate the kurtosis and confidence interval values. After removing the 258^{th} observation, the sample standard deviation for the Complete/Long subgroup drops from 51.67 to 30.02, thus showing this extreme effect of this single observation. A second observation we can make when removing the extreme value is the reduction of the 95% confidence interval width. The confidence interval drops from (31.79, 53.19) to (31.84, 44.35) when removing this single point. Lastly, the kurtosis value can be looked at. Prior to the omission of the extreme observation the kurtosis value was 40.31, amplified mainly by one observation. Once this observation is removed the kurtosis value is now 3.78. Now, the kurtosis value found in the Complete/Long group is similar to that of the Complete/Short group. This makes intuitive sense when looking at the corresponding histograms of both subgroups.

Multiple Regression Models

The first data set I chose to analyze with multiple regression models was the Non-Censored data set, where all of the divorces were finalized. For this data set, I chose to fit four separate regression models where time to judgment as the response variable and combination of marriage duration category and the children indicator as the explanatory variables. From the four models used to predict time to judgment, the model where the children indicator variable is the only predictor best fits our data adequately. The duration cat variable is insignificant at its four levels along with the interaction between children and the durationcat variable at its four

levels. Another way to further justify that this is an adequate model is by running a partial f test where we can make comparisons between models. The partial f test indicates there is a strong relationship between whether the couple has a child and how long it takes to finalize a divorce. Table 4 indicates how significant the intercept and child indicator variable actually are, at the 5% significance level. The children indicator variable coefficient can be interpreted as, if a couple has children the amount of time it takes for their divorce to be finalized is on average 10.131 months greater. Table 4 also indicates that the couples without children the reference group.

Table 4: Non-Censored Data Model (n=245)

Term	Coef	SE	T-Value	P-Value
Intercept	26.35	2.83	9.306	<0.0001
Children	10.13	4.05	2.504	0.0129
Marriage Duration = Short	1.24	8.55	0.145	0.8850
Marriage Duration = Medium	-0.24	8.64	-0.027	0.9782
Marriage Duration = Long	-0.65	8.71	-0.074	0.9408

In analyzing which model best predicts time to judgment where the possible explanatory variables are the children indicator variable and the length of their marriage categorical variable, the LOOM method indicates that a simpler model is adequate. This model includes the marriage duration category variable as the only explanatory variable, where the very short marriage duration group is the reference group. The reason why the model selected using the LOOM cross-validation method is different than the model selected when a partial f test is used is mainly due to the fact that the LOOM method only looks at the models predictive capabilities. An ANOVA table of this model tells us that none of the four marriage duration categories yield different estimates, insignificant at the 5% significance level. Another reason why these two separate techniques yield contradictory models is the fact that extreme observation may highly

influence our models. This is evident from the standard error values present on Table 4 which indicates large variability within groups.

The last data set I examined was the complete data set (n=287). The multiple regression models created for this data set are comparable to the ones made for the two previous data sets, with the addition of the censored observations being included with the non-censored observations. I was able to generate several models which included the two categorical predictors along with a combination of their interactions. From the models examined, there is no model that yields significant p-values for the corresponding terms. An example of an observed model is found in Table 5. For this model, the couples married for less than one year with children are the reference group. A reason why all terms in models for the complete data set are insignificant is due to the fact that we are not accounting for the possibility of right censored observations. Extreme values for right-censored observations don't allow us to find significant terms.

Table 5: Complete Data Set Without Censor Indicator (n=287)

Term	Coef	SE	T-Value	P-Value
Intercept	49.06	10.09	4.861	<0.0001
Children	6.23	5.86	1.064	0.2880
Marriage Duration = Short	-13.19	11.23	-1.174	0.2410
Marriage Duration = Medium	-10.76	11.34	-0.949	0.3440
Marriage Duration = Long	-10.76	11.41	-0.943	0.3460

When analyzing the nine models' predictive capabilities, the LOOM cross-validation technique indicates that the model found in Table 5 is most appropriate. This is concluded by the corresponding model having the smallest cross-validation error term value compared to other models.

Parametric Survival Regression Models

A proper way of analyzing a time-to-event data such as this is by creating a parametric survival regression model. This is possible through the survreg function found in R which fits a parametric survival regression model to our data. Since the survreg function can only be used when a distribution is assumed for the response variable, time to judgment, I chose to fit several distributions. The best fitting distribution to time to judgment is the lognormal distribution as shown in Figure 4. As the name indicates, a lognormal distribution is a random variable whose logarithm is normally distributed. Once the log of time to judgment has been taken, the data is normally distributed with mean 3.245 and standard deviation 1.019.

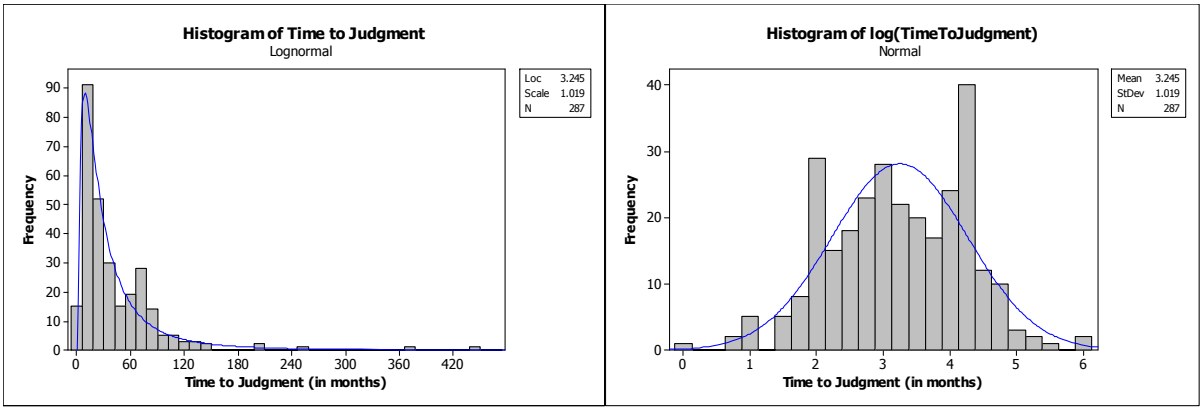

Figure 4: Lognormal Distribution Imposed on Time to Judgment

A similar procedure to previous multiple regression models is applied to our survival regression models. The fact that there are right censored time-to-event observations is accounted for by response variable, using the surv function found in R. There are only two explanatory variables we can apply to this model, them being marriage duration category and the children indicator variable. There are four separate models possible when dealing with only two explanatory variables. Of the four, the model I found appropriate was the model where both the children indicator and marriage duration category variables are the explanatory variables as seen

on Table 6. In this model, the reference groups are the couples with no children along with the couples that have been married for less than a year. At the 5 % significance level, children and duration = medium are significant predictors of variation in our model. Even though there are only two significant predictors at the 5% significance level in our model our model, also shows that Duration = Short, and Duration = Long are also moderately significant. If we were to run our model at the 10% significance level, all of the predictors in the model would be significant.

When interpreting the results seen in Table 6 one must consider the fact that I took the log of the original response variable. To adequately interpret each case, we must first sum their coefficients and then exponetiate the entire value. We can estimate time to judgment for a couple with children that are married five to ten years. We first sum the coefficient values that corresponding to the intercept, children indicator variable, and the Marriage Duration = Medium term, which amounts to a value of 3.53. We expenetiate this value and now have a time to judgment estimate for the couple. This same process can be applied to each of the other cases.

Table 6: Survival Regression Model (Lognormal Distribution)(n=287)

Term	Coef	SE	Z-Value	P-Value
Intercept	3.52	0.24	14.530	<0.0001
Children	0.56	0.14	4.060	<0.0001
Marriage Duration = Short	-0.49	0.27	-1.840	0.0665
Marriage Duration = Medium	-0.55	0.27	-2.030	0.0420
Marriage Duration = Long	-0.49	0.27	-1.780	0.0753

Lastly for the Santa Barbara County divorce survival regression model, I decided to make predictions for each of the marriage duration by the children indicator variable. These estimates, found on Figure 5, show the differences between each of the eight subcategories. From Figure 5, the same prediction pattern is followed for both couple with children and no children due to the fact there is no significant interaction between these two categorical groups. Also from Figure 5,

couples that are married for less than one year and have children yield the largest time to judgment prediction, 59.14 months. These predictions can be computed from the coefficient values found in Table 6, but a graphical representation allows us to see the magnitude in differences.

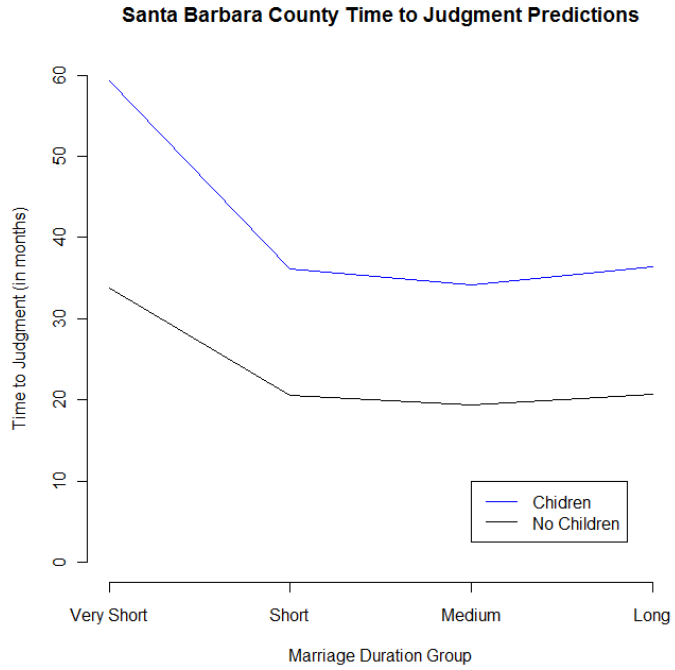

Figure 5: Santa Barbara County Time to Judgment Predictions

The last statistical procedure I chose to produce for my senior project was to apply a similar parametric survival regression model to the data I collected for my Stat 417 project. A problem that arises when using this data set is that it is rather small. With a small data set, there may exist a problem in detecting significant differences when in reality one exists. This is due to the large sample variation due to the small sample size. Another problem that arises is visually evident in Figure 6. Since we are fitting a parametric regression model to our data, the time to judgment response variable must fit one of the six distributions given by the survreg function. From the six distributions I imposed on the data, the lognormal distributed fit best. Once the log

has been taken, time to judgment seems to somewhat follow a normal distribution with a mean of 0.0182 and standard deviation 0.475.

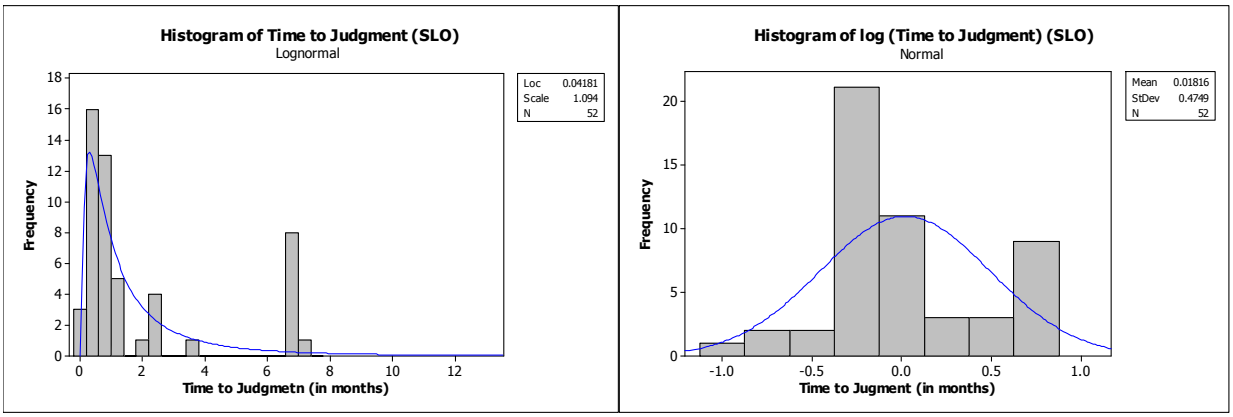

Figure 6: Lognormal Distribution Imposed of Time to Judgment (SLO Data)(n=52)

When comparing the four models created, the model that best fits our data is where the length of marriage categorical variable is the only explanatory variable, found in Table 7. The reference level in this model are the marriages that lasted than one year. When comparing the three additional categories in our model, only the medium duration category is significantly different. The two other categories aren't significant at any relevant significance level. As stated before, the standard error is rather large and this it is hard to find significant results.

Table 7: Survival Regression Model for SLO County (n=52)

Term	Coef	SE	Z-Value	P-Value
Intercept	-0.50	0.78	-0.633	0.5267
Children	-0.35	0.34	-1.023	0.3064
Duration = Short	1.45	0.84	1.726	0.0844
Duration = Medium	2.13	0.88	2.425	0.0153
Duration = Long	1.45	0.87	1.409	0.1590

We get a last look at any differences between couples' time to judgment with predictions. A few things are obvious when looking at the predictions found in Figure 7. First is that the lines separated by whether a couple has children is parallel. This is due to the fact that the interaction between these two categorical predictors was insignificant. Next, the couple with children

yielded larger predictions, similar to our predictions for the Santa Barbara County survival model. Lastly, couples with children and were married five to ten years yielded the largest response, about five months.

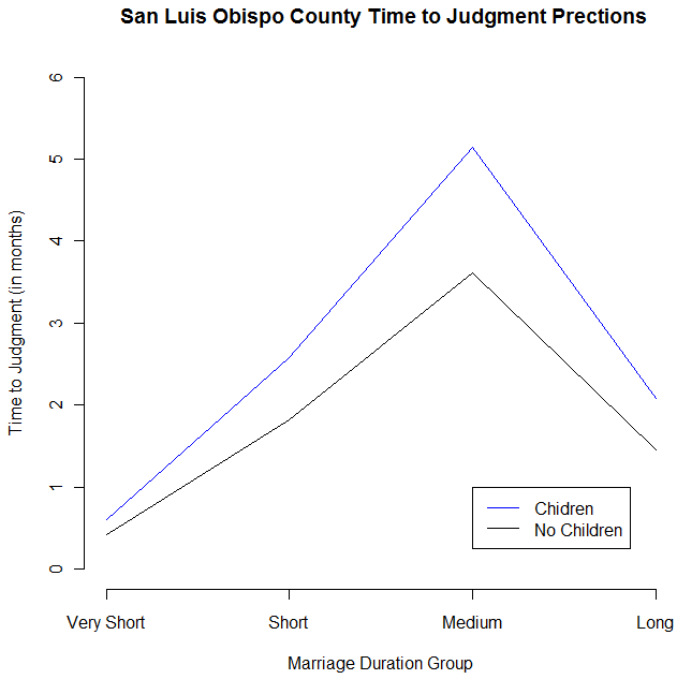

Figure 7: San Luis Obispo County Time to Judgment Predictions

Summary of Findings

Using survival analysis methods I am able to conclude that whether a couple has a child and the marriage duration category variables are at least moderately significant. There are graphical and tabular summaries throughout the previous sections in this project that further prove this. Our analysis also allowed us to estimate which regression models best predicts time to judgment, which is found in the middle of our Results section. In future studies, improvements that can be made on such studies is to take a random sample of all Americans. Since this data sets only consists of Santa Barbara County divorces, we can only draw inference on Santa Barbara county divorces. With a random sample, we can draw inference on the American population. Something else that could be further investigated is why there were missing entries for certain observations. It would be interesting if different results can be achieved with the completion of each missing observation.

Appendix

```
library(moments)
SantaBarbaraDivorce = read.csv("SantaBarbaraDivorce.csv", header = FALSE)
names(SantaBarbaraDivorce) = c("DOM","DOS","DOJ","Children")
SantaBarbaraDivorce$Children = as.character(SantaBarbaraDivorce$Children)
SantaBarbaraDivorce$DOJ = as.character(SantaBarbaraDivorce$DOJ)

### SantaBarbaraDivorce[SantaBarbaraDivorce[,3]=="",]
for (itor in 1:length(SantaBarbaraDivorce$DOJ))
{
	if (SantaBarbaraDivorce[itor, 3] =="" ) SantaBarbaraDivorce[itor, 3] = "06/11/2003"
	if (SantaBarbaraDivorce[itor, 4] =="No ") SantaBarbaraDivorce[itor, 4] = "No"
	if (SantaBarbaraDivorce[itor, 4] =="Yes ") SantaBarbaraDivorce[itor, 4] = "Yes"
}
SantaBarbaraDivorce$DOM = as.Date(SantaBarbaraDivorce$DOM, "%m/%d/%Y")
SantaBarbaraDivorce$DOS = as.Date(SantaBarbaraDivorce$DOS, "%m/%d/%Y")
SantaBarbaraDivorce$DOJ = as.Date(SantaBarbaraDivorce$DOJ, "%m/%d/%Y")
SantaBarbaraDivorce$Children = as.factor(SantaBarbaraDivorce$Children)

MarriageLength = ((SantaBarbaraDivorce$DOS-SantaBarbaraDivorce$DOM)/365)
SeperationLength = ((SantaBarbaraDivorce$DOJ-SantaBarbaraDivorce$DOS)/30)
SantaBarbaraDivorce = data.frame(cbind(SantaBarbaraDivorce, MarriageLength, SeperationLength))

#############################################################################
## Creating the censored variable
#############################################################################
CensorIndicator = NULL
CensorIndicator[1:287] = 1
CensorIndicator[SantaBarbaraDivorce$DOJ == "2003-06-11"] = 0
SantaBarbaraDivorce = data.frame(cbind(SantaBarbaraDivorce, CensorIndicator))
#############################################################################
## Categorical Marriage Length
#############################################################################
Length(Cesored$DurationCat)
N = nrow(SantaBarbaraDivorce)
DurationCat = NULL
for (DataItor in 1:N)
{
	if(SantaBarbaraDivorce$MarriageLength[DataItor] < 1)
	{
		DurationCat[DataItor] = 'VeryShort'
	}
	if((SantaBarbaraDivorce$MarriageLength[DataItor] >= 1) &
(SantaBarbaraDivorce$MarriageLength[DataItor] < 5))
	{
		DurationCat[DataItor] = 'Short1'
	}
```

```
        if((SantaBarbaraDivorce$MarriageLength[DataItor] >= 5) &
(SantaBarbaraDivorce$MarriageLength[DataItor] < 10))
        {
                DurationCat[DataItor] = 'Medium'
        }
        if(SantaBarbaraDivorce$MarriageLength[DataItor] >= 10)
        {
                DurationCat[DataItor] = 'Long'
        }
}
Duration = factor(DurationCat, levels = c('VeryShort', 'Short1', 'Medium', 'Long'))
DurationCat = factor(DurationCat, levels = c('VeryShort', 'Short1', 'Medium', 'Long'))
SantaBarbaraDivorce = data.frame(cbind(SantaBarbaraDivorce, DurationCat))


SantaBarbaraDivorce$MarriageLength = gsub("days", "", SantaBarbaraDivorce$MarriageLength)
SantaBarbaraDivorce$SeperationLength = gsub("days", "", SantaBarbaraDivorce$SeperationLength)

SantaBarbaraDivorce$MarriageLength = as.numeric(SantaBarbaraDivorce$MarriageLength)
SantaBarbaraDivorce$SeperationLength = as.numeric(SantaBarbaraDivorce$SeperationLength)
#############################################################################
## Seperation of the Censored and Uncensored Data
#############################################################################
CensorData = grep('0', SantaBarbaraDivorce$CensorIndicator)
Censored = SantaBarbaraDivorce[CensorData,]
nrow(Censored)
NonCensor = grep('1', SantaBarbaraDivorce$CensorIndicator)
NonCensored = SantaBarbaraDivorce[NonCensor,]
nrow(NonCensored)
#############################################################################
## multiple regression to predict T2J (Time to Judgement)
#############################################################################
## Regression Models for NonCensored data set
#############################################################################
par(mfrow=c(2,2))
NonCensoredLinearModel1 = lm((as.numeric(NonCensored$SeperationLength)) ~
as.factor(NonCensored$DurationCat))
summary(NonCensoredLinearModel1)
NonCensoredLinearModel1Resid = NonCensoredLinearModel1$resid
NonCensoredLinearModel1Fitted = NonCensoredLinearModel1$fitted
qqnorm(NonCensoredLinearModel1Resid, main = "Non-Censored Linear Model 1 NPP")
qqline(NonCensoredLinearModel1Resid)
plot(NonCensoredLinearModel1Fitted, NonCensoredLinearModel1Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Non-Censored Linear Model 1 Versus Fits")
hist(NonCensoredLinearModel1Resid, xlab = "Residuals", main = "Non-Censored Linear Model 1
        Histogram")
bartlett.test((as.numeric(NonCensored$SeperationLength)) ~ 0 + as.factor(NonCensored$DurationCat))
NonCensoredLinearModel1Shapiro = shapiro.test(NonCensoredLinearModel1Resid)
```

```
par(mfrow=c(2,2))
NonCensoredLinearModel2 = lm((as.numeric(NonCensored$SeperationLength)) ~
        as.factor(NonCensored$Children))
summary(NonCensoredLinearModel2)
NonCensoredLinearModel2Resid = NonCensoredLinearModel2$resid
NonCensoredLinearModel2Fitted = NonCensoredLinearModel2$fitted
qqnorm(NonCensoredLinearModel2Resid,  main = "Non-Censored Linear Model 2 NPP")
qqline(NonCensoredLinearModel2Resid)
plot(NonCensoredLinearModel2Fitted, NonCensoredLinearModel2Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Non-Censored Linear Model 2 Versus Fits")
hist(NonCensoredLinearModel2Resid, xlab = "Residuals", main = "Non-Censored Linear Model 2
        Histogram")
bartlett.test((as.numeric(NonCensored$SeperationLength)) ~ 0 + as.factor(NonCensored$Children))
NonCensoredLinearModel2Shapiro = shapiro.test(NonCensoredLinearModel2Resid)

par(mfrow=c(2,2))
NonCensoredLinearModel3 = lm((as.numeric(NonCensored$SeperationLength)) ~
        as.factor(NonCensored$Children)+ as.factor(NonCensored$DurationCat))
summary(NonCensoredLinearModel3)
NonCensoredLinearModel3Resid = NonCensoredLinearModel3$resid
NonCensoredLinearModel3Fitted = NonCensoredLinearModel3$fitted
qqnorm(NonCensoredLinearModel3Resid,  main = "Non-Censored Linear Model 3 NPP")
qqline(NonCensoredLinearModel3Resid)
plot(NonCensoredLinearModel3Fitted, NonCensoredLinearModel3Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Non-Censored Linear Model 3 Versus Fits")
hist(NonCensoredLinearModel3Resid, xlab = "Residuals", main = "Non-Censored Linear Model 3
        Histogram")
bartlett.test((as.numeric(NonCensored$SeperationLength)) ~ 0 + as.factor(NonCensored$Children)+
as.factor(NonCensored$DurationCat))
NonCensoredLinearModel3Shapiro = shapiro.test(NonCensoredLinearModel3Resid)

par(mfrow=c(2,2))
NonCensoredLinearModel4 = lm(((as.numeric(NonCensored$SeperationLength))) ~
as.factor(NonCensored$Children)*as.factor(NonCensored$DurationCat))
summary(NonCensoredLinearModel4)
NonCensoredLinearModel4Resid = NonCensoredLinearModel4$resid
NonCensoredLinearModel4Fitted = NonCensoredLinearModel4$fitted
qqnorm(NonCensoredLinearModel4Resid,  main = "Non-Censored Linear Model 4 NPP")
qqline(NonCensoredLinearModel4Resid)
plot(NonCensoredLinearModel4Fitted, NonCensoredLinearModel4Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Non-Censored Linear Model 4 Versus Fits")
hist(NonCensoredLinearModel4Resid, xlab = "Residuals", main = "Non-Censored Linear Model 4
        Histogram")
bartlett.test((as.numeric(NonCensored$SeperationLength)) ~ 0 +
as.factor(NonCensored$Children)*as.factor(NonCensored$DurationCat))
NonCensoredLinearModel4Shapiro = shapiro.test(NonCensoredLinearModel4Resid)
```

```
summary(NonCensoredLinearModel4)

###Partial F tests for the NonCensored Data
anova(NonCensoredLinearModel1, NonCensoredLinearModel3)
anova(NonCensoredLinearModel2, NonCensoredLinearModel3)
anova(NonCensoredLinearModel1, NonCensoredLinearModel4)
anova(NonCensoredLinearModel2, NonCensoredLinearModel4)

#############################################################################
## Regression Models for Censored data set
#############################################################################
par(mfrow=c(2,2))
CensoredLinearModel1 = lm(((as.numeric(Censored$SeperationLength)))
        ~as.factor(Censored$DurationCat))
summary(CensoredLinearModel1)
CensoredLinearModel1Resid = CensoredLinearModel1$resid
CensoredLinearModel1Fitted = CensoredLinearModel1$fitted
qqnorm(CensoredLinearModel1Resid , main = "Censored Linear Model 1 NPP")
qqline(CensoredLinearModel1Resid)
plot(CensoredLinearModel1Fitted, CensoredLinearModel1Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Censored Linear Model 1 Versus Fits")
hist(CensoredLinearModel1Resid, xlab = "Residuals", main = "Censored Linear Model 1 Histogram")
CensoredLinearModel1Bartlett =  bartlett.test(((as.numeric(Censored$SeperationLength))) ~ 0 +
        as.factor(Censored$DurationCat))
CensoredLinearModel1Shapiro = shapiro.test(CensoredLinearModel1Resid)

par(mfrow=c(2,2))
CensoredLinearModel2 = lm(((as.numeric(Censored$SeperationLength))) ~
        as.factor(Censored$Children))
summary(CensoredLinearModel2)
CensoredLinearModel2Resid = CensoredLinearModel2$resid
CensoredLinearModel2Fitted = CensoredLinearModel2$fitted
qqnorm(CensoredLinearModel2Resid , main = "Censored Linear Model 2 NPP")
qqline(CensoredLinearModel2Resid)
plot(CensoredLinearModel2Fitted, CensoredLinearModel2Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Censored Linear Model 2 Versus Fits")
hist(CensoredLinearModel2Resid, xlab = "Residuals", main = "Censored Linear Model 2 Histogram")
CensoredLinearModel2Bartlett = bartlett.test(((as.numeric(Censored$SeperationLength))) ~ 0 +
        as.factor(Censored$Children))
CensoredLinearModel2Shapiro = shapiro.test(CensoredLinearModel2Resid)

par(mfrow=c(2,2))
CensoredLinearModel3 = lm(((as.numeric(Censored$SeperationLength))) ~ \
        as.factor(Censored$Children)+ as.factor(Censored$DurationCat))
summary(CensoredLinearModel3)
CensoredLinearModel3Resid = CensoredLinearModel3$resid
CensoredLinearModel3Fitted = CensoredLinearModel3$fitted
qqnorm(CensoredLinearModel3Resid , main = "Censored Linear Model 3 NPP")
```

```
qqline(CensoredLinearModel3Resid)
plot(CensoredLinearModel3Fitted, CensoredLinearModel3Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Censored Linear Model 3 Versus Fits")
hist(CensoredLinearModel3Resid, xlab = "Residuals", main = "Censored Linear Model 3 Histogram")
CensoredLinearModel3Bartlett = bartlett.test(((as.numeric(Censored$SeperationLength))) ~ 0 +
        as.factor(Censored$Children)+ as.factor(Censored$DurationCat))
CensoredLinearModel3Shapiro = shapiro.test(CensoredLinearModel3Resid)

par(mfrow=c(2,2))
CensoredLinearModel4 = lm(((as.numeric(Censored$SeperationLength))) ~
        as.factor(Censored$Children)*as.factor(Censored$DurationCat))
summary(CensoredLinearModel4)
CensoredLinearModel4Resid = CensoredLinearModel4$resid
CensoredLinearModel4Fitted = CensoredLinearModel4$fitted
qqnorm(CensoredLinearModel4Resid , main = "Censored Linear Model 4 NPP")
qqline(CensoredLinearModel4Resid)
plot(CensoredLinearModel4Fitted, CensoredLinearModel4Resid, xlab = "Fitted Values", ylab =
        "Residuals", main = "Censored Linear Model 4 Versus Fits")
hist(CensoredLinearModel4Resid, xlab = "Residuals", main = "Censored Linear Model 4 Histogram")
CensoredLinearModel4Bartlett = bartlett.test(((as.numeric(Censored$SeperationLength))) ~ 0 +
        as.factor(Censored$Children)*as.factor(Censored$DurationCat))
CensoredLinearModel4Shapiro = shapiro.test(CensoredLinearModel4Resid)

###  Partial F tests for the Censored Data
anova(CensoredLinearModel1, CensoredLinearModel3)
anova(CensoredLinearModel2, CensoredLinearModel3)
anova(CensoredLinearModel1, CensoredLinearModel4)
anova(CensoredLinearModel2, CensoredLinearModel4)

#############################################################################
## Multiple Regression of the whole data set
#############################################################################

par(mfrow=c(2,2))
SantaBarbaraDivorceLinearModel1 = lm((as.numeric(SantaBarbaraDivorce$SeperationLength)) ~
        (as.factor(SantaBarbaraDivorce$DurationCat)))
summary(SantaBarbaraDivorceLinearModel1)
SantaBarbaraDivorceLinearModel1Resid = SantaBarbaraDivorceLinearModel1$resid
SantaBarbaraDivorceLinearModel1Fitted = SantaBarbaraDivorceLinearModel1$fitted
qqnorm(SantaBarbaraDivorceLinearModel1Resid , main = "Complete Linear Model 1 NPP")
qqline(SantaBarbaraDivorceLinearModel1Resid )
plot(SantaBarbaraDivorceLinearModel1Fitted , SantaBarbaraDivorceLinearModel1Resid , xlab = "Fitted
        Values", ylab = "Residuals", main = "Complete Linear Model 1 Versus Fits")
hist(SantaBarbaraDivorceLinearModel1Resid , xlab = "Residuals", main = "Complete Linear Model 1
Histogram")
SantaBarbaraDivorceLinearModel1Bartlett  =
        bartlett.test(((as.numeric(SantaBarbaraDivorce$SeperationLength))) ~ 0 +
        as.factor(SantaBarbaraDivorce$DurationCat))
```

```
SantaBarbaraDivorceLinearModel1Shapiro = shapiro.test(SantaBarbaraDivorceLinearModel1Resid)


par(mfrow=c(2,2))
SantaBarbaraDivorceLinearModel2 = lm(((as.numeric(SantaBarbaraDivorce$SeperationLength))) ~
        as.factor(SantaBarbaraDivorce$Children))
summary(SantaBarbaraDivorceLinearModel2)
SantaBarbaraDivorceLinearModel2Resid = SantaBarbaraDivorceLinearModel2$resid
SantaBarbaraDivorceLinearModel2Fitted = SantaBarbaraDivorceLinearModel2$fitted
qqnorm(SantaBarbaraDivorceLinearModel2Resid , main = "Complete Linear Model 2 NPP")
qqline(SantaBarbaraDivorceLinearModel2Resid )
plot(SantaBarbaraDivorceLinearModel2Fitted , SantaBarbaraDivorceLinearModel2Resid , xlab = "Fitted
        Values", ylab = "Residuals", main = "Complete Linear Model 2 Versus Fits")
hist(SantaBarbaraDivorceLinearModel2Resid , xlab = "Residuals", main = "Complete Linear Model 2
        Histogram")
SantaBarbaraDivorceLinearModel2Bartlett  =
        bartlett.test(((as.numeric(SantaBarbaraDivorce$SeperationLength))) ~ 0 +
        as.factor(SantaBarbaraDivorce$Children))
SantaBarbaraDivorceLinearModel2Shapiro = shapiro.test(SantaBarbaraDivorceLinearModel2Resid)

par(mfrow=c(2,2))
SantaBarbaraDivorceLinearModel3 = lm((as.numeric(SantaBarbaraDivorce$SeperationLength)) ~
        as.factor(SantaBarbaraDivorce$Censor))
summary(SantaBarbaraDivorceLinearModel3)
SantaBarbaraDivorceLinearModel3Resid = SantaBarbaraDivorceLinearModel3$resid
SantaBarbaraDivorceLinearModel3Fitted = SantaBarbaraDivorceLinearModel3$fitted
qqnorm(SantaBarbaraDivorceLinearModel3Resid , main = "Complete Linear Model 3 NPP")
qqline(SantaBarbaraDivorceLinearModel3Resid )
plot(SantaBarbaraDivorceLinearModel3Fitted , SantaBarbaraDivorceLinearModel3Resid , xlab = "Fitted
        Values", ylab = "Residuals", main = "Complete Linear Model 3 Versus Fits")
hist(SantaBarbaraDivorceLinearModel3Resid , xlab = "Residuals", main = "Complete Linear Model 3
        Histogram")
SantaBarbaraDivorceLinearModel3Bartlett  =
        bartlett.test((as.numeric(SantaBarbaraDivorce$SeperationLength)) ~ 0 +
        as.factor(SantaBarbaraDivorce$Censor))
SantaBarbaraDivorceLinearModel3Shapiro = shapiro.test(SantaBarbaraDivorceLinearModel3Resid)


par(mfrow=c(2,2))
SantaBarbaraDivorceLinearModel4 = lm((as.numeric(SantaBarbaraDivorce$SeperationLength)) ~
        as.factor(SantaBarbaraDivorce$DurationCat) + as.factor(SantaBarbaraDivorce$Children))
summary(SantaBarbaraDivorceLinearModel4)
SantaBarbaraDivorceLinearModel4Resid = SantaBarbaraDivorceLinearModel4$resid
SantaBarbaraDivorceLinearModel4Fitted = SantaBarbaraDivorceLinearModel4$fitted
qqnorm(SantaBarbaraDivorceLinearModel4Resid , main = "Complete Linear Model 4 NPP")
qqline(SantaBarbaraDivorceLinearModel4Resid )
plot(SantaBarbaraDivorceLinearModel4Fitted , SantaBarbaraDivorceLinearModel4Resid , xlab = "Fitted
        Values", ylab = "Residuals", main = "Complete Linear Model 4 Versus Fits")
```

```
hist(SantaBarbaraDivorceLinearModel4Resid , xlab = "Residuals", main = "Complete Linear Model 4
        Histogram")
SantaBarbaraDivorceLinearModel4Bartlett  =
        bartlett.test(((as.numeric(SantaBarbaraDivorce$SeperationLength))) ~ 0 +
        as.factor(SantaBarbaraDivorce$DurationCat) + as.factor(SantaBarbaraDivorce$Children))
SantaBarbaraDivorceLinearModel4Shapiro = shapiro.test(SantaBarbaraDivorceLinearModel4Resid)


SantaBarbaraDivorceLinearModel4 = lm((as.numeric(SantaBarbaraDivorce$SeperationLength)) ~
        as.factor(SantaBarbaraDivorce$DurationCat) + as.factor(SantaBarbaraDivorce$Children))
summary(SantaBarbaraDivorceLinearModel4)
#####################################################################
### Building the 4 distinct subgroups (very short, short, medium, long) within our 3 data sets.
#############################################################################################
SantaBarbaraDivorce$SeperationLength = as.numeric(SantaBarbaraDivorce$SeperationLength)

CompleteVeryShort = grep('VeryShort', SantaBarbaraDivorce$DurationCat)
CompleteVeryShort = SantaBarbaraDivorce[CompleteVeryShort,]
CompleteMeanVeryShort = mean(CompleteVeryShort$SeperationLength)
CompleteShort = grep("Short1", SantaBarbaraDivorce$DurationCat)
CompleteShort = SantaBarbaraDivorce[CompleteShort,]
CompleteMeanShort = mean(CompleteShort$SeperationLength)
CompleteMedium = grep('Medium', SantaBarbaraDivorce$DurationCat)
CompleteMedium = SantaBarbaraDivorce[CompleteMedium,]
CompleteMeanMedium = mean(CompleteMedium$SeperationLength)
CompleteLong = grep('Long', SantaBarbaraDivorce$DurationCat)
CompleteLong = SantaBarbaraDivorce[CompleteLong,]
CompleteMeanLong = mean(CompleteLong$SeperationLength)

CensoredVeryShort = grep('VeryShort', Censored$DurationCat)
CensoredVeryShort = Censored[CensoredVeryShort,]
CensoredMeanVeryShort = mean(as.numeric(CensoredVeryShort$SeperationLength))
CensoredShort = grep("Short1", Censored$DurationCat)
CensoredShort = Censored[CensoredShort,]
CensoredMeanShort = mean(as.numeric(CensoredShort$SeperationLength))
CensoredMedium = grep('Medium', Censored$DurationCat)
CensoredMedium = Censored[CensoredMedium,]
CensoredMeanMedium = mean(as.numeric(CensoredMedium$SeperationLength))
CensoredLong = grep('Long', Censored$DurationCat)
CensoredLong = Censored[CensoredLong,]
CensoredMeanLong = mean(as.numeric(CensoredLong$SeperationLength))

NonCensoredVeryShort = grep('VeryShort', NonCensored$DurationCat)
NonCensoredVeryShort = NonCensored[NonCensoredVeryShort,]
NonCensoredMeanVeryShort = mean(as.numeric(NonCensoredVeryShort$SeperationLength))
NonCensoredShort = grep("Short1", NonCensored$DurationCat)
NonCensoredShort = NonCensored[NonCensoredShort,]
NonCensoredMeanShort = mean(as.numeric(NonCensoredShort$SeperationLength))
```

```
NonCensoredMedium = grep('Medium', NonCensored$DurationCat)
NonCensoredMedium = NonCensored[NonCensoredMedium,]
NonCensoredMeanMedium = mean(as.numeric(NonCensoredMedium$SeperationLength))
NonCensoredLong = grep('Long', NonCensored$DurationCat)
NonCensoredLong = NonCensored[NonCensoredLong,]
NonCensoredMeanLong = mean(as.numeric(NonCensoredLong$SeperationLength))

#########################################################################
##### Confidence Intervals for each of the Three Seperated Groups
#########################################################################

CICompleteVSLower  = (t.test(CompleteVeryShort$SeperationLength))$conf.int[1]
CICompleteVSUpper = (t.test(CompleteVeryShort$SeperationLength))$conf.int[2]
CICompleteVS = cbind(CICompleteVSLower , CICompleteVSUpper)
CICompleteSLower  = (t.test(CompleteShort$SeperationLength))$conf.int[1]
CICompleteSUpper = (t.test(CompleteShort$SeperationLength))$conf.int[2]
CICompleteS = cbind(CICompleteSLower , CICompleteSUpper)
CICompleteMLower  = (t.test(CompleteMedium$SeperationLength))$conf.int[1]
CICompleteMUpper = (t.test(CompleteMedium$SeperationLength))$conf.int[2]
CICompleteM = cbind(CICompleteMLower , CICompleteMUpper)
CICompleteLLower  = (t.test(CompleteLong$SeperationLength))$conf.int[1]
CICompleteLUpper = (t.test(CompleteLong$SeperationLength))$conf.int[2]
CICompleteL = cbind(CICompleteLLower , CICompleteLUpper)
CIComplete = rbind(CICompleteVS, CICompleteS, CICompleteM, CICompleteL)
CICompleteUpper = CIComplete[,1]
CICompleteLower = CIComplete[,2]

CICensoredVSLower  = (t.test(as.numeric(CensoredVeryShort$SeperationLength)))$conf.int[1]
CICensoredVSUpper  = (t.test(as.numeric(CensoredVeryShort$SeperationLength)))$conf.int[2]
CICensoredVS = cbind(CICensoredVSLower, CICensoredVSUpper)
CICensoredSLower  = (t.test(as.numeric(CensoredShort$SeperationLength)))$conf.int[1]
CICensoredSUpper  = (t.test(as.numeric(CensoredShort$SeperationLength)))$conf.int[2]
CICensoredS = cbind(CICensoredSLower, CICensoredSUpper)
CICensoredMLower  = (t.test(as.numeric(CensoredMedium$SeperationLength)))$conf.int[1]
CICensoredMUpper  = (t.test(as.numeric(CensoredMedium$SeperationLength)))$conf.int[2]
CICensoredM = cbind(CICensoredMLower, CICensoredMUpper)
CICensoredLLower  = (t.test(as.numeric(CensoredLong$SeperationLength)))$conf.int[1]
CICensoredLUpper  = (t.test(as.numeric(CensoredLong$SeperationLength)))$conf.int[2]
CICensoredL = cbind(CICensoredLLower, CICensoredLUpper)
CICensored = rbind(CICensoredVS, CICensoredS, CICensoredM, CICensoredL)
CICensoredUpper = CICensored[,1]
CICensoredLower = CICensored[,2]

CINonCensoredVSLower  = (t.test(as.numeric(NonCensoredVeryShort$SeperationLength)))$conf.int[1]
CINonCensoredVSUpper  = (t.test(as.numeric(NonCensoredVeryShort$SeperationLength)))$conf.int[2]
CINonCensoredVS = cbind(CINonCensoredVSLower, CINonCensoredVSUpper)
CINonCensoredSLower  = (t.test(as.numeric(NonCensoredShort$SeperationLength)))$conf.int[1]
CINonCensoredSUpper  = (t.test(as.numeric(NonCensoredShort$SeperationLength)))$conf.int[2]
```

```
CINonCensoredS = cbind(CINonCensoredSLower, CINonCensoredSUpper)
CINonCensoredMLower  = (t.test(as.numeric(NonCensoredMedium$SeperationLength)))$conf.int[1]
CINonCensoredMUpper  = (t.test(as.numeric(NonCensoredMedium$SeperationLength)))$conf.int[2]
CINonCensoredM = cbind(CINonCensoredMLower, CINonCensoredMUpper)
CINonCensoredLLower  = (t.test(as.numeric(NonCensoredLong$SeperationLength)))$conf.int[1]
CINonCensoredLUpper  = (t.test(as.numeric(NonCensoredLong$SeperationLength)))$conf.int[2]
CINonCensoredL = cbind(CINonCensoredLLower, CINonCensoredLUpper)
CINonCensored = rbind(CINonCensoredVS, CINonCensoredS, CINonCensoredM, CINonCensoredL)
CINonCensoredUpper = CINonCensored[,1]
CINonCensoredLower = CINonCensored[,2]

#########################################################################
### Kurtosis
#########################################################################

KurtosisCompleteVS = kurtosis(CompleteVeryShort$SeperationLength)
KurtosisCompleteS = kurtosis(CompleteShort$SeperationLength)
KurtosisCompleteM = kurtosis(CompleteMedium$SeperationLength)
KurtosisCompleteL = kurtosis(CompleteLong$SeperationLength)
KurtosisComplete = rbind(KurtosisCompleteVS, KurtosisCompleteS, KurtosisCompleteM,
KurtosisCompleteL)

par(mfrow=c(2,1))
hist(CompleteLong$SeperationLength, xlim = c(0,500), breaks = 50, main = "Complete/Long", xlab =
"Time to Judgement (months)", col = "magenta")
hist(CompleteShort$SeperationLength, xlim = c(0,500),  breaks = 15, main = "Complete/Very Short", xlab
= "Time to Judgement (months)", col = "turquoise")
par(mfrow=c(1,1))

KurtosisCensoredVS = kurtosis(as.numeric(CensoredVeryShort$SeperationLength))
KurtosisCensoredS = kurtosis(as.numeric(CensoredShort$SeperationLength))
KurtosisCensoredM = kurtosis(as.numeric(CensoredMedium$SeperationLength))
KurtosisCensoredL = kurtosis(as.numeric(CensoredLong$SeperationLength))
KurtosisCensored = rbind(KurtosisCensoredVS, KurtosisCensoredS, KurtosisCensoredM,
KurtosisCensoredL)

KurtosisNonCensoredVS = kurtosis(as.numeric(NonCensoredVeryShort$SeperationLength))
KurtosisNonCensoredS = kurtosis(as.numeric(NonCensoredShort$SeperationLength))
KurtosisNonCensoredM = kurtosis(as.numeric(NonCensoredMedium$SeperationLength))
KurtosisNonCensoredL = kurtosis(as.numeric(NonCensoredLong$SeperationLength))
KurtosisNonCensored = rbind(KurtosisNonCensoredVS, KurtosisNonCensoredS, KurtosisNonCensoredM,
KurtosisNonCensoredL)

par(mfrow=c(2,1))
hist(CompleteLong$SeperationLength, xlim = c(0,500), breaks = 50, main = "Complete/Long", xlab =
"Time to Judgement (months)", col = "magenta")
hist(CompleteShort$SeperationLength, xlim = c(0,500),  breaks = 15, main = "Complete/Short", xlab =
"Time to Judgement (months)", col = "turquoise")
```

```
par(mfrow=c(1,1))

CompleteLongSD = sd(CompleteLong$SeperationLength)
Complete81Omit = c(CompleteLong$SeperationLength[1:80], CompleteLong$SeperationLength[82:92])
Complete81OmitSD = sd(Complete81Omit)
Complete81OmitLower = t.test(Complete81Omit)$conf.int[1]
Complete81OmitUpper = t.test(Complete81Omit)$conf.int[2]
Complete81OmitCI = cbind(Complete81OmitLower, Complete81OmitUpper)
Complete81OmitKurtosis = kurtosis(Complete81Omit)

#########################################################################
### Plots
#########################################################################
plot.default(SantaBarbaraDivorce$SeperationLength ~ as.factor(SantaBarbaraDivorce$DurationCat),
        main = "Full Data Set",xlab = "Marriage Duration", ylab = "Time to Judgement", ylim = c(0,500),
        axes = FALSE)
axis(1, at=1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at =c(0,100,200,300,400, 500), lab=c(0,100,200,300,400,500))
CompleteMeans = rbind(CompleteMeanVeryShort, CompleteMeanShort, CompleteMeanMedium,
        CompleteMeanLong)
lines(CompleteMeans, col = "red", type = "l")
plot(CompleteMeans, type = 'l', col = "green", main = "Complete (n=287)", ylab="Time to Judgement
(months)", ylim = c(0,170),xlab = "Marriage Duration", axes = FALSE)
lines(CICompleteUpper, lty = 2, col = "red")
lines(CICompleteLower, lty = 2, col = "red")
axis(1, at = 1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at = c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170), lab =
        c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170))
CompleteMean = mean(CompleteMeans)
abline(h=CompleteMean, col = 'black', lty = 3)
legend(x=2.5, y=22,legend = c("Point Estimate", "Mean", "95% Confidence Interval"),lty = c(1,3,2),col =
        c("green","black","red"))

plot.default(Censored$SeperationLength ~ as.factor(Censored$DurationCat), main = "Censored Data
Set" ,xlab = "Marriage Duration", ylab = "Time to Judgement", ylim = c(0,500), axes = FALSE)
axis(1, at=1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at =c(0,100,200,300,400, 500), lab=c(0,100,200,300,400,500))
CensoredMeans = rbind(CensoredMeanVeryShort, CensoredMeanShort, CensoredMeanMedium,
        CensoredMeanLong)
lines(CensoredMeans, col = "red", type = "l")
plot(CensoredMeans, col = "green", type = 'l', main = "Censored (n=48)", ylab="Time to Judgement
        (months)", xlab = "Marriage Duration", ylim = c(0,170), axes = FALSE)
lines(CICensoredUpper, lty = 2, col = "red")
lines(CICensoredLower, lty = 2, col = "red")
axis(1, at = 1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at = c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170), lab =
        c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170))
CensoredMean = mean(CensoredMeans)
```

```
abline(h=CensoredMean, col = "black", lty = 3)
legend(x=2.5, y=22,legend = c("Point Estimate", "Mean", "95% Confidence Interval"),lty = c(1,3,2),col =
        c("green","black","red"))

str(NonCensored)
table(NonCensored$DurationCat)
plot.default(NonCensored$SeperationLength ~ as.factor(NonCensored$DurationCat), main =
        "NonCensored Data Set" ,xlab = "Marriage Duration", ylab = "Time to Judgement", ylim = c(0,
        500), axes = FALSE)
axis(1, at=1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at =c(0,100,200,300,400, 500), lab=c(0,100,200,300,400,500))
NonCensoredMeans = rbind(NonCensoredMeanVeryShort, NonCensoredMeanShort,
        NonCensoredMeanMedium, NonCensoredMeanLong)
lines(NonCensoredMeans, col = "red", type = "l")
plot(NonCensoredMeans, col = "green", type = 'l', main = "Non Censored (n=245)", ylab="Time to
        Judgement (months)", xlab = "Marriage Duration", ylim = c(0,170), axes = FALSE)
lines(CINonCensoredUpper, lty = 2, col = "red")
lines(CINonCensoredLower, lty = 2, col = "red")
axis(1, at = 1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at = c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170), lab =
        c(0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160, 170))
NonCensoredMean = mean(NonCensoredMeans)
abline(h=NonCensoredMean, col = "black", lty =3)
legend(x=2.5, y=22,legend = c("Point Estimate", "Mean", "95% Confidence Interval"),lty = c(1,3,2),col =
        c("green","black","red"))

#########################################################################
##### Comparison of the Complete, Censored, and NonCensored data sets
#########################################################################
plot(CompleteMeans, col = "red", type = 'l', ylim = c(0, 120), ylab= "Time to Judgement (months)", xlab =
        "Marriage Duration", main = "Means Seperated Into Three Groups", axes = FALSE)
lines(CensoredMeans, col = "blue", type = 'l')
lines(NonCensoredMeans, col = "green", type = 'l')
axis(1, at = 1:4, lab=c("Very Short", "Short", "Medium", "Long"))
axis(2, at = c(0,10,20,30,40,50,60,70,80,90,100,110,120), lab =
        c(0,10,20,30,40,50,60,70,80,90,100,110,120))
legend(x=3.0,y=20,legend=c("Complete","Censored", "NonCensored"),lty=c(1,1,1),col=c("red", "blue",
        "green"))

#########################################################################
### Cross-Validation
#########################################################################

NonCensored$SeperationLength = as.numeric(NonCensored$SeperationLength)
DataSet = SantaBarbaraDivorce
N = nrow(DataSet)
PredictedSepLength = NULL
for (i in 1:N)
```

```
{
        TestData = DataSet[i,]
        TrainData = DataSet[-i,]
        ModelTemp = lm(as.numeric(DataSet$SeperationLength) ~as.factor(DataSet$DurationCat) *
as.factor(DataSet$Children))
        PredictionTemp = predict.lm(ModelTemp, newdata = TestData)
        PredictedSepLength[i] = PredictionTemp
}
CVE = sum((DataSet$SeperationLength-PredictedSepLength)^2)
CVE

#####################################################################
## Using the survreg function for the
#####################################################################
library(survival)

SurvRegModel1 = survreg(Surv(SeperationLength, CensorIndicator)~ Children, dist = "lognormal", data =
        SantaBarbaraDivorce)
summary(SurvRegModel1)
SurvRegModel2 = survreg(Surv(SeperationLength, CensorIndicator)~ DurationCat, dist = "lognormal",
        data = SantaBarbaraDivorce)
summary(SurvRegModel2)
SurvRegModel3 = survreg(Surv(SeperationLength, CensorIndicator)~ Children + DurationCat, dist =
        "lognormal",data = SantaBarbaraDivorce)
summary(SurvRegModel3)
SurvRegModel4 = survreg(Surv(SeperationLength, CensorIndicator)~ Children*DurationCat, dist =
        "lognormal",data = SantaBarbaraDivorce)
summary(SurvRegModel4)

SLOData = read.csv("SLOData.csv", header = TRUE)
SLOSurvivalRegression1 = survreg(Surv(LOS,Censor) ~ Kids, dist = "lognormal", data = SLOData)
summary(SLOSurvivalRegression1)
SLOSurvivalRegression2= survreg(Surv(LOS,Censor) ~ SLOData$LOMInt, dist = "lognormal", data =
SLOData)
summary(SLOSurvivalRegression2)
SLOSurvivalRegression3= survreg(Surv(LOS,Censor) ~ Kids + LOMInt, dist = "lognormal", data = SLOData)
summary(SLOSurvivalRegression3)
SLOSurvivalRegression4= survreg(Surv(LOS,Censor) ~ Kids * LOMInt, dist = "lognormal", data = SLOData)
summary(SLOSurvivalRegression4)

DurationCat1 = c('VeryShort', 'Short1', 'Medium', 'Long')
DurationCat2 = c('VeryShort', 'Short1', 'Medium', 'Long')
Children1 = c("No", "No", "No", "No")
Children2 = c("Yes", "Yes", "Yes", "Yes")
explanatory1 = data.frame(DurationCat = DurationCat1, Children = Children1)
explanatory2 = data.frame(DurationCat = DurationCat2, Children = Children2)
SurvRegPredictions1 = predict(SurvRegModel3, newdata = data.frame(explanatory1), type = 'response')
SurvRegPredictions2 = predict(SurvRegModel3, newdata = data.frame(explanatory2), type = 'response')
```

```
plot(SurvRegPredictions1, ylim  = c(0,60), axes = FALSE, xlab = "Marriage Duration Group", ylab = "Time
        to Judgment (in months)", main = "Santa Barbara County Time to Judgment Predictions",type  =
        'l', col = "black")
lines(SurvRegPredictions2, pch = 19, col = "Blue")
axis(1, at = 1:4, lab = c("Very Short", "Short", "Medium", "Long"))
axis(2, at = c(0,10,20,30,40,50,60), lab = c(0,10,20,30,40,50,60))
legend(x = 3, y=10, legend = c("Chidren", "No Children"), lty = c(1,1), col = c("blue", "black"))

Duration3 = c('avery_short','short','medium','long')
Duration4 = c('avery_short','short','medium','long')
Children3 = c(1,1,1,1)
Children4 = c(0,0,0,0)
predictors1 = data.frame(LOMInt = Duration3, Kids = Children3)
predictors2 = data.frame(LOMInt = Duration4, Kids = Children4)
SloSurvPreds1 = predict(SLOSurvivalRegression3, newdata = data.frame(predictors1), type = 'response')
SloSurvPreds2 = predict(SLOSurvivalRegression3, newdata = data.frame(predictors2), type = 'response')

plot(SloSurvPreds1, ylim =c(0,6), axes = FALSE, xlab = "Marriage Duration Group", ylab = "Time to
        Judgment (in months)", main = " San Luis Obispo County Time to Judgment Prections", type = 'l',
        col = "black")
lines(SloSurvPreds2, type = 'l', col = "blue")
axis(1, at = 1:4, lab = c("Very Short", "Short", "Medium", "Long"))
axis(2, at = 0:6, lab = 0:6)
legend(x = 3, y=1, legend = c("Chidren", "No Children"), lty = c(1,1), col = c("blue", "black"))
```