Average Age of Death between Marital Status (For Year 2015)

Steven Seeger, Seth Pixton

12/20/2018

Introduction

The question we decided to research is this: Does the marital status of an individual in the United States affect the average age of death? The marital statuses we will be looking at are Divorced and Married. The value of this question is that if we can determine that marital status has an effect on average age of death, it could influence the insurance industry and encourage further research as to why this is the case.

All four groups' EDA are included for the purpose of viewing and later interest. We will not be exploring this further than that.

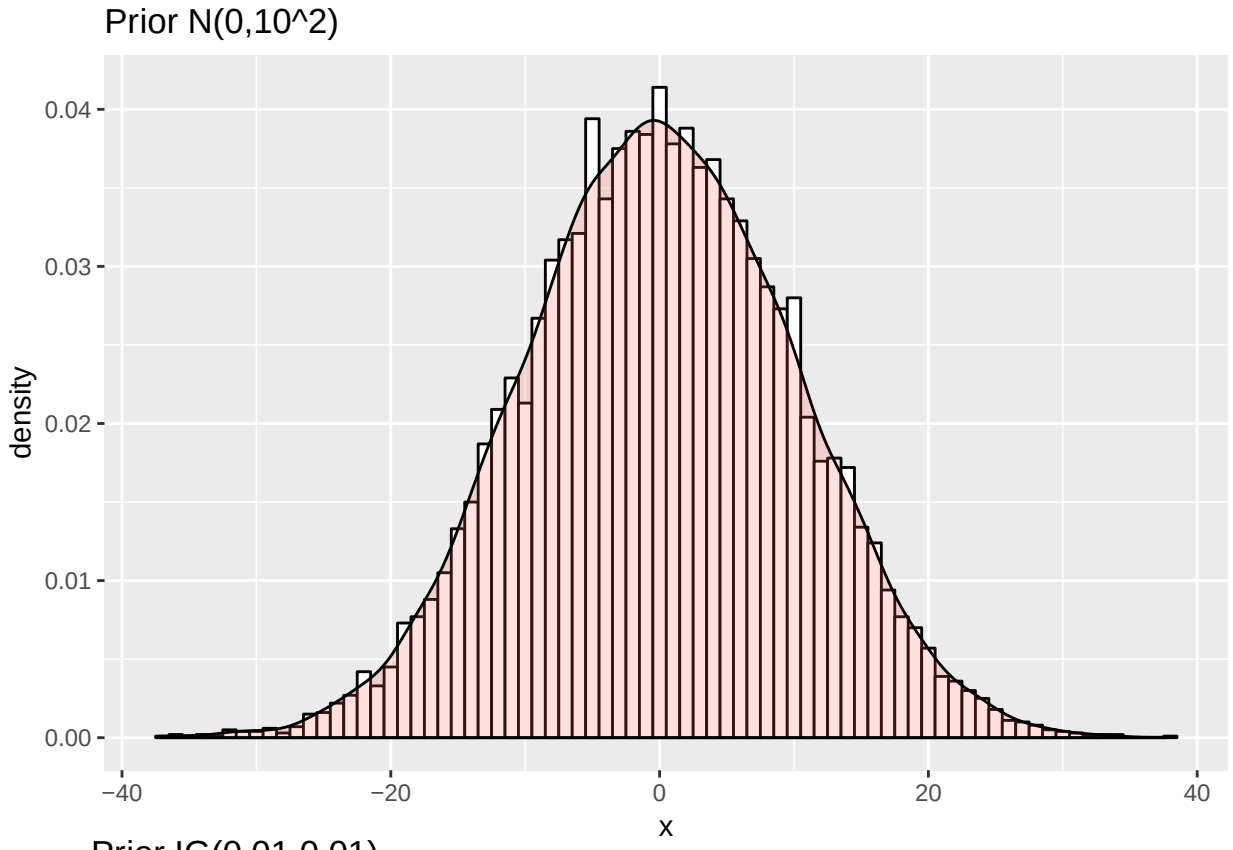
Literature Review

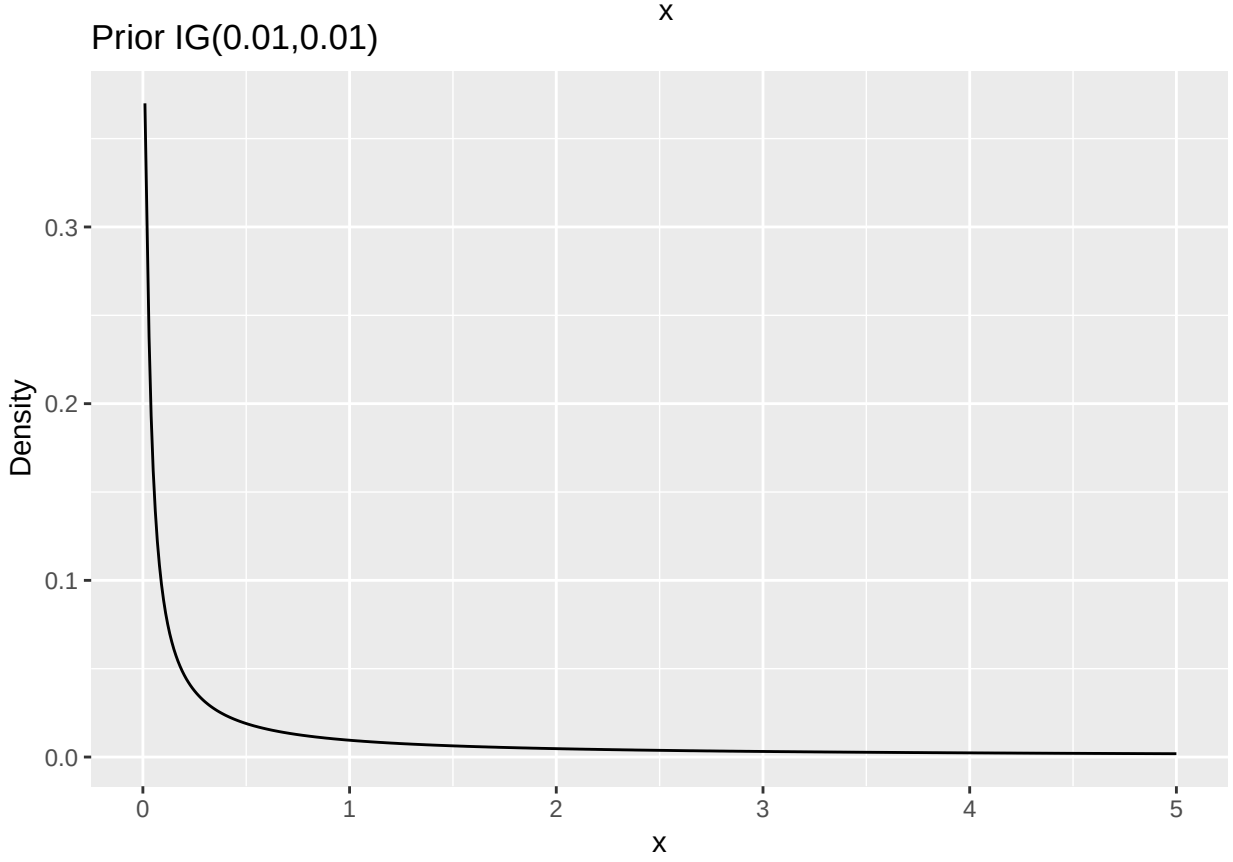
A study was conducted by Dustin C. Brown about Life Expectancy Differentials by Marital Status, Individuals' Own Education, and Spousal Education in the United States. (http://paa2014.princeton.edu/papers/142823) This study was interesting, but did not exactly match the study that we are conducting. He used life expectancy as his research and we are using average age of death. His conclusion was that there is a difference between many groups for life expectancy at age 55. We are looking at all people that are married at any age and all that are married and then divorced.

Methods

We gathered our data from Kaggle.com. (Cite: https://www.kaggle.com/cdc/mortality/home) From the documentation for this data set, it states that "[a]ll data comes from the CDC's National Vital Statistics Systems, with the exception of the Icd10Code, which are sourced from the World Health Organization." We are curious about the mean age of death for people who are married versusthat of people who are divorced. We choose to go with a Normal-Normal-Inverse-Gamma prior distribution. We chose this distribution because we decided that deaths will probably be normally spread out around the center of the data. We went with an uninformative prior of mu ~ N(0, 10^2) and sigma^2 ~ IG(0.01, 0.01). We felt that this was appropriate given the vast amount of data we had.

The following two graphs show the Prior Distributions we decided to use.





The posterior distribtuions will answer our research question by allowing us to see if our data is different at a

significant level. We can use the difference in means in the posterior distributions to detect if there is a true difference between the age of death between the two groups.

Exploratory Data Analysis

Included in the following section is a summary of the data, as well as density distributions of the data. The first chart that will be shown is a boxplot. We are able to include a boxplot in this report because the extreme outliers were removed, and the chart looks normal and is easier to understand. All four groups summary statistics were included, but because married and divorced were the most normally distributed, and they were the most similar to each other in a sensical way (ie married and died later) we only included analysis on those two groups.

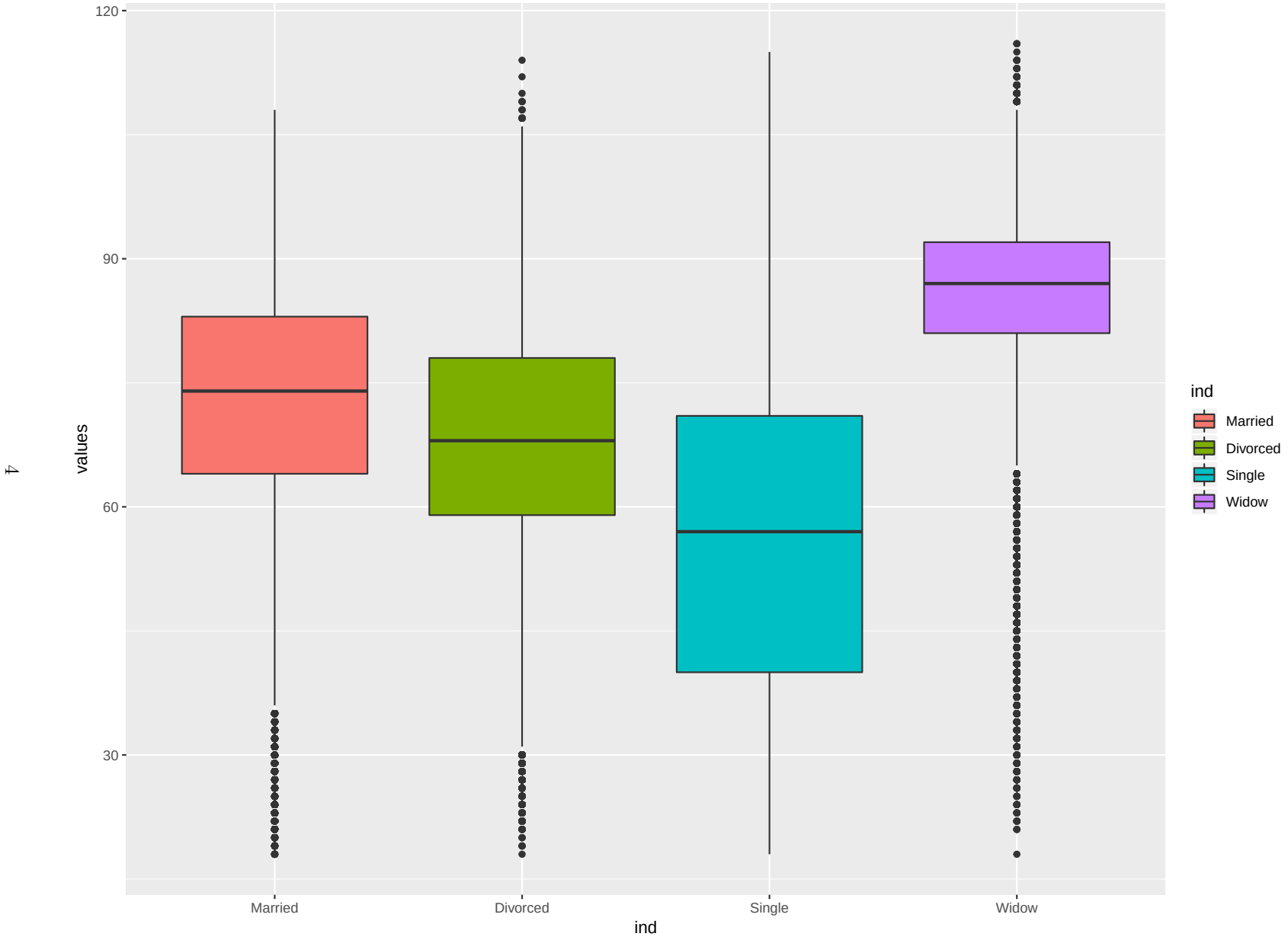


Figure 1: Boxplot

Shown in the table below, are the summary statistics for each of the groups we were testing.

Table 1: Summary Statistics (age of death)

	mean	sd	median	min	max	n
Married	72.13459	13.741310	74	18	108	1002469
Divorced	68.25280	13.685330	68	18	114	422284
Single	56.09009	20.718660	57	18	115	316921
Widow	85.35779	9.619899	87	18	116	921989

The mean, median and standard deviation for the groups are shown. The mean and the median for each group are almost identical.

Shown on the next page are several density distribution charts. This helped us determine which of the two groups to compare.

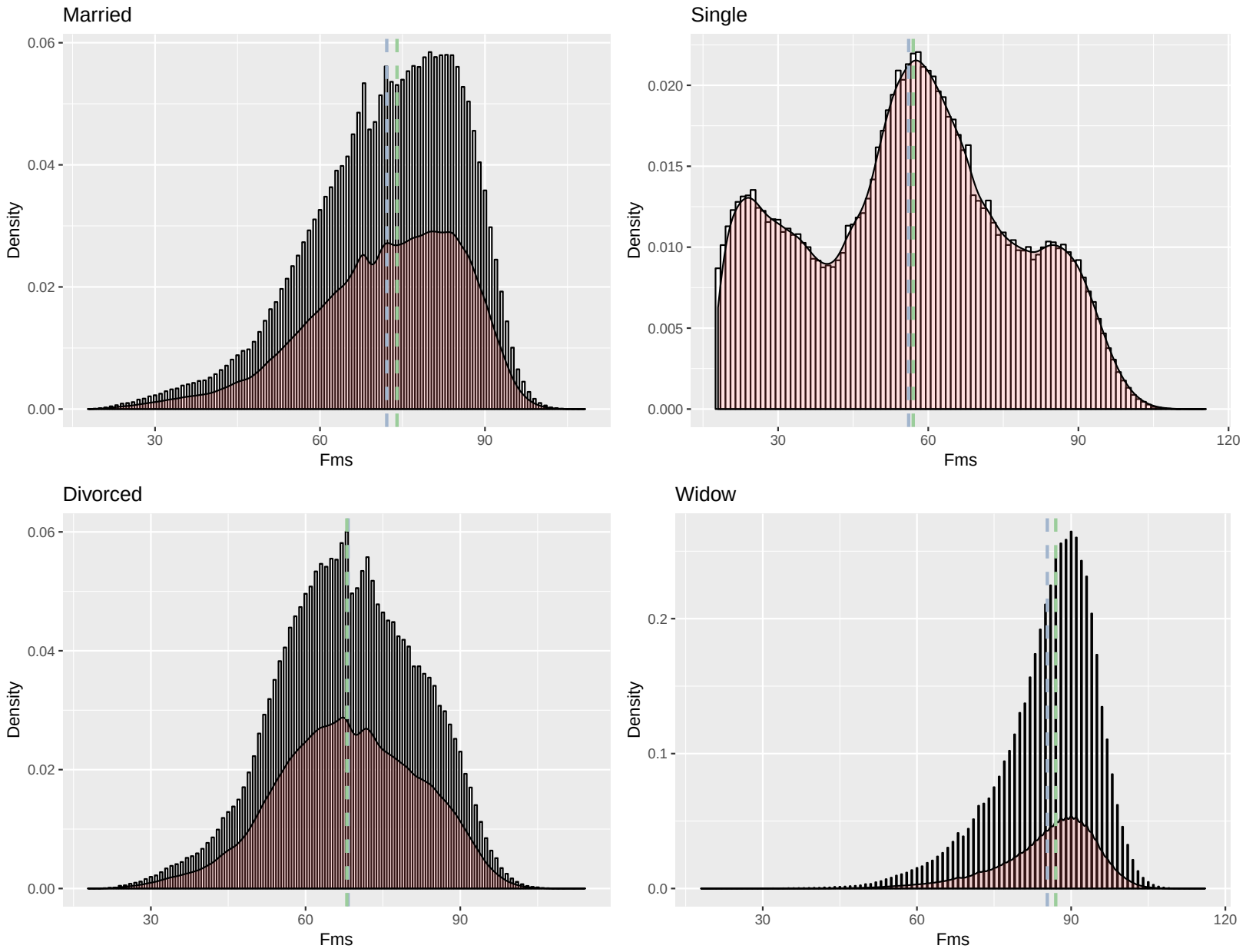
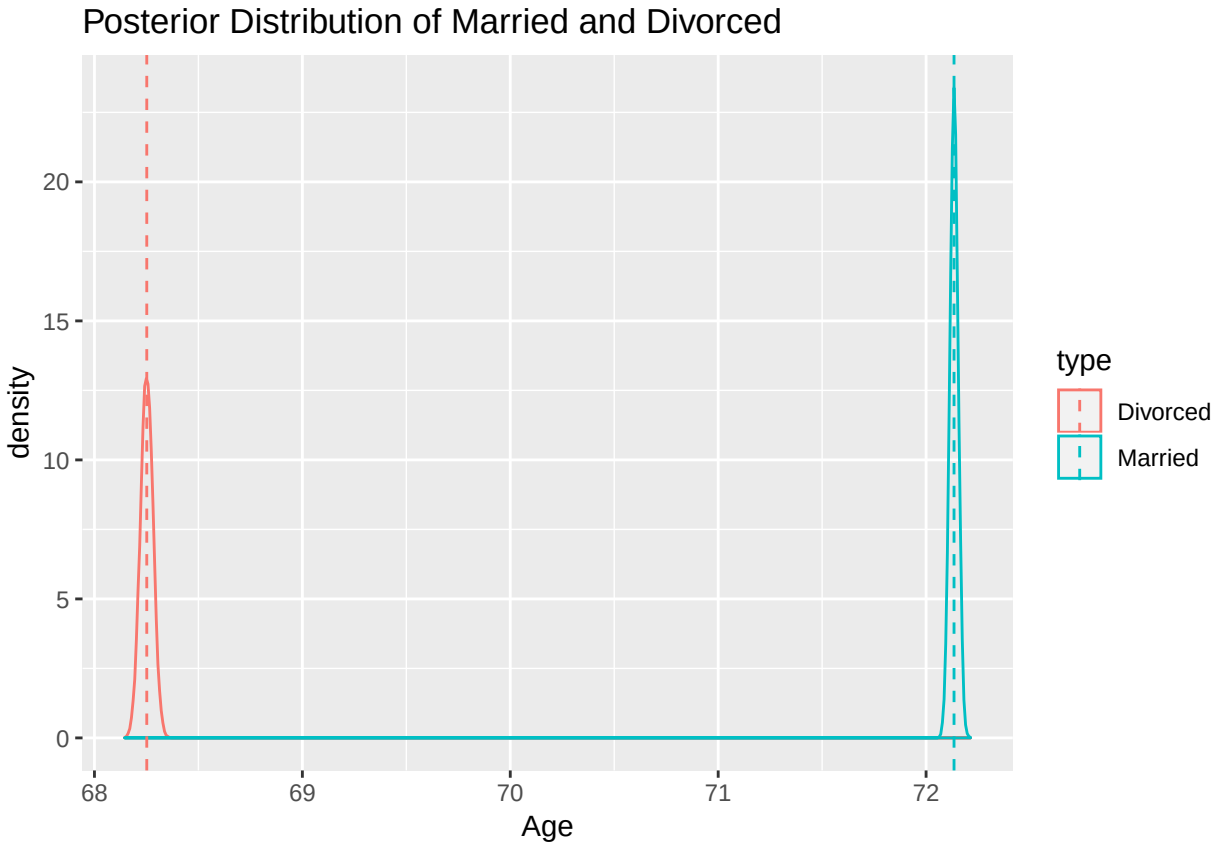


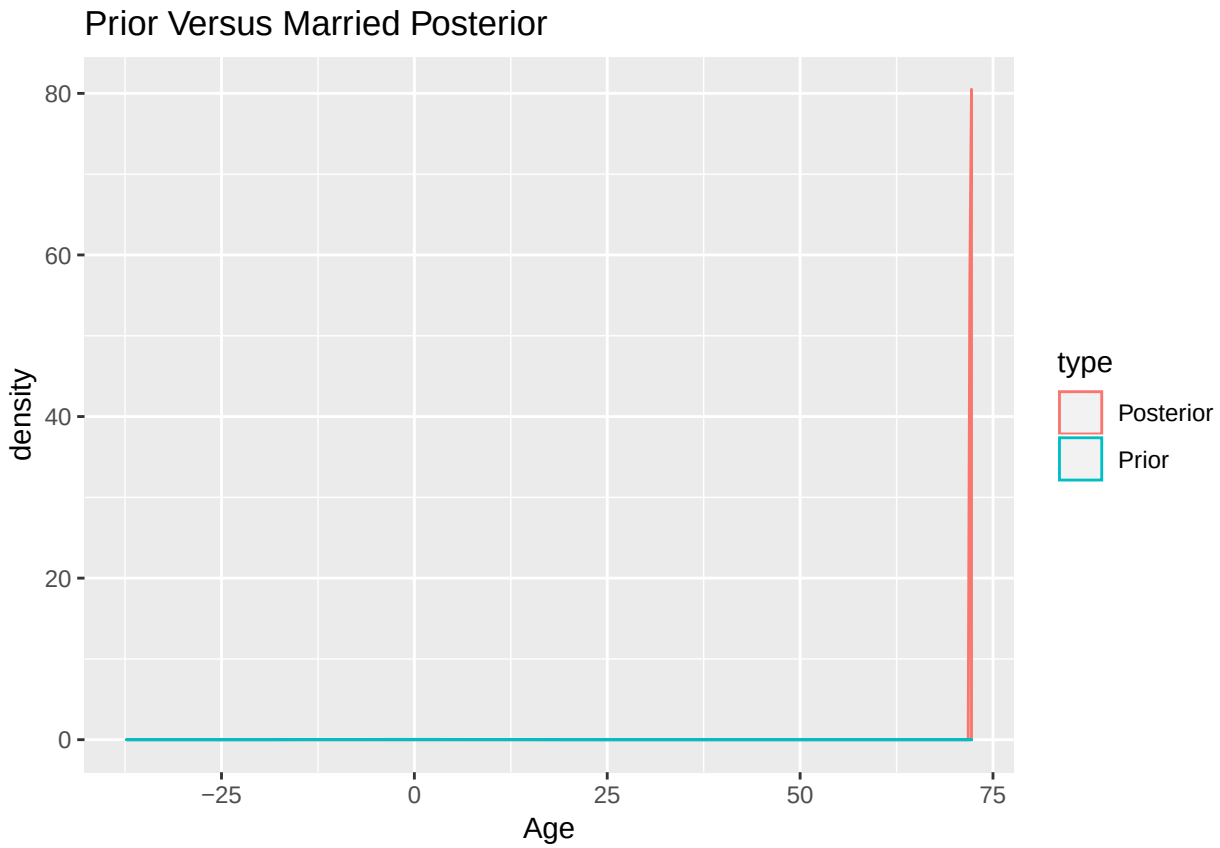
Figure 2: Density Plots

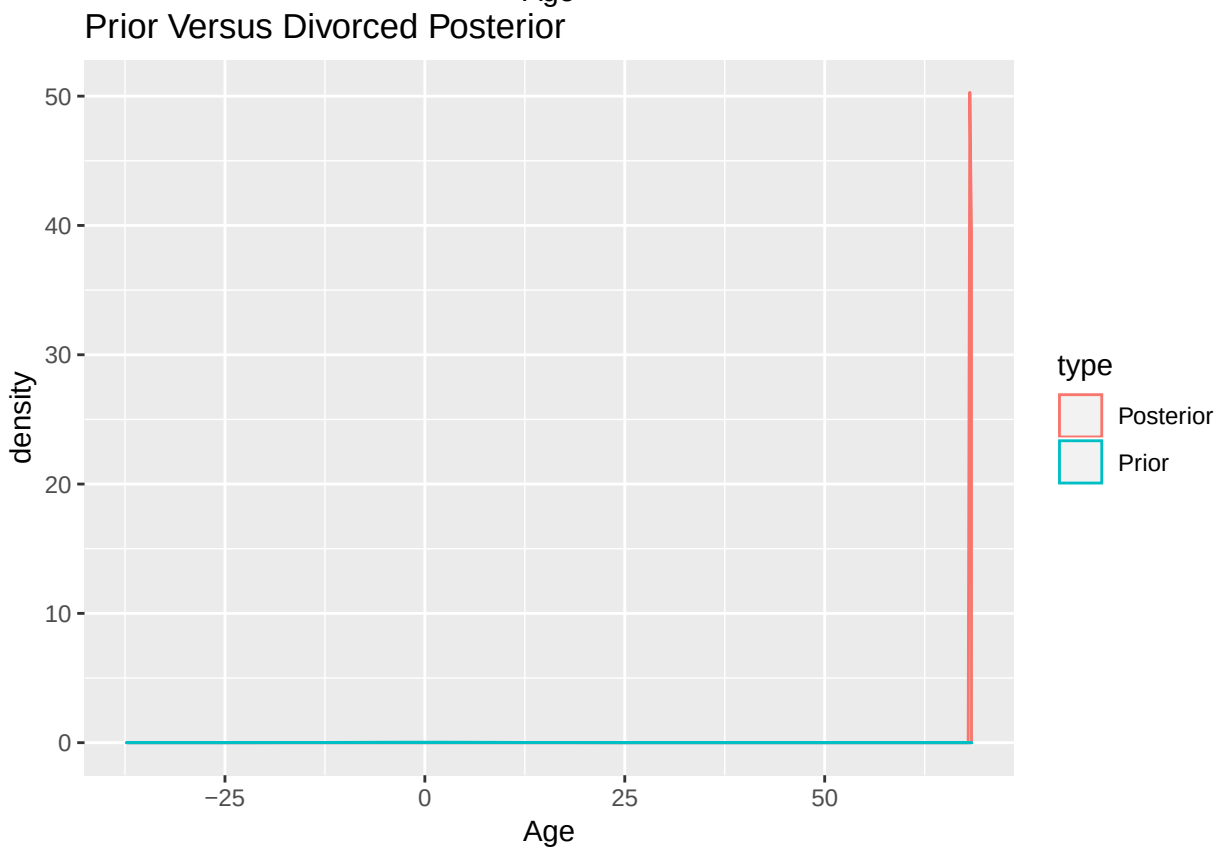
Results

Included next is a graph of the posterior distriution of the married and divorced average age of death. The dashed lines are the means of the distributions. We see that there is a difference between the average of death between the two groups.



The following two graphs of are the prior distribution versus the posterior distributions for the two groups. Because we decided on a prior distribution of N(0,10^2), we can see a large difference between the two groups on each graph. It is hard to see the N(0,10^2) graph because it is spread out much more than the posterior distributions.





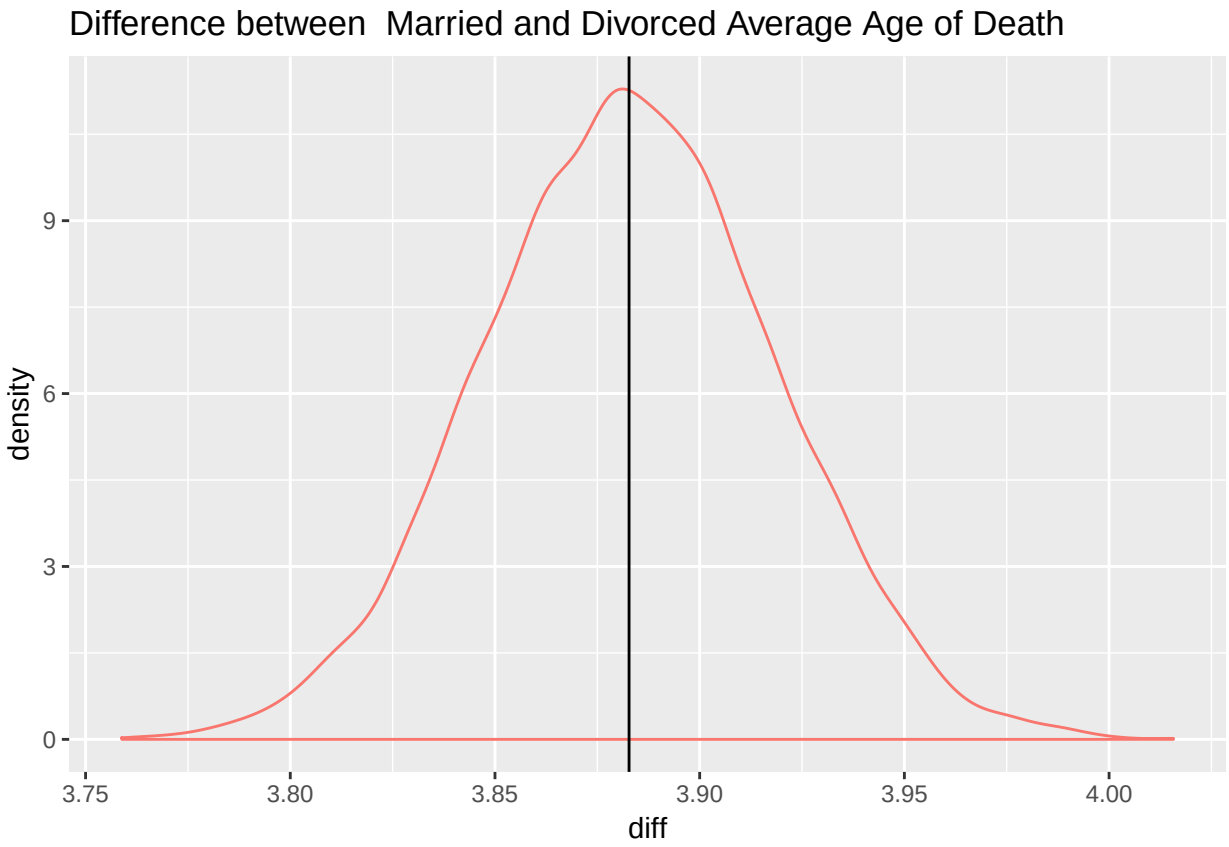
Next is a table of the credible intervals of the posterior means. There is a 95% probility that the average of

death among married is between the credible interval of 72.10 and 72.17 and there is a 95% probility that the average of death among divorced is between the credible interval of 68.19 and 68.31. We also calculated the difference between the means of the distribution. It 95% probable that the true mean of difference lies in the credible interval of 3.81 and 3.95.

Table 2: Posterior Credible Intervals of Means (on Age of Death)

	2.5%	97.5%
Married	72.10	72.17
Divorced	68.19	68.31
Difference	3.81	3.95

On the following graph, we included the density distribution of the difference between the groups Married and Divorced for average age of death. The mean for the differnce is indicated by the black vertical line.



Discussion

Our research used the data tha only include each individual marital status, and the age of death. We did not have the age at which the individuls married or divorced, or how many times they were married. Additionally, our data only included individuals in the United States. Further research can be conducted that will take into account these variables, but we are able to show that on average, married individuals live almost 4 years longer than divorced individuals.

Appendix

Rcode

```
## ----setup, echo=FALSE----------------------------------------------------
library(knitr)
library(kableExtra)
library(invgamma)
library(ggplot2)
load("divorced.RData")
load("married.RData")
load("Summary Table.RData")

## ---- echo=FALSE, warning=FALSE,message=FALSE-----------------------------
df <- rnorm(10000, 0, 10)

df <- as.data.frame(df)

ggplot(df, aes(x=df)) +
 geom_histogram(aes(y=..density..), colour="black", fill="white", binwidth = 1)+
 geom_density(alpha=.2, fill="#FF6666")+
  ggtitle("Prior N(0,10^2)")+
  xlab("x")

x <- seq(0, 5, .01)
qplot(x, dinvgamma(x, 0.01, 0.01), geom = "line") +
  ggtitle("Prior IG(0.01,0.01)")+
  ylab("Density")

## ----label1, out.width = "100%", fig.cap = "Boxplot", echo = FALSE-------
include_graphics("Report Files/Boxplot.pdf")

## ----SummaryStats, echo=FALSE---------------------------------------------
kable(TableSummary, caption = "Summary Statistics (age of death)")%>%
kable_styling(latex_options = c("striped", "hold_position", "scale_down"))

## ----label, out.width = "100%", fig.cap = "Density Plots", echo = FALSE----
include_graphics("Report Files/Density Plot.pdf")

## ---- echo=FALSE, results='hide', message=FALSE--------------------------
diff<-marriedMu[-c(1:500)]-divorcedMu[-c(1:500)]
diffMean <- mean(diff)
ci <- quantile(diff,c(0.025,0.975))

marriedD <- data.frame(Age=marriedMu[-c(1:500)])
marriedD$type <- rep("Married",length(marriedMu[-c(1:500)]), by=1)

divorcedD <- data.frame(Age=divorcedMu[-c(1:500)])
divorcedD$type <- rep("Divorced",length(divorcedMu[-c(1:500)]), by=1)

MDD <- rbind(marriedD, divorcedD)
```

```
library(plyr)
mu <- ddply(MDD, "type", summarise, grp.mean=mean(Age))

prior <- rnorm(length(divorcedMu[-c(1:500)]), 0,10)

priorD <- data.frame(Age=prior)
priorD$type <- rep("Prior",length(marriedMu[-c(1:500)]), by=1)

marriedD1 <- data.frame(Age=marriedMu[-c(1:500)])
marriedD1$type <- rep("Posterior",length(marriedMu[-c(1:500)]), by=1)

marriedDP <- rbind(marriedD1, priorD)

divorcedD1 <- data.frame(Age=divorcedMu[-c(1:500)])
divorcedD1$type <- rep("Posterior",length(divorcedMu[-c(1:500)]), by=1)

divorcedDP <- rbind(divorcedD1,priorD)


## ---- echo=FALSE-----------------------------------------------------------
ggplot(MDD,aes(x=Age, color=type)) + geom_density()+
  geom_vline(data=mu, aes(xintercept=grp.mean, color=type),
             linetype="dashed")+
  ggtitle("Posterior Distribution of Married and Divorced")

## ---- echo=FALSE-----------------------------------------------------------
ggplot(marriedDP,aes(x=Age, color=type)) + geom_density()+
  ggtitle("Prior Versus Married Posterior")

ggplot(divorcedDP,aes(x=Age, color=type)) + geom_density()+
  ggtitle("Prior Versus Divorced Posterior")

## ---- echo=FALSE-----------------------------------------------------------

marriedN <- marriedMu[-c(1:500)]
divorcedN <- divorcedMu[-c(1:500)]
Mci <- quantile(marriedN,c(0.025,0.975))
Dci <- quantile(divorcedN,c(0.025,0.975))

outputTable <- rbind(round(Mci, 2), round(Dci, 2), round(ci, 2))

rownames(outputTable) <- c("Married", "Divorced", "Difference")

kable(outputTable, caption = "Posterior Credible Intervals
of Means (on Age of Death)")%>%
kable_styling(latex_options = c("striped", "hold_position"))


## ---- echo=FALSE-----------------------------------------------------------
ggplot(data.frame(diff), aes(x=diff, color='red'))+
  geom_density()+
  geom_vline(xintercept=diffMean, color='black')+
  ggtitle("Difference between  Married and Divorced Average Age of Death")+
  theme(legend.position = 'none')
```

Source Files

```
## ------ LOAD DATA FILE -----
# library(readr)
#
# AllData <- read_csv("\2015_data.csv")
# AllData <- AllData[c(8,15)]
#
# save(AllData, file="data.RData")

load("data.RData")

library(dplyr)
married <- AllData %>% filter(marital_status == "M")
married <- married %>% mutate(detail_age = as.numeric(detail_age))
married <- married %>% filter(detail_age >= 18 & detail_age <= 150)

single <- AllData %>% filter(marital_status == "S")
single <- single %>% mutate(detail_age = as.numeric(detail_age))
single <- single %>% filter(detail_age >= 18 & detail_age <= 150)

widow <- AllData %>% filter(marital_status == "W")
widow <- widow %>% mutate(detail_age = as.numeric(detail_age))
widow <- widow %>% filter(detail_age >= 18 & detail_age <= 150)

divorced <- AllData %>% filter(marital_status == "D")
divorced <- divorced %>% mutate(detail_age = as.numeric(detail_age))
divorced <- divorced %>% filter(detail_age >= 18 & detail_age <= 150)


## -- Creating the posterior(Divorced)
library(MASS)
library(invgamma)

y<-divorced$detail_age
n<-length(y)
ybar<-mean(y)

m<-0
v<-10^2
a<-0.01
b<-0.01

#Create vectors
mu<-numeric()
sig2<-numeric()

nRep<-10000

sig2[1]<-var(y)
```

```
mu[1]<-ybar


for(j in 2:nRep) {
  #update mu based on sigma^2
  vstar<-1/(n/sig2[j-1]+1/v)
  mstar<-vstar*(n*ybar/sig2[j-1]+m/v)
  mu[j]<-rnorm(1,mstar,sqrt(vstar))

  #update sigma^2 based on updated mu
  astar<-n/2+a
  bstar<-sum((y-mu[j])^2)+b
  sig2[j]<-rinvgamma(1,astar,rate=bstar)
}

divorcedMu <- mu
divorcedSig2 <- sig2

save(divorcedMu, divorcedSig2, file="divorced.RData")


par(mfrow=c(2,2))
plot(divorcedMu, type='l', main = "Mean of Divorced Age of Death")
acf(divorcedMu[-c(1:50)])
plot(divorcedSig2, type='l')
acf(divorcedSig2[-c(1:50)])
par(mfrow=c(1,1))

mean(divorcedMu[-c(1:500)])
sqrt(mean(divorcedSig2[-c(1:500)]))

quantile(divorcedMu[-c(1:500)], c(0.025, 0.975))
quantile(divorcedSig2[-c(1:500)], c(0.025, 0.975))

## -- Creating the posterior(Married)
library(MASS)
library(invgamma)

y<-married$detail_age
n<-length(y)
ybar<-mean(y)

m<-0
v<-10^2
a<-0.01
b<-0.01

#Create vectors
mu<-numeric()
sig2<-numeric()

nRep<-10000

sig2[1]<-var(y)
```

```
mu[1]<-ybar

for(j in 2:nRep) {
  #update mu based on sigma^2
  vstar<-1/(n/sig2[j-1]+1/v)
  mstar<-vstar*(n*ybar/sig2[j-1]+m/v)
  mu[j]<-rnorm(1,mstar,sqrt(vstar))

  #update sigma^2 based on updated mu
  astar<-n/2+a
  bstar<-sum((y-mu[j])^2)+b
  sig2[j]<-rinvgamma(1,astar,rate=bstar)
}

marriedMu <- mu
marriedSig2 <- sig2

save(marriedMu, marriedSig2, file="married.RData")

plot(marriedMu, type='l')
plot(marriedSig2, type='l')

mean(marriedMu[-c(1:500)])
sqrt(mean(marriedSig2[-c(1:500)]))

quantile(marriedMu[-c(1:500)], c(0.025, 0.975))
quantile(marriedSig2[-c(1:500)], c(0.025, 0.975))

## ---- Graphs -----
library(ggplot2)

Sgraph <- ggplot(single, aes(x=detail_age)) +
  geom_histogram(aes(y=..density..), colour="black", fill="white", binwidth = 5)+
  geom_density(alpha=.2, fill="#FF6666") +
  geom_vline(aes(xintercept=mean(detail_age)),
             color="lightsteelblue3", linetype="dashed", size=1) +
  geom_vline(aes(xintercept=median(detail_age)),
             color="darkseagreen3", linetype="dashed", size=1) +
  xlab("Fms") +
  ylab("Density") +
  ggtitle("Single Age of Death Density Distribution")

Mgraph <- ggplot(married, aes(x=detail_age)) +
  geom_histogram(aes(y=..density..), colour="black", fill="white", binwidth = 5)+
  geom_density(alpha=.2, fill="#FF6666") +
  geom_vline(aes(xintercept=mean(detail_age)),
             color="lightsteelblue3", linetype="dashed", size=1) +
  geom_vline(aes(xintercept=median(detail_age)),
             color="darkseagreen3", linetype="dashed", size=1) +
  xlab("Fms") +
  ylab("Density") +
  ggtitle("Married Age of Death Density Distribution")

Wgraph <- ggplot(widow, aes(x=detail_age)) +
```

```
  geom_histogram(aes(y=..density..), colour="black", fill="white", binwidth = 5)+
  geom_density(alpha=.2, fill="#FF6666") +
  geom_vline(aes(xintercept=mean(detail_age)),
             color="lightsteelblue3", linetype="dashed", size=1) +
  geom_vline(aes(xintercept=median(detail_age)),
             color="darkseagreen3", linetype="dashed", size=1) +
  xlab("Fms") +
  ylab("Density") +
  ggtitle("Widow Age of Death Density Distribution")

Dgraph <- ggplot(divorced, aes(x=detail_age)) +
  geom_histogram(aes(y=..density..), colour="black", fill="white", binwidth = 5)+
  geom_density(alpha=.2, fill="#FF6666") +
  geom_vline(aes(xintercept=mean(detail_age)),
             color="lightsteelblue3", linetype="dashed", size=1) +
  geom_vline(aes(xintercept=median(detail_age)),
             color="darkseagreen3", linetype="dashed", size=1) +
  xlab("Fms") +
  ylab("Density") +
  ggtitle("Divorced Age of Death Density Distribution")

## --- EDA Files -----
source("ParallelGraphs.R")

library(rowr)

dfGraph <- cbind.fill(married$detail_age,
                      divorced$detail_age,
                      single$detail_age,
                      widow$detail_age, fill=NA)

graphNames <- c("Married", "Divorced", "Single", "Widow")

output <- makeGraph(dfGraph, graphNames)

# source("Graph Functions/ggplot2_multiplot.R")
# multiplot(output, cols=2)

# test2 <- dfGraph
# colnames(test2) <- graphNames
#
# ggplot(na.omit(stack(test2)), aes(x = ind, y = values, fill=ind)) +
#   geom_boxplot()

df <- dfGraph

TableSummary <- do.call(data.frame,
                        list(mean = apply(df, 2, mean, na.rm=TRUE),
                             sd = apply(df, 2, sd, na.rm=TRUE),
                             median = apply(df, 2, median, na.rm=TRUE),
                             min = apply(df, 2, min, na.rm=TRUE),
                             max = apply(df, 2, max, na.rm=TRUE),
                             n = apply(df, 2, function(x){sum(!is.na(x))})))
```

```
row.names(TableSummary) <- graphNames

save(TableSummary, file="Summary Table.RData")

## Multiplot Function for ggplot2

# Multiple plot function
#
# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
# - cols:   Number of columns in layout
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.
#
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
# then plot 1 will go in the upper left, 2 will go in the upper right, and
# 3 will go all the way across the bottom.
#
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  require(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                     ncol = cols, nrow = ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])

  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                      layout.pos.col = matchidx$col))
    }
  }
}


## Parallel Processing Creation of Graph ---
```

```
makeGraph <- function(df, nms=NULL)
{
  if(!require(foreach)){
    install.packages("foreach")
    library(foreach)
  }

  if(!require(doParallel)){
    install.packages("doParallel")
    library(doParallel)
  }

  if(!require(ggplot2)){
    install.packages("ggplot2")
    library(ggplot2)
  }

  if(is.null(nms))
  {
    names <- c(rep("No Title"), length(df))
  }

  cores=detectCores()
  cl <- makeCluster(cores[1]-1) #not to overload your computer
  registerDoParallel(cl)

  m.list <- list()


  m.list <- foreach(i = 1:length(df),
                    .packages = "ggplot2") %dopar% {
                      breaks <- pretty(range(na.omit(df[,i])), n =
                      nclass.FD(na.omit(df[,i])), min.n = 1)
                      bwidth <- breaks[2]-breaks[1]

                      gmean <- mean(na.omit(df[,i]))
                      gmedian <- median(na.omit(df[,i]))

                      graph <- ggplot(na.omit(df[i]), aes_string(x=colnames(df[i]))) +
                        geom_histogram(aes(y=..density..),
                        colour="black", fill="white", binwidth = bwidth)+
                        geom_density(alpha=.2, fill="#FF6666") +
                        geom_vline(aes(xintercept=mean(gmean)),
                                   color="lightsteelblue3", linetype="dashed", size=1) +
                        geom_vline(aes(xintercept=median(gmedian)),
                                   color="darkseagreen3", linetype="dashed", size=1) +
                        ggtitle(nms[i]) +
                        xlab("Fms") +
                        ylab("Density")

                      graph
                    }

  stopCluster(cl)
```

```
  stopImplicitCluster()

  return(m.list)

}
```