Suicide Rates Data

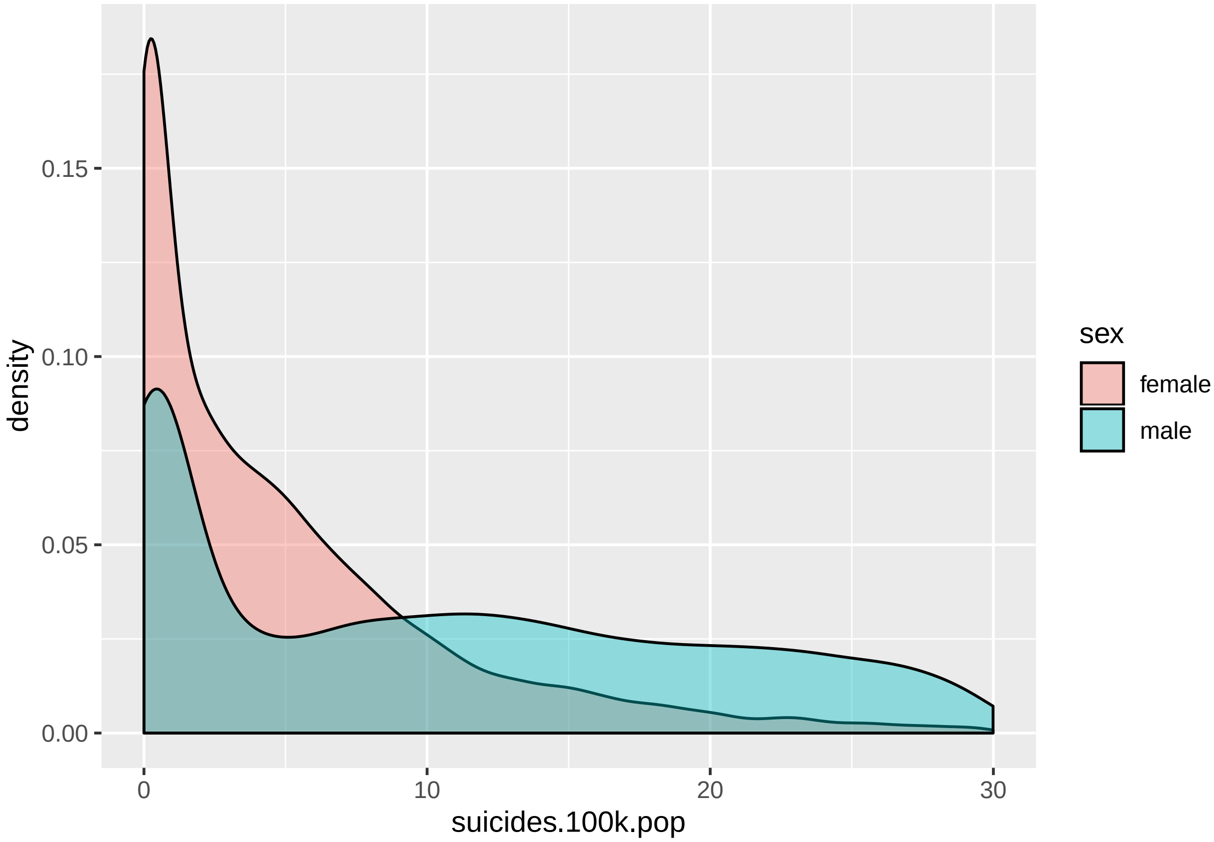
Load CSV and use summary function

```
sData <- read.csv("~/Downloads/master.csv")

summary(sData)
```

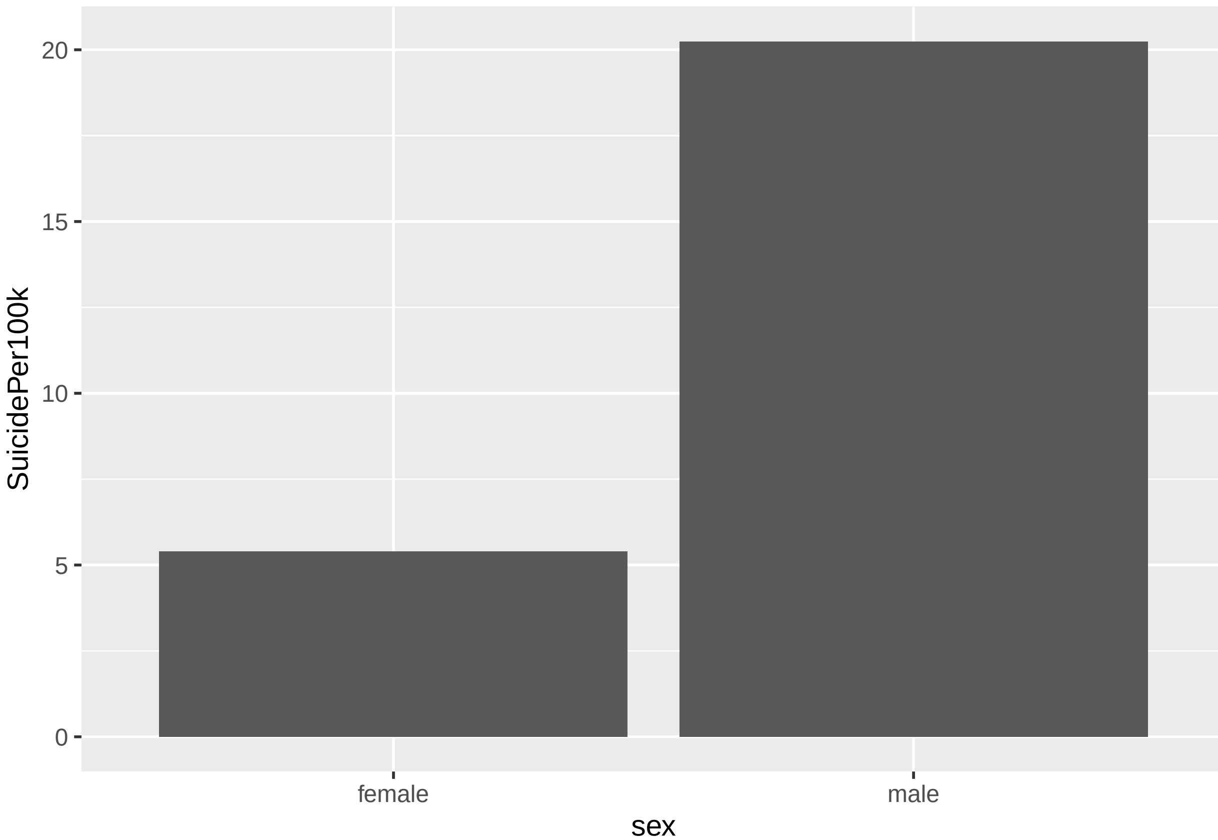
```
##         country           year          sex               age
##  Austria    :  382   Min.   :1985   female:13910   15-24 years:4642
##  Iceland    :  382   1st Qu.:1995   male  :13910   25-34 years:4642
##  Mauritius  :  382   Median :2002                  35-54 years:4642
##  Netherlands:  382   Mean   :2001                  5-14 years :4610
##  Argentina  :  372   3rd Qu.:2008                  55-74 years:4642
##  Belgium    :  372   Max.   :2016                  75+ years  :4642
##  (Other)    :25548
##   suicides_no        population       suicides.100k.pop
##  Min.   :    0.0   Min.   :     278   Min.   :  0.00
##  1st Qu.:    3.0   1st Qu.:   97498   1st Qu.:  0.92
##  Median :   25.0   Median :  430150   Median :  5.99
##  Mean   :  242.6   Mean   : 1844794   Mean   : 12.82
##  3rd Qu.:  131.0   3rd Qu.: 1486143   3rd Qu.: 16.62
##  Max.   :22338.0   Max.   :43805214   Max.   :224.97
##
##       country.year    HDI.for.year            gdp_for_year....
##  Albania1987:   12   Min.   :0.483   1,002,219,052,968:   12
##  Albania1988:   12   1st Qu.:0.713   1,011,797,457,139:   12
##  Albania1989:   12   Median :0.779   1,016,418,229    :   12
##  Albania1992:   12   Mean   :0.777   1,018,847,043,277:   12
##  Albania1993:   12   3rd Qu.:0.855   1,022,191,296    :   12
##  Albania1994:   12   Max.   :0.944   1,023,196,003,075:   12
##  (Other)    :27748   NA's   :19456   (Other)          :27748
##  gdp_per_capita....           generation
##  Min.   :   251     Boomers        :4990
##  1st Qu.:  3447     G.I. Generation:2744
##  Median :  9372     Generation X   :6408
##  Mean   : 16866     Generation Z   :1470
##  3rd Qu.: 24874     Millenials     :5844
##  Max.   :126352     Silent         :6364
##
```

Gender Data

```
#Filtering data for density because outliers are rare and skew density graph
densityData <- sData %>% filter(suicides.100k.pop < 30)
library(ggplot2)
# Density based on Gender, Males tend to commit suicide far more often, shifted data to not include ext
suicideBySex<- sData %>% select(sex, suicides.100k.pop) %>% group_by(sex) %>% summarise(SuicidePer100k=

ggplot(densityData, aes(x=suicides.100k.pop,fill=sex))+
  geom_density(alpha=0.4)
```

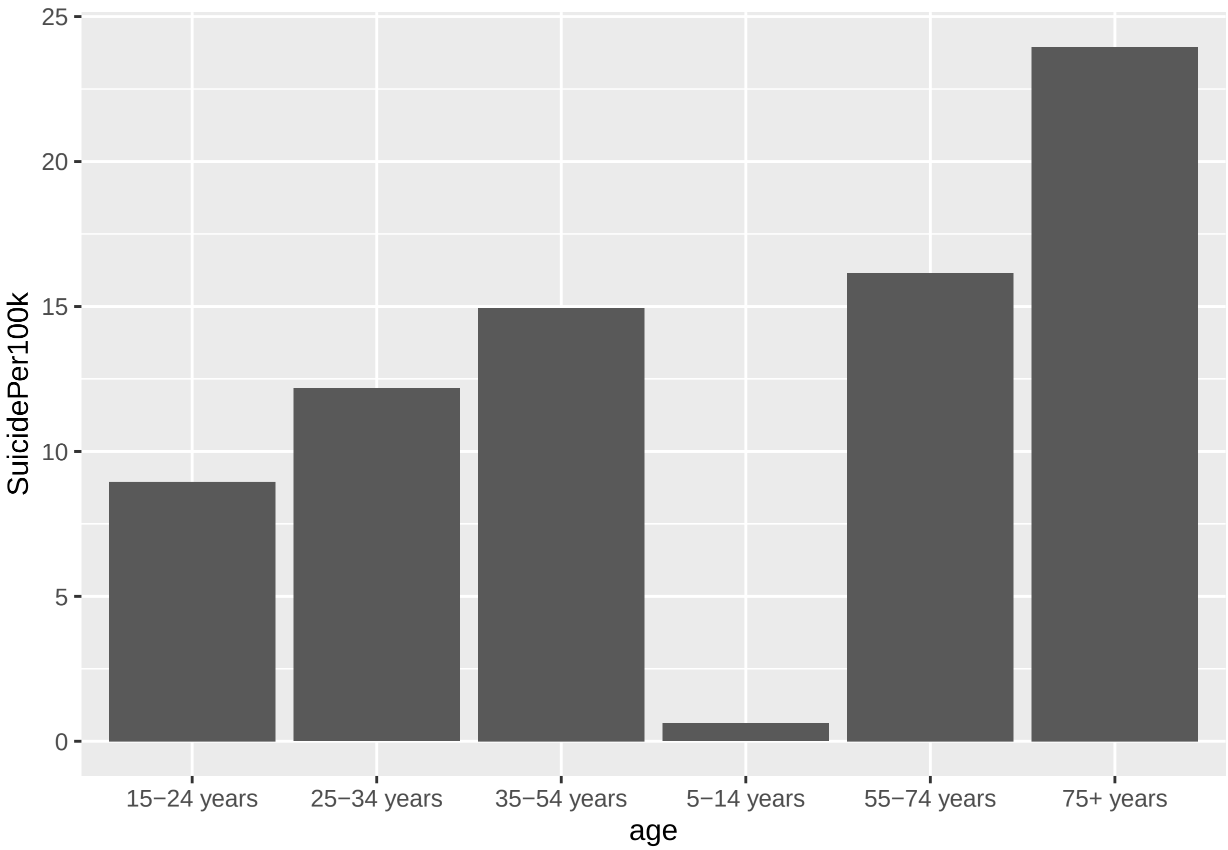



```r
ggplot(suicideBySex, aes(x=sex, y=SuicidePer100k)) +
  geom_bar(stat="identity")
```



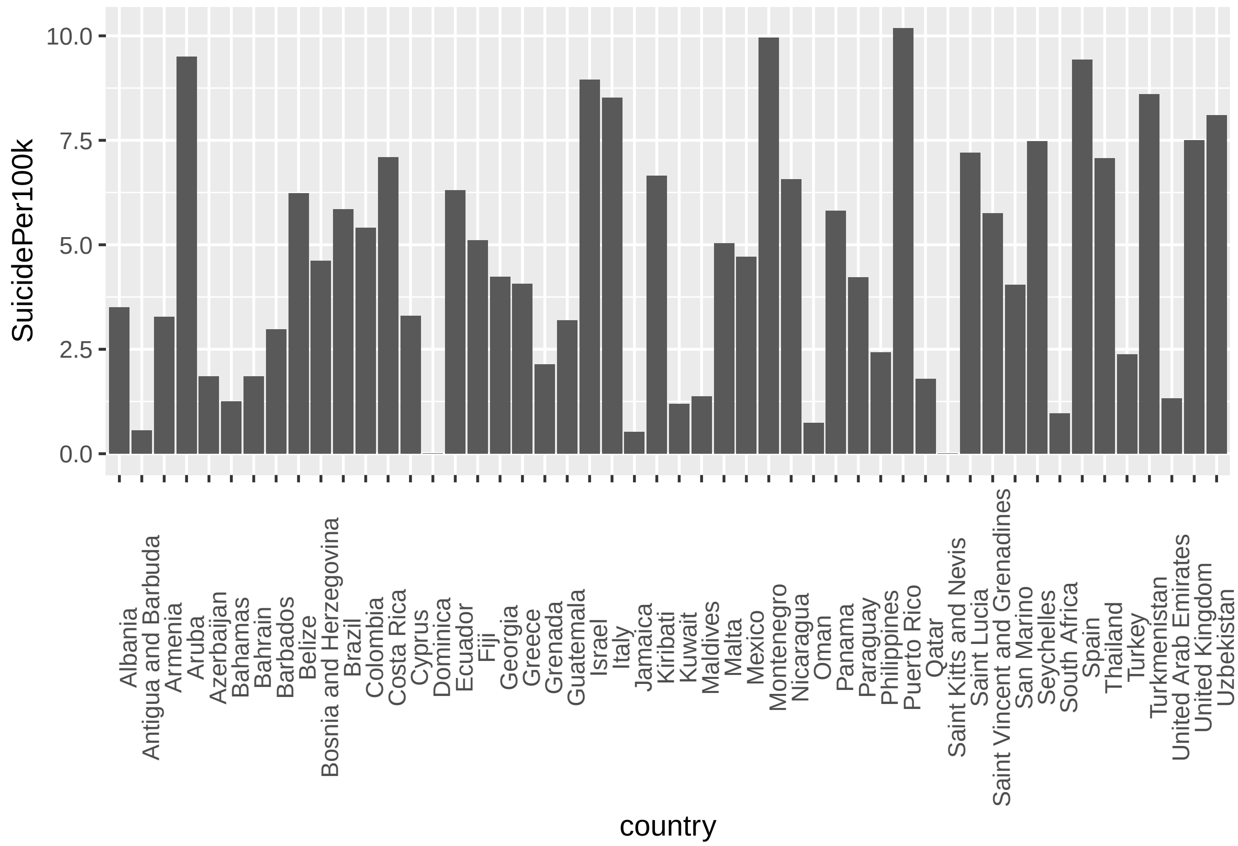
Age Data

```
#Density based on age groups, rates steadily climb as people get older

suicideByAge <- sData %>% select(age, suicides.100k.pop) %>% group_by(age) %>% summarise(SuicidePer100k=

generationAverages<-ggplot(data=suicideByAge, aes(x=age, y=SuicidePer100k)) +
  geom_bar(stat="identity")
generationAverages
```

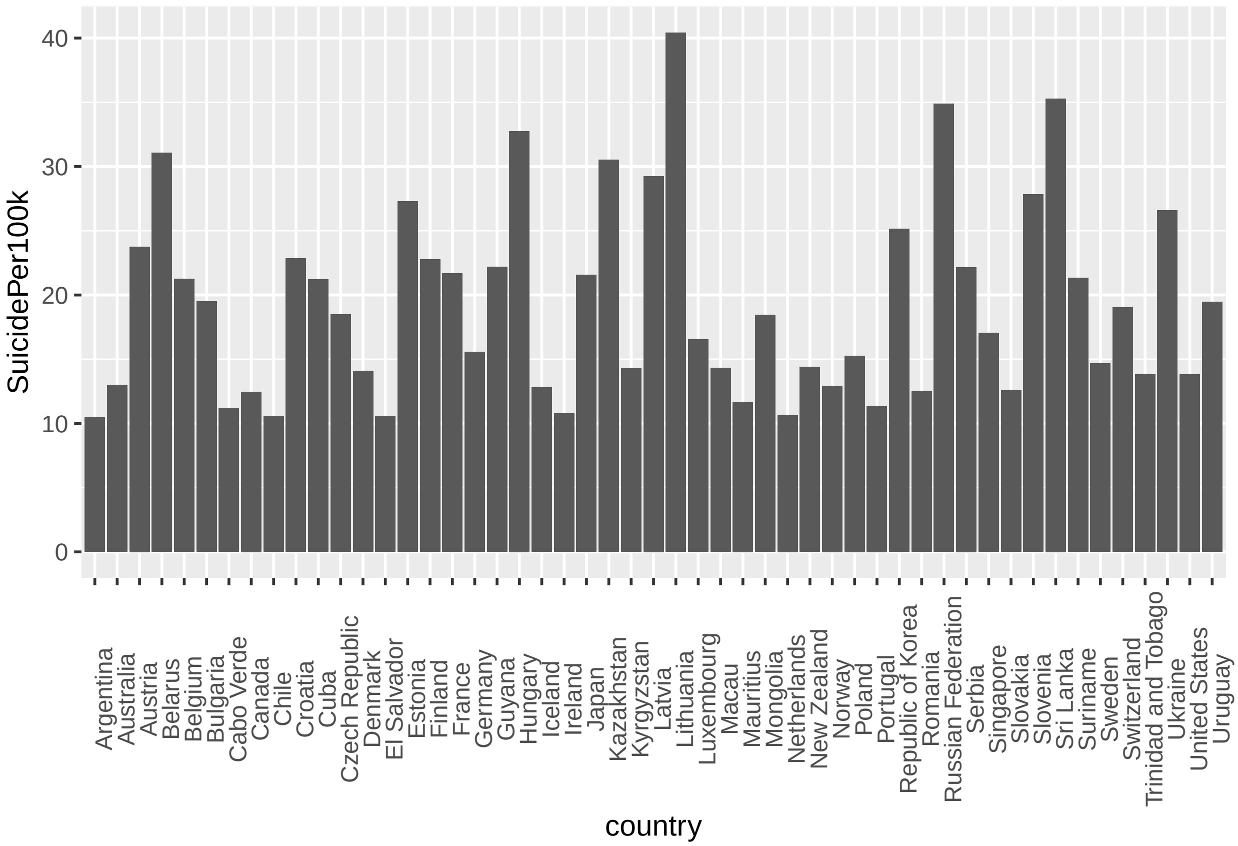


Country Data

```
#Suicide by country, broken down into two sides, arranged by rate so countryone has lowe rate countries
suicideByCountry <- sData %>% select(country, suicides.100k.pop) %>% group_by(country) %>% summarise(Su

suicideByCountry <- arrange(suicideByCountry,SuicidePer100k)
countryOne <- suicideByCountry[1:50,]
countryTwo <- suicideByCountry[51:101,]


countryOneAverage<-ggplot(data=countryOne, aes(x=country, y=SuicidePer100k)) +
  geom_bar(stat="identity") + theme(axis.text.x = element_text(angle = 90))
countryOneAverage
```

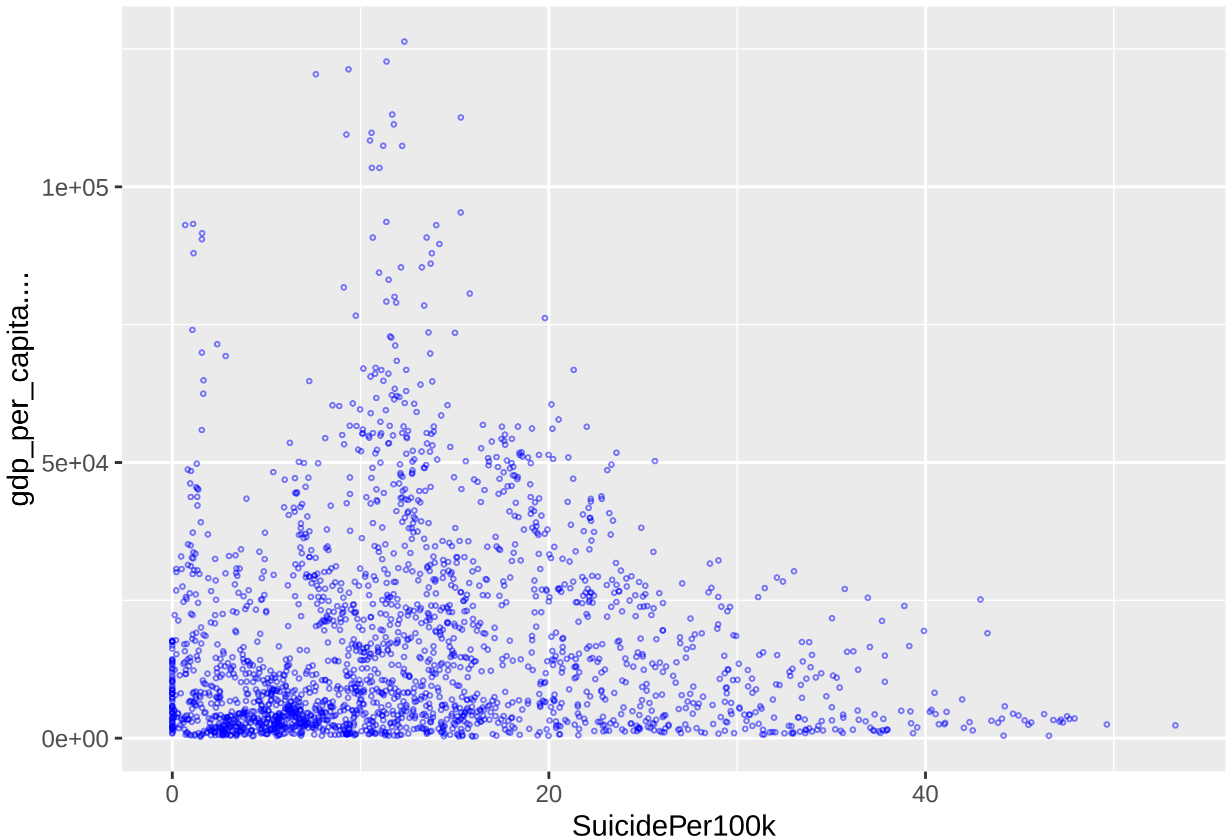



```r
countryTwoAverage<-ggplot(data=countryTwo, aes(x=country, y=SuicidePer100k)) +
  geom_bar(stat="identity") + theme(axis.text.x = element_text(angle = 90))
countryTwoAverage
```



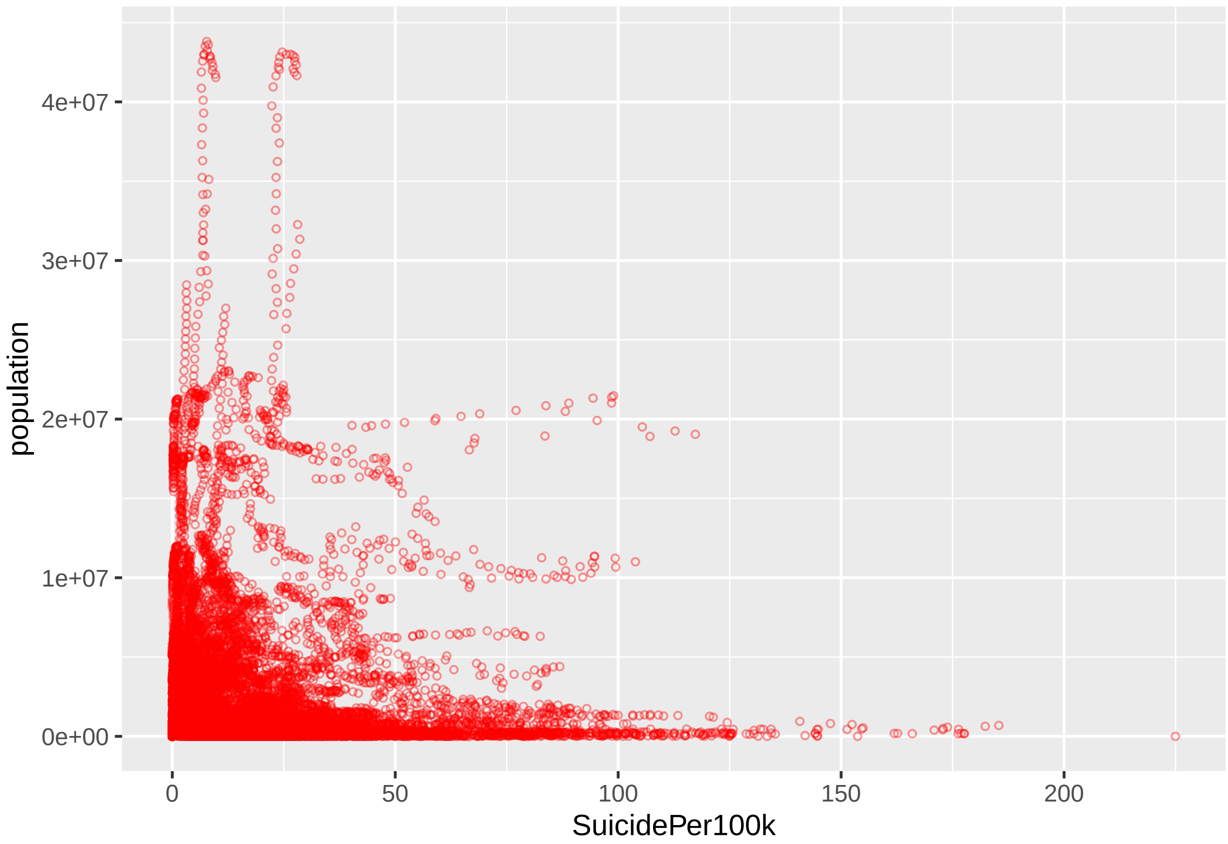
GDP Data

```
#suicide by GDP, found some meaningful correlation rates are similar until 20 + that's when poorer coun
suicideByGDP <- sData %>% select(gdp_per_capita...., suicides.100k.pop) %>% group_by(gdp_per_capita....)

ggplot(suicideByGDP, aes(x=SuicidePer100k, y=gdp_per_capita....)) +
  geom_point(size=.5, shape=1, colour = "blue",alpha=0.5)
```



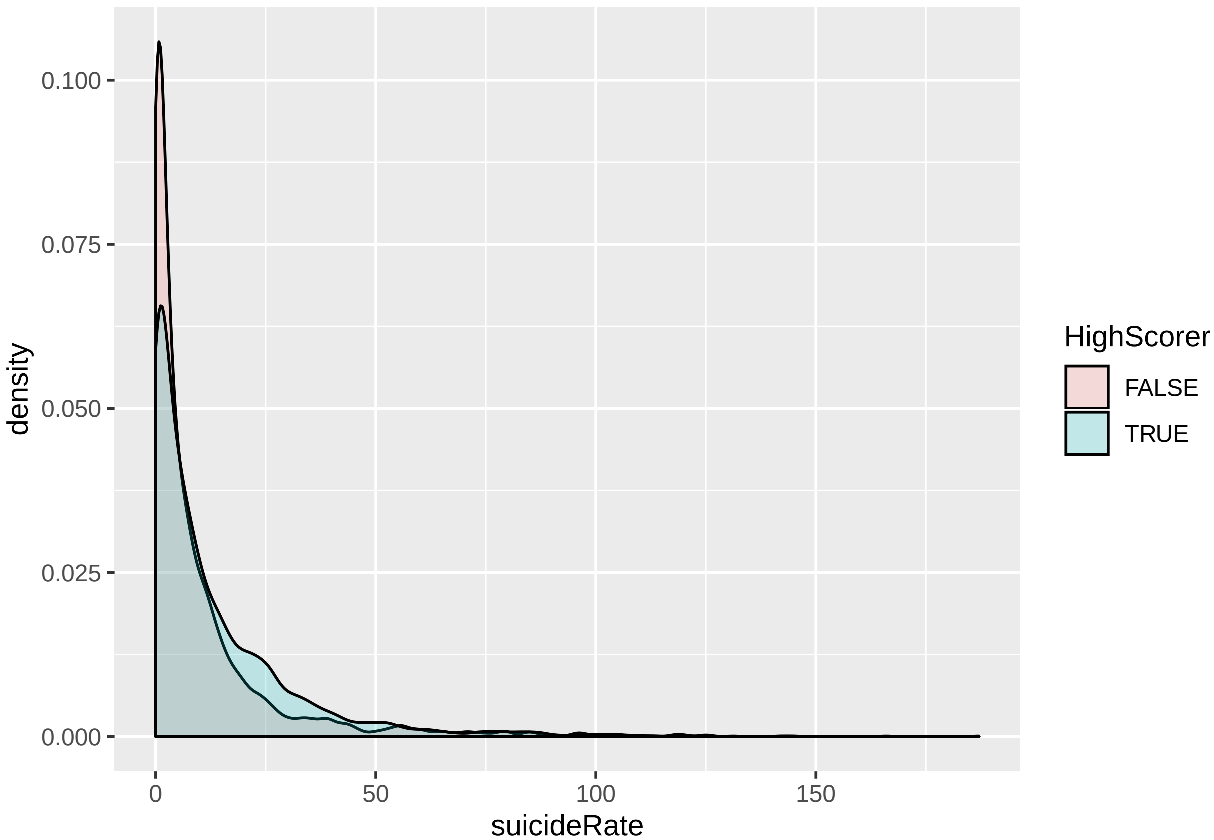
Population Data

```
#suicide by population, this was completely random no correlation
suicideByPop <- sData %>% select(population, suicides.100k.pop) %>% group_by(population) %>% summarise(S

ggplot(suicideByPop, aes(x=SuicidePer100k, y=population)) +
  geom_point(size=1, shape=1, color = "red",alpha=0.4)
```

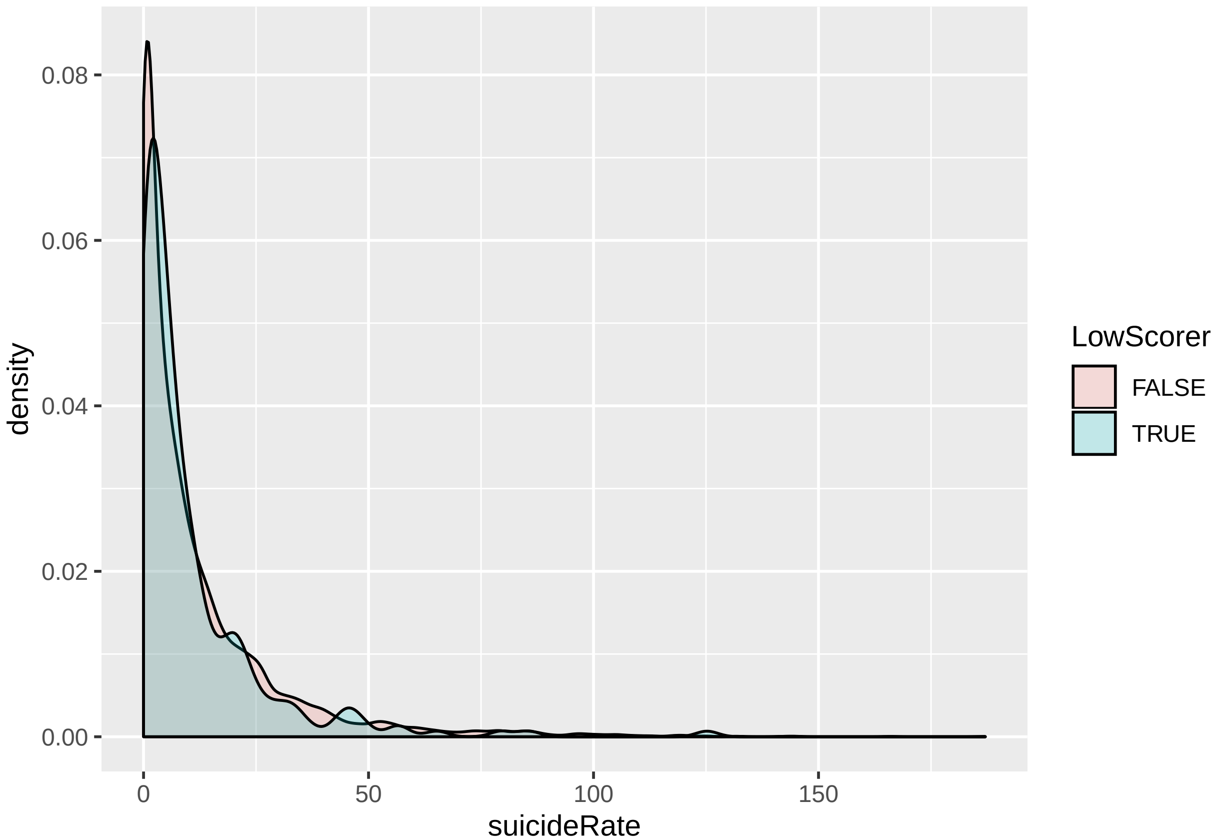


HDI Data

```
# Suicide based on countries with human development index, seperated into countries with 7.5 above/belo
suicideByHDI <- sData %>%select(HDI.for.year,suicideRate = suicides.100k.pop) %>%drop_na() %>% group_by

ggplot(suicideByHDI, aes(x=suicideRate,fill=HighScorer))+
  geom_density(alpha=0.2)
```

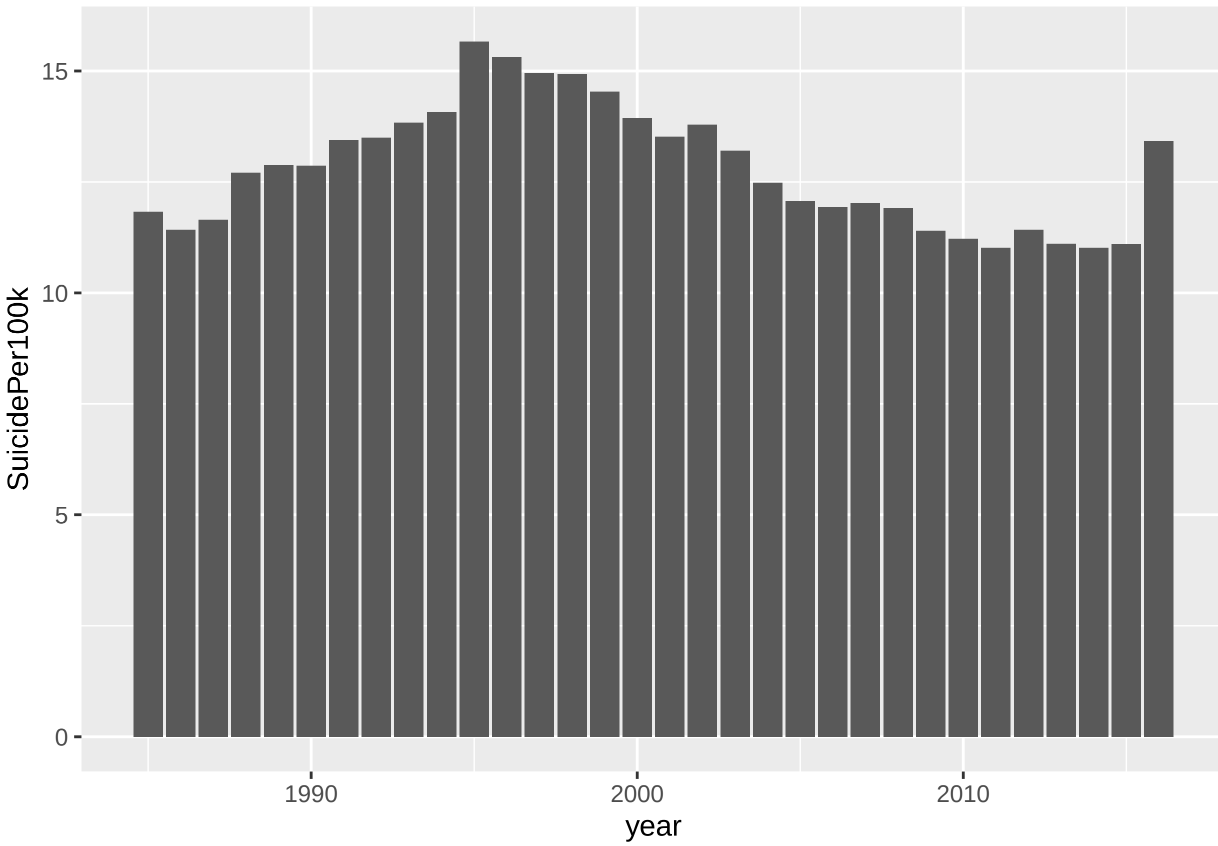



```r
ggplot(suicideByHDI, aes(x=suicideRate,fill=LowScorer))+
  geom_density(alpha=0.2)
```



Year Data

```
#Suicide based on year, went with plain bar graph no real insight gained
suicideByYear <- sData %>% select(year, suicides.100k.pop) %>% group_by(year) %>% summarise(SuicidePer1(

yearBar<-ggplot(data=suicideByYear, aes(x=year, y=SuicidePer100k)) +
  geom_bar(stat="identity")

yearBar
```



Clustering

```r
summary(sData)
```

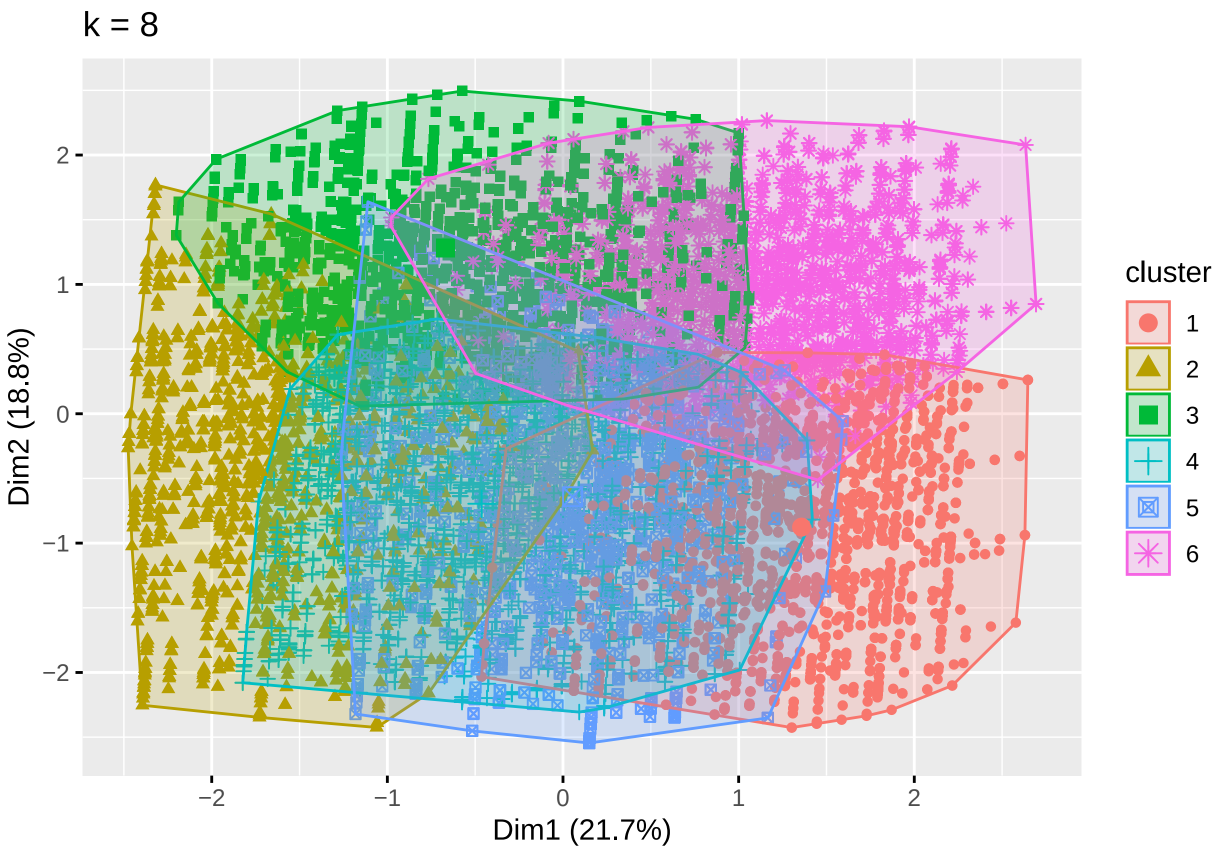
```
##        country          year          sex              age
##  Austria    :  382   Min.   :1985   female:13910   15-24 years:4642
##  Iceland    :  382   1st Qu.:1995   male  :13910   25-34 years:4642
##  Mauritius  :  382   Median :2002                  35-54 years:4642
##  Netherlands:  382   Mean   :2001                  5-14 years :4610
##  Argentina  :  372   3rd Qu.:2008                  55-74 years:4642
##  Belgium    :  372   Max.   :2016                  75+ years  :4642
##  (Other)    :25548
##   suicides_no       population       suicides.100k.pop
##  Min.   :    0.0   Min.   :     278   Min.   :  0.00
##  1st Qu.:    3.0   1st Qu.:   97498   1st Qu.:  0.92
##  Median :   25.0   Median :  430150   Median :  5.99
##  Mean   :  242.6   Mean   : 1844794   Mean   : 12.82
##  3rd Qu.:  131.0   3rd Qu.: 1486143   3rd Qu.: 16.62
##  Max.   :22338.0   Max.   :43805214   Max.   :224.97
##
##      country.year    HDI.for.year           gdp_for_year....
##  Albania1987:   12   Min.   :0.483   1,002,219,052,968:   12
##  Albania1988:   12   1st Qu.:0.713   1,011,797,457,139:   12
##  Albania1989:   12   Median :0.779   1,016,418,229    :   12
##  Albania1992:   12   Mean   :0.777   1,018,847,043,277:   12
##  Albania1993:   12   3rd Qu.:0.855   1,022,191,296    :   12
```

```
##  Albania1994:   12   Max.   :0.944   1,023,196,003,075:   12
##  (Other)    :27748   NA's   :19456   (Other)          :27748
##  gdp_per_capita....          generation
##  Min.   :   251     Boomers        :4990
##  1st Qu.:  3447     G.I. Generation:2744
##  Median :  9372     Generation X   :6408
##  Mean   : 16866     Generation Z   :1470
##  3rd Qu.: 24874     Millenials     :5844
##  Max.   :126352     Silent         :6364
##
```

```r
sData2 <- sData
library(dplyr)
library(cluster)
library(tidyverse)

#drop values that are redundant or not useful for the model
drops <- c("country.year","HDI.for.year",'suicides_no',"gdp_for_year....","generation")
sData2 <- sData2[ , !(names(sData2) %in% drops)]
#Drop na data
sData2 <- na.omit(sData2)

#bin variables based on quartiles
sData2$population<-cut(sData2$population, c(278,97498,430150,1486143,438025124))
sData2$suicides.100k.pop <- cut(sData2$suicides.100k.pop, c(0.00,0.92,5.99,16.62,224.97))
sData2$gdp_per_capita....<- cut(sData2$gdp_per_capita....,c(251,3447,9372,24874,126352))
sData2$year<-cut(sData2$year, c(1985,1995,2002,2008,2016))

#mutate all variables into numeric, drop NA values, normalize the dataset
sData2 <- mutate_all(sData2, function(x) as.numeric(x))
sData2 <- na.omit(sData2)
sData2 <- normalize.Dataset(sData2)

clusters <- kmeans(sData2,centers=6,nstart=50)
library(factoextra)
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

```r
p3 <- fviz_cluster(clusters,geom="point", sData2) + ggtitle('k = 8')
p3
```



Naive Bayes Full Data Set

```
#model without HDI
library(e1071)

#Cut sData2 into a train/test datasets, mutate train/test back into factor,
train <- sData2[1:18467,]
test <- sData2[18468:nrow(sData2),]
train <- mutate_all(train, function(x) as.factor(x))
test <- mutate_all(test, function(x) as.factor(x))

#create model using all values stored in sData2
bayesModel <- naiveBayes(as.factor(suicides.100k.pop)~country + year+sex+age + population + gdp_per_capi
                    data = train)


#predict using bayesmodel
pred.raw <- predict(bayesModel, test, type = "class")

#create confusion matrix based on how well it predicts suicide.100k.pop then calculate accuracy
confusion <- table(predict(bayesModel, test),
      test$suicides.100k.pop,
      dnn=c("prediction","truth"))

confusion
```

```
##                    truth
## prediction            0 0.333333333333333 0.666666666666667    1
##   0                 455               171                 0    0
##   0.333333333333333  93               761               470  108
##   0.666666666666667  27               299               493  183
##   1                  10               116               367 1064
```

```r
sum(diag(confusion)/nrow(test))
```

```
## [1] 0.6006065
```

```r
#60% accuracy not bad considering there is 4 potential options
```

Bayes With HDI

```r
#Model with HDI but a lot less rows
#reference sData(original dataset) to use with the new set used for Naive Bayes
bayesWithHDI <- sData
#Drop values that won't be used in the model
drops <- c("country.year",'suicides_no',"gdp_for_year....","generation")
bayesWithHDI <- bayesWithHDI[ , !(names(bayesWithHDI) %in% drops)]
#drop any NAs
bayesWithHDI <- na.omit(bayesWithHDI)

#Use this to bin continuuous values into categorical values, uses their quartiles as binning cuts/break
bayesWithHDI$population<-cut(bayesWithHDI$population, c(278,97498,430150,1486143,438025124))
bayesWithHDI$suicides.100k.pop <- cut(bayesWithHDI$suicides.100k.pop, c(0.00,0.92,5.99,16.62,224.97))
bayesWithHDI$gdp_per_capita....<- cut(bayesWithHDI$gdp_per_capita....,c(251,3447,9372,24874,126352))
bayesWithHDI$year<-cut(bayesWithHDI$year, c(1985,1995,2002,2008,2016))
bayesWithHDI$HDI.for.year<-cut(bayesWithHDI$HDI.for.year, c(.4830,.7130,.7790,.8550,.9440))

#mutate dataset into numeric, omit any nas again, normalize dataset
bayesWithHDI <- mutate_all(bayesWithHDI, function(x) as.numeric(x))
bayesWithHDI <- na.omit(bayesWithHDI)
bayesWithHDI <- normalize.Dataset(bayesWithHDI)

#make train/test datasets, revert back to factor
trainHDI <- bayesWithHDI[1:6000,]
testHDI <- bayesWithHDI[6001:nrow(bayesWithHDI),]
trainHDI <- mutate_all(trainHDI, function(x) as.factor(x))
testHDI <- mutate_all(testHDI, function(x) as.factor(x))

#naive bayes model this time uses HDI
modelHDI <- naiveBayes(suicides.100k.pop~country+year+sex+age + population + gdp_per_capita.... + HDI.f
                    data = trainHDI)
#predict class labels for test dataset based on suicides.100k
pred.raw <- predict(modelHDI, testHDI, type = "class")
confusion <- table(predict(modelHDI, testHDI),
      testHDI$suicides.100k.pop,
      dnn=c("prediction","truth"))


confusion
```

```
##                     truth
## prediction            0 0.333333333333333 0.666666666666667   1
##   0                  84                39                 0   0
##   0.333333333333333   7               155                96   8
##   0.666666666666667   1               102                75  45
##   1                   0                18                64 120
```

```r
#Find the accuracy of the model
sum(diag(confusion)/nrow(testHDI))
```

```
## [1] 0.5331695
```

```r
#only 53% lower than without, probably due to dropping a ton of data to use HDI
```