Data Visualization with the Titanic Data

Karen Mazidi

Data exploration in R with the Titanic data set

for one Variable X, quantitative or qualitative

Graphical Parameters: read more here

Colors in R Graphs: read more here

Load the data

Load the Titanic data, changing certain columns to factors.

```
df <- read.csv("data/titanic.csv", na.strings="NA", header=TRUE)
str(df)
```

```
## 'data.frame':    1309 obs. of  14 variables:
##  $ pclass   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ survived : int  1 1 0 0 0 1 1 0 1 0 ...
##  $ name     : chr  "Allen, Miss. Elisabeth Walton" "Allison, Master. Hudson Trevor" "Allison, Miss. H
##  $ sex      : chr  "female" "male" "female" "male" ...
##  $ age      : num  29 0.917 2 30 25 ...
##  $ sibsp    : int  0 1 1 1 1 0 1 0 2 0 ...
##  $ parch    : int  0 2 2 2 2 0 0 0 0 0 ...
##  $ ticket   : chr  "24160" "113781" "113781" "113781" ...
##  $ fare     : num  211 152 152 152 152 ...
##  $ cabin    : chr  "B5" "C22 C26" "C22 C26" "C22 C26" ...
##  $ embarked : chr  "S" "S" "S" "S" ...
##  $ boat     : chr  "2" "11" "" "" ...
##  $ body     : int  NA NA NA 135 NA NA NA NA NA 22 ...
##  $ home.dest: chr  "St Louis, MO" "Montreal, PQ / Chesterville, ON" "Montreal, PQ / Chesterville, ON
```

```
df$survived <- as.factor(df$survived)
df$pclass <- as.factor(df$pclass)
df$sex <- factor(df$sex, levels=c("male", "female"))
```

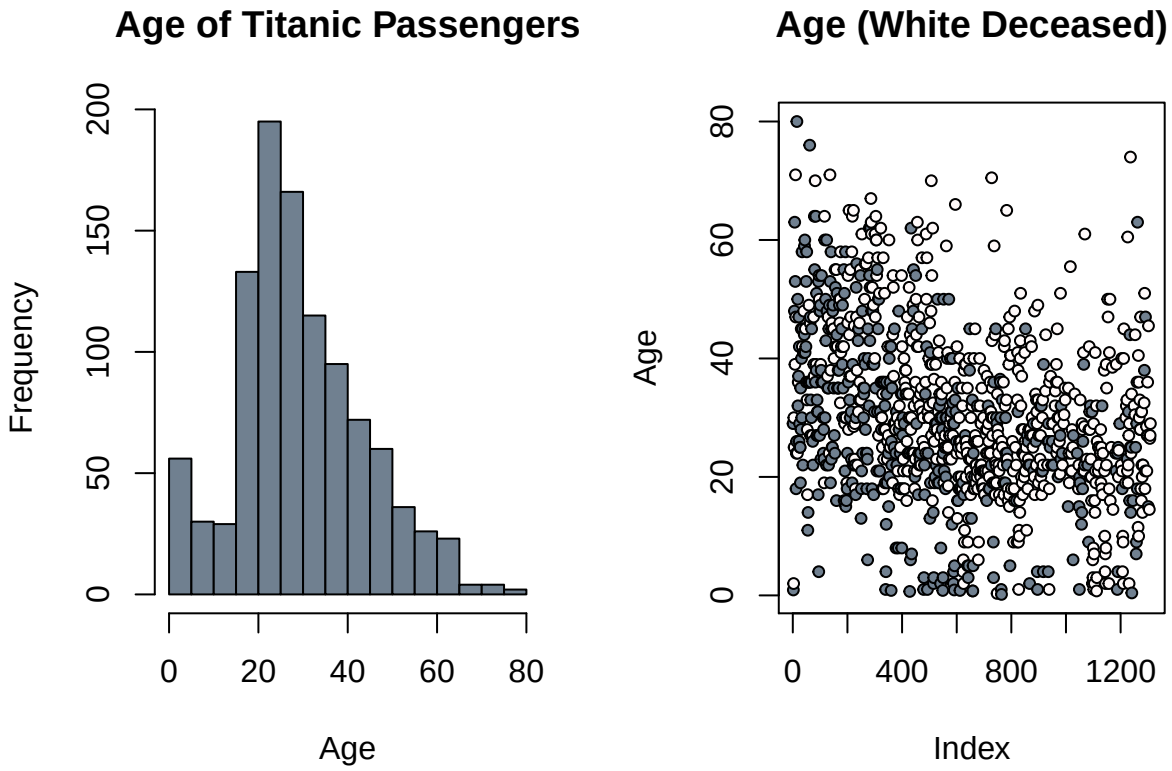
Plotting one dimension, X is quantitative

We use the par() function to specify we want to display graphs in a 1x2 grid. To easily restore the parameters, we save them before changing them so that we can restore them with the last line in the following code block.

The most common graph for one quantitative variable is the histogram. You can specify the bins but the bins that were created automatically seem fine.

Another plot that can be used for a single quantitative variable is a simple scatterplot. In this case, the x axis will just be index numbers. In the graph below we color coded the dots to be white if the person did not survive.

```
opar <- par()     # copy original settings
par(mfrow=c(1,2))
hist(df$age, col="slategray", main="Age of Titanic Passengers", xlab="Age")
plot(df$age, pch=21, cex=0.75, bg=c("snow", "slategray")[unclass(df$survived)], ylab="Age", main="Age (
```

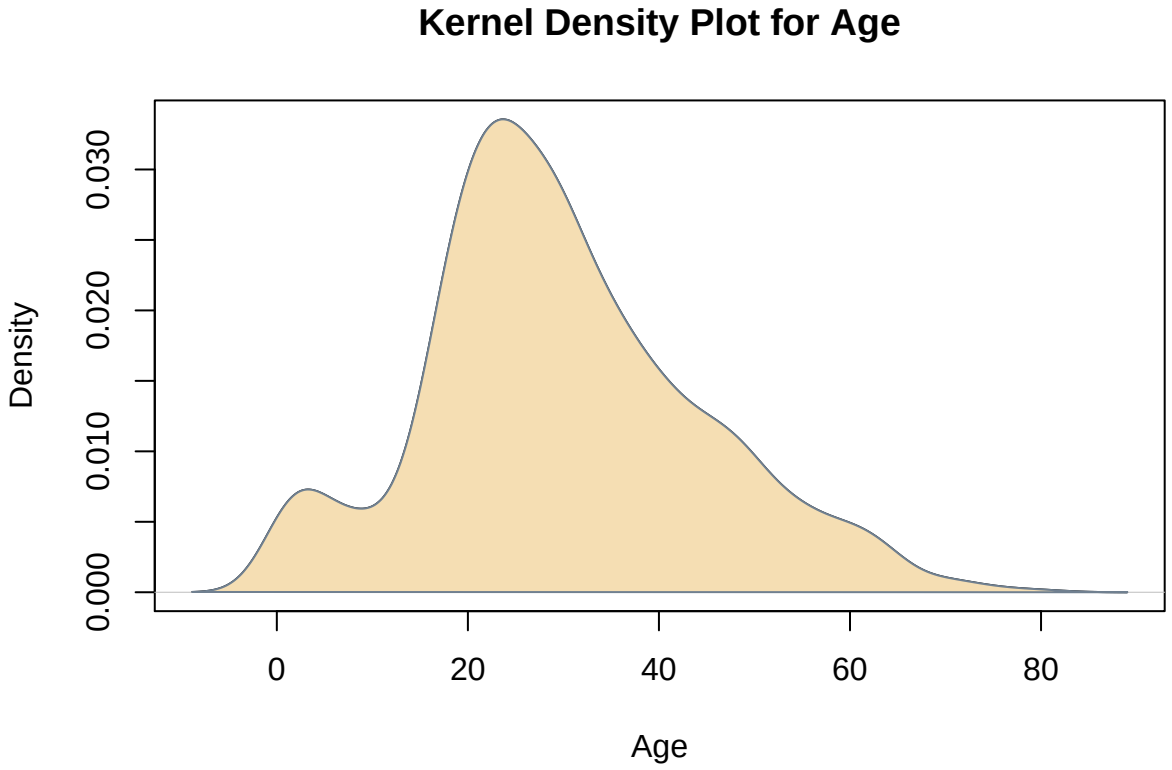



```
par(opar)
```

Another option for quantitative data is the kernel density plot. First we compute the density, then plot it. This plot gives you similar information to the histogram, but smoothing is applied.

In order to color the plot, we make a polygon in the last line below.

```
d <- density(df$age, na.rm = TRUE)
plot(d, main="Kernel Density Plot for Age", xlab="Age")
polygon(d, col="wheat", border="slategrey")
```



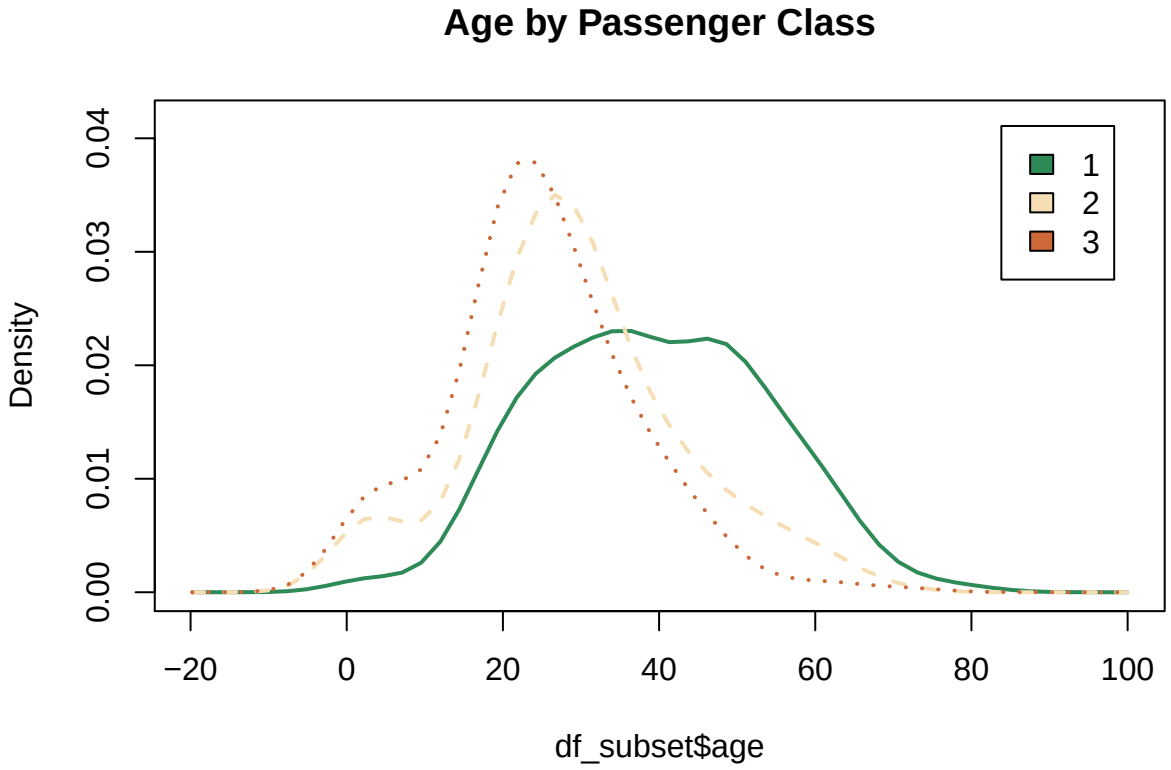
We can overlay several kernel density plots using package sm. First we subset the data frame to just be the two columns of interest so that we can use complete.cases() to get rid of NAs.

```
library(sm)
```

```
## Warning in fun(libname, pkgname): couldn't connect to display ":0"
```

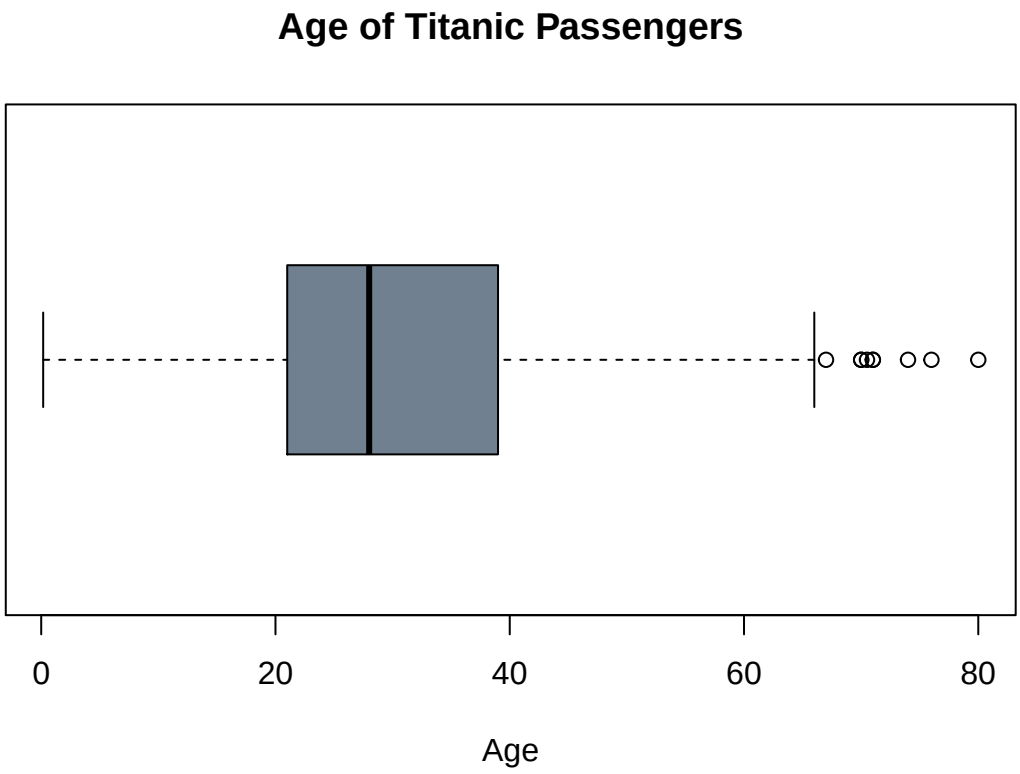
```
## Package 'sm', version 2.2-5.6: type help(sm) for summary information
```

```
df_subset <- df[,c(1,5)]
df_subset <- df_subset[complete.cases(df_subset),]
sm.density.compare(df_subset$age, df_subset$pclass, col=c("seagreen", "wheat", "sienna3"), lwd=2)
title(main="Age by Passenger Class")
legend("topright", inset=0.05, legend=c(1:3), fill=c("seagreen", "wheat", "sienna3"))
```



We can create a boxplot for a single quantitative variable. Here we made it horizontal. The box shows the 2nd and 3rd quartiles of the data. The "whiskers" at either end of the dashed lines show the 1st and 4th quartiles. Dots beyond a whisker indicate suspected outliers. The bold line through the box indicates the median.

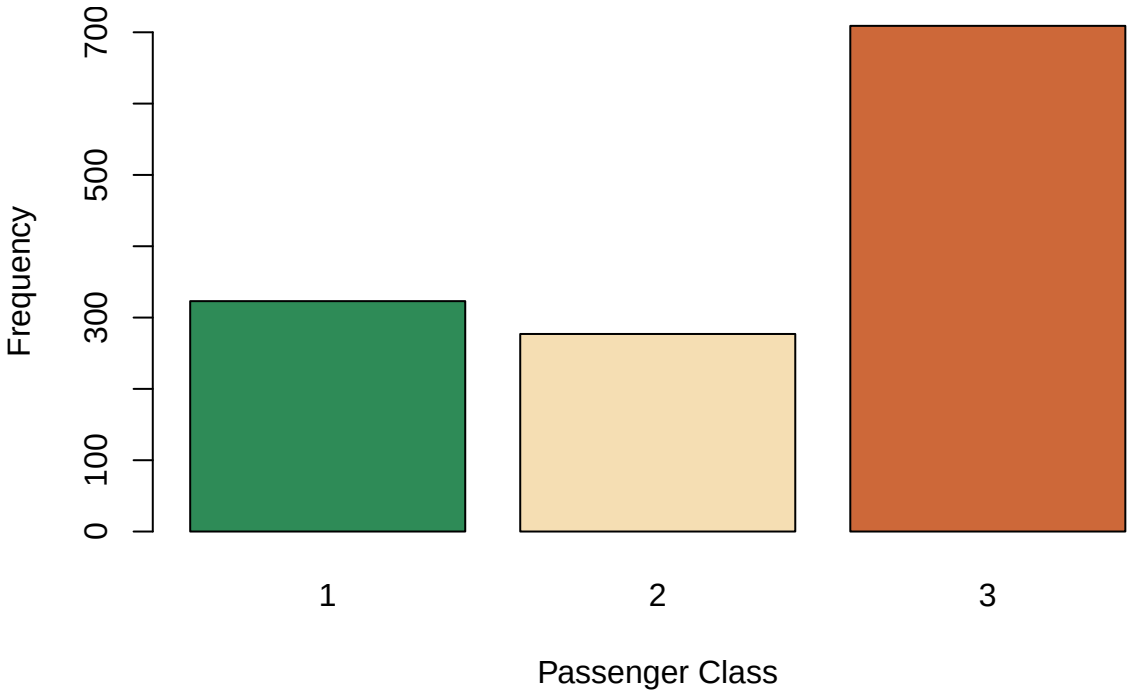
```
boxplot(df$age, col="slategray", horizontal=TRUE, xlab="Age", main="Age of Titanic Passengers")
```



Plotting one dimension, X is qualitative

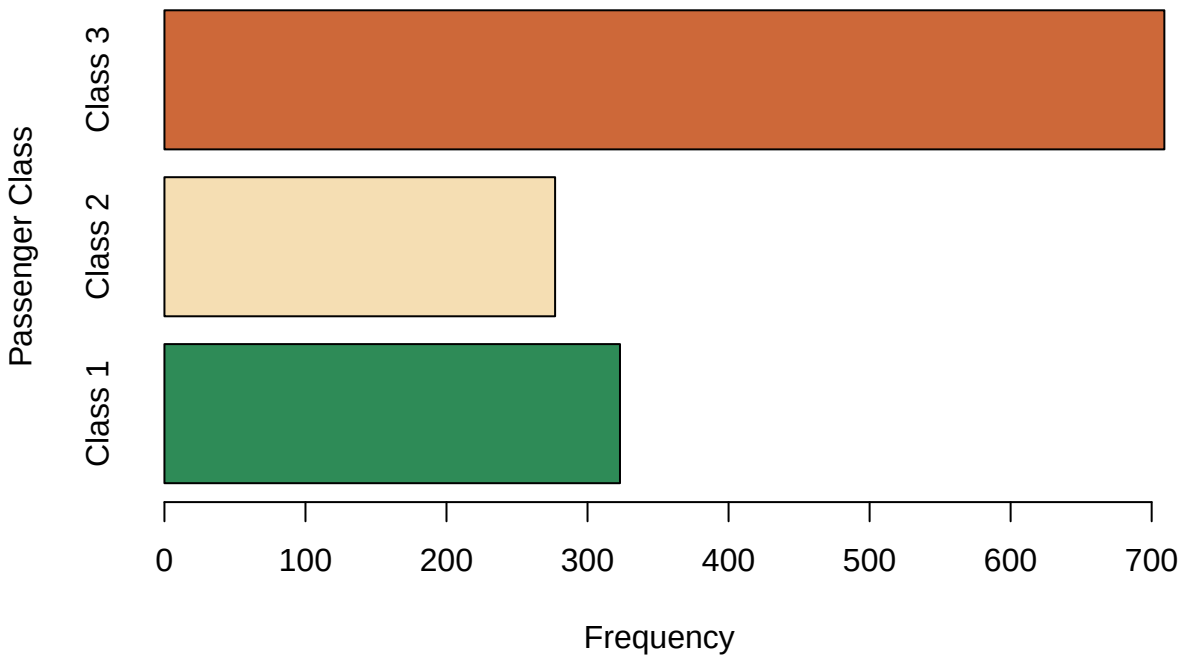
Barplots can be used for qualitative data. The can be vertical or horizontal.

```
counts <- table(df$pclass)
barplot(counts, xlab="Passenger Class", ylab="Frequency", col=c("seagreen","wheat","sienna3"))
```



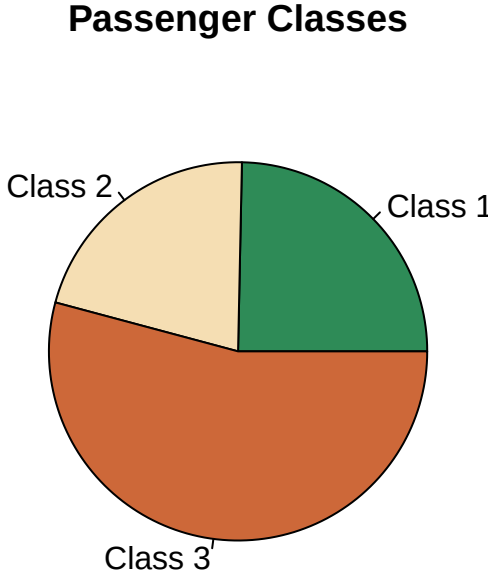
Here is the same plot, but with horizontal bars.

```
counts <- table(df$pclass)
barplot(counts, horiz=TRUE, names=c("Class 1", "Class 2", "Class 3"), col=c("seagreen","wheat","sienna3
```



A pie chart can be made with relative frequencies of a quantitative variable. First we specify frequencies for each of the 3 classes, then supply labels. With slices and labels defined, we can make a pie chart.

```
slices <- c(sum(df$pclass==1, na.rm = TRUE), sum(df$pclass==2, na.rm = TRUE), sum(df$pclass==3, na.rm =
lbls <- c("Class 1", "Class 2", "Class 3")
pie(slices, labels=lbls, main="Passenger Classes", col=c("seagreen","wheat","sienna3"))
```



‘‘