Notes2_Practice

Oxide Example

Oxide Layer Thicknesses Example: (f):

```
Thickness<-c(90.0, 92.2, 94.9, 92.7, 91.6, 88.2, 92.0, 98.2, 96.0, 91.1, 89.8, 91.5, 91.5, 90.6, 93.1, 8
Thickness
```

```
##  [1] 90.0 92.2 94.9 92.7 91.6 88.2 92.0 98.2 96.0 91.1 89.8 91.5 91.5 90.6 93.1
## [16] 88.9 92.5 92.4
```

```
length(Thickness)
```

```
## [1] 18
```

```
sort(Thickness)
```

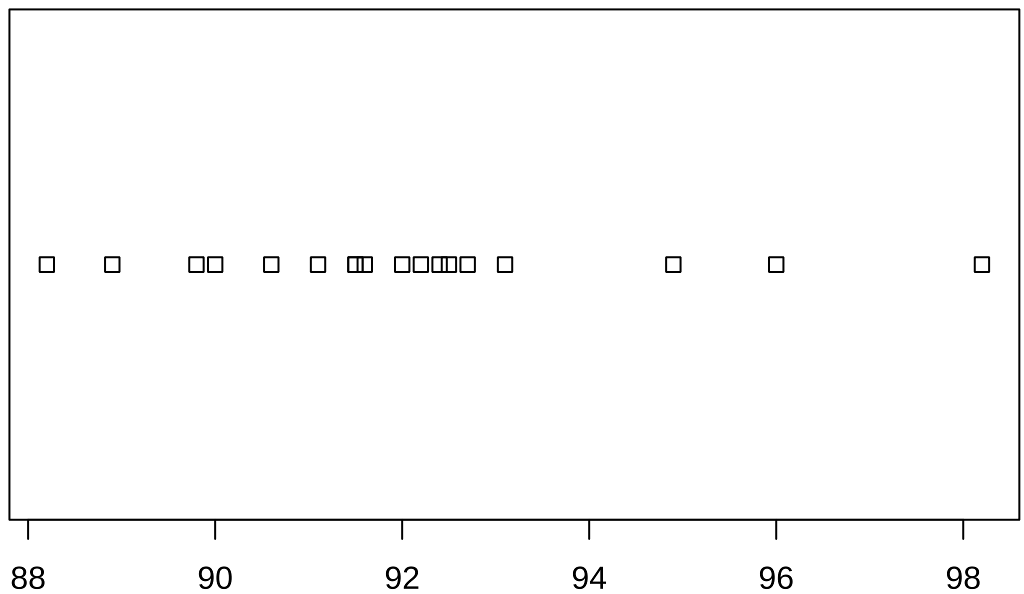
```
##  [1] 88.2 88.9 89.8 90.0 90.6 91.1 91.5 91.5 91.6 92.0 92.2 92.4 92.5 92.7 93.1
## [16] 94.9 96.0 98.2
```

```
Thickness<-sort(Thickness, decreasing=FALSE)
#Thickness <-sort(Thickness , decreasing = TRUE)
Thickness
```

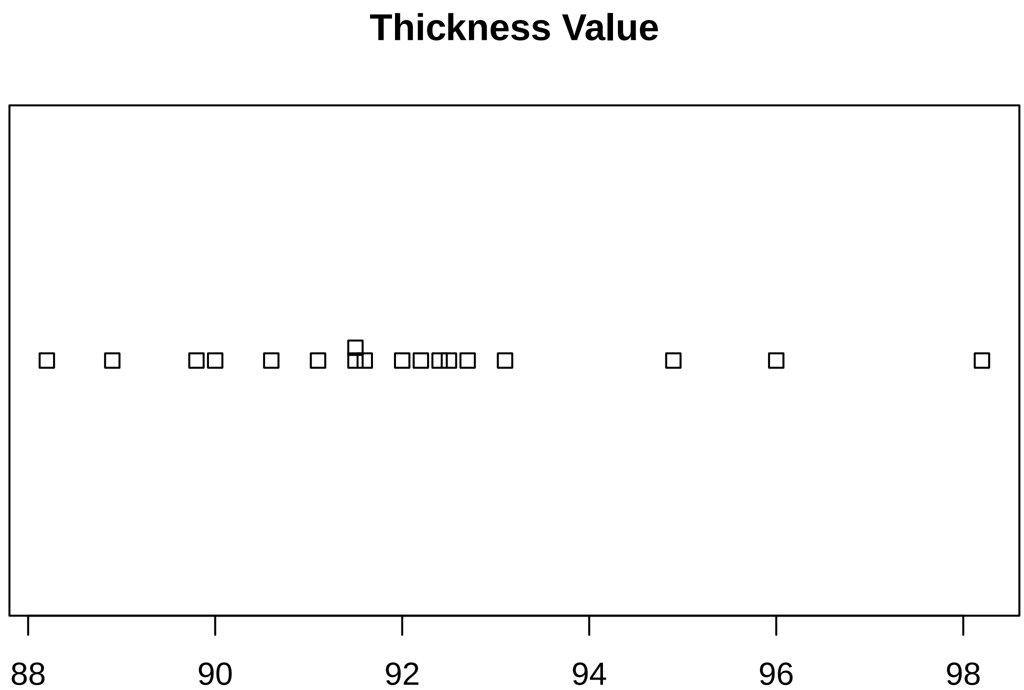
```
##  [1] 88.2 88.9 89.8 90.0 90.6 91.1 91.5 91.5 91.6 92.0 92.2 92.4 92.5 92.7 93.1
## [16] 94.9 96.0 98.2
```

Oxide Layer Thicknesses Example (g): : Construct a dot plot, frequency histogram, relative frequency histogram, and density histogram for the oxide layer thicknesses data

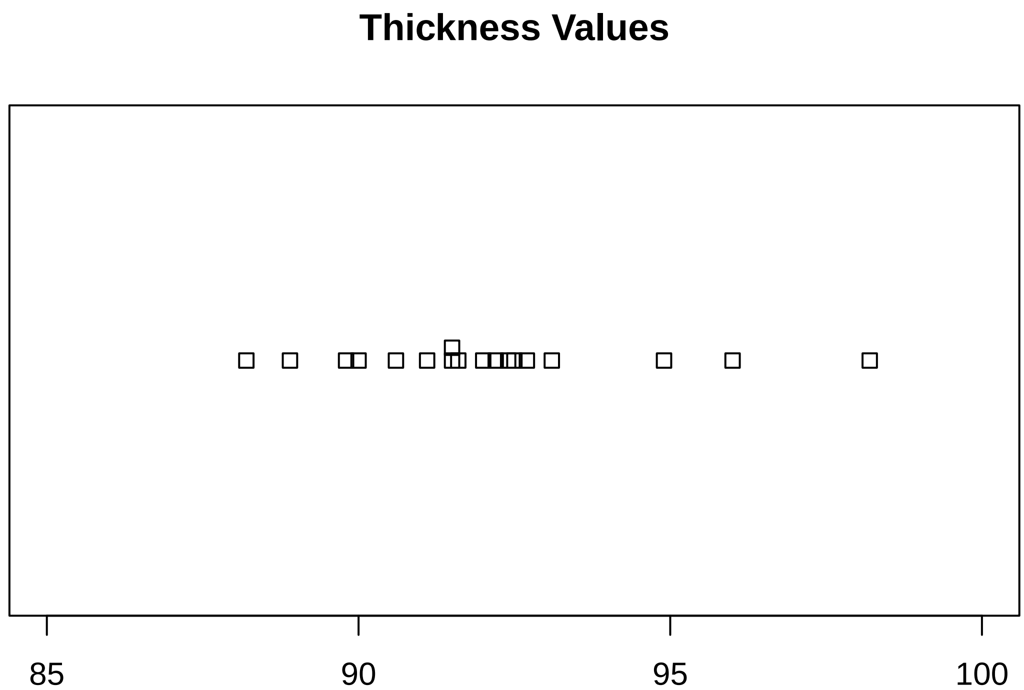
```
#Dotplot
stripchart(Thickness)
```



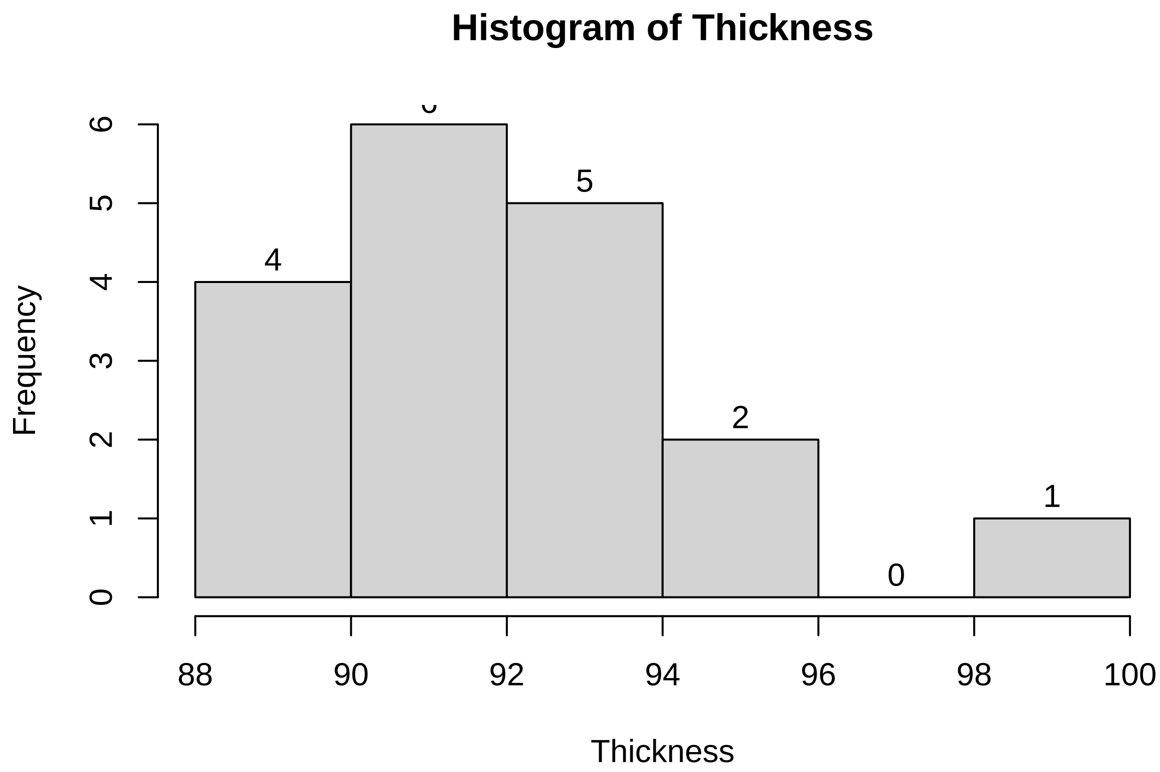

```r
stripchart(Thickness, method="stack", main="Thickness Value")
```




```
stripchart(Thickness, method="stack", main="Thickness Values", xlim=c(85, 100))
```




```
#default hist is frequency and bins set for us
hist(Thickness)
hist(Thickness, labels=TRUE)
```

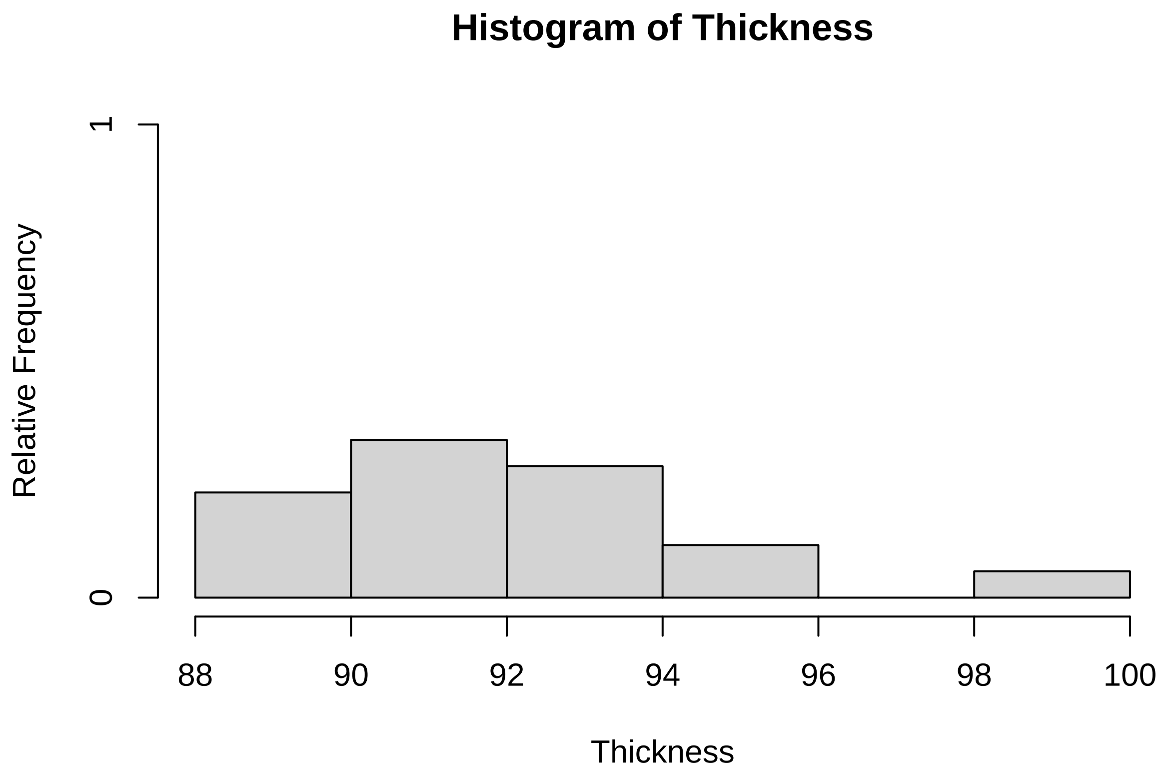



```r
hist(Thickness, labels=TRUE, ylim=c(0,7))
```

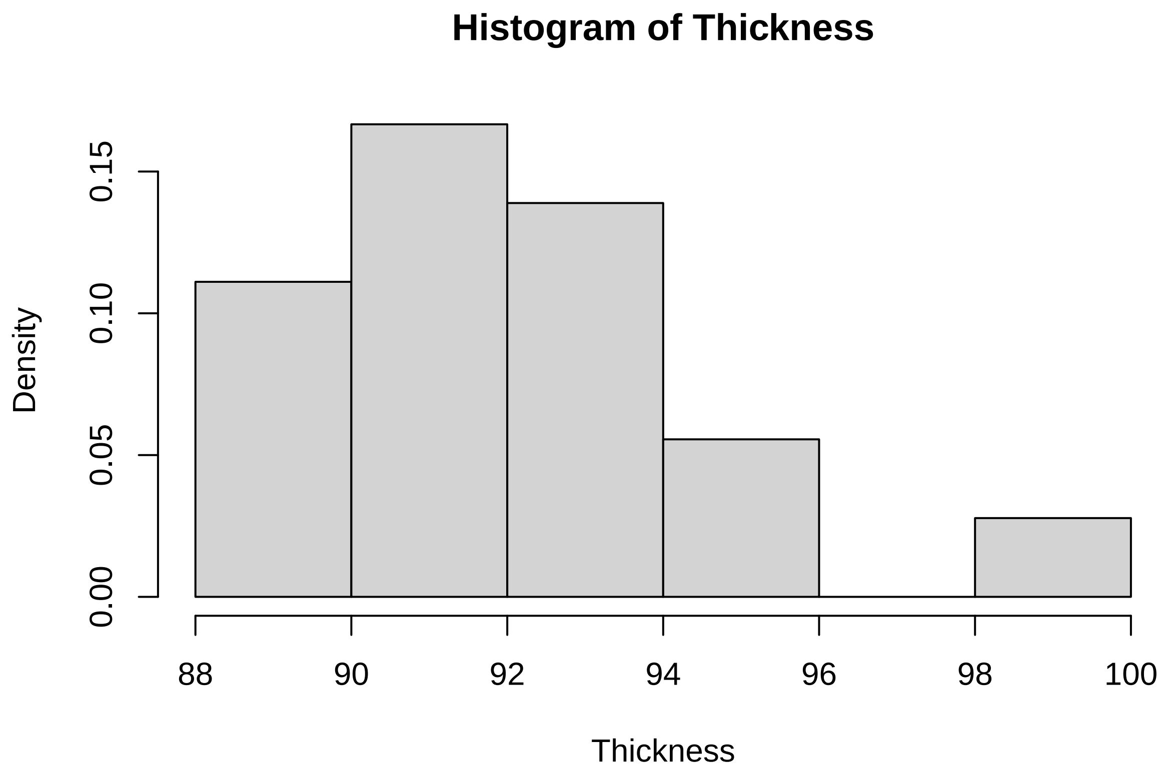

```r
hist(Thickness, breaks=seq(86,100,2), labels=TRUE, ylim=c(0,10))
```


```
hist(Thickness, breaks=seq(86,100,1), labels=TRUE, ylim=c(0,6))
```


```
#Relative freq hist
h<-hist(Thickness, plot=FALSE)
h$counts<-h$counts/length(Thickness)  #note: length(Thickness) is 18
plot(h, ylab="Relative Frequency", labels=FALSE, ylim=c(0,1))
```




```
#Density histogram will become more useful when we have data sets with a larger number of unique observ
hist(Thickness, freq=FALSE)
```



Oxide Layer Thicknesses Example (i): Compute and compare the sample Mode, Mean, and Median of the observed thicknesses

```
mean(Thickness) #92.06667
```

```
## [1] 92.06667
```

```
sum(Thickness)/length(Thickness)  #1657.2/18 long hand
```

```
## [1] 92.06667
```

```
median(Thickness)  #91.8
```

```
## [1] 91.8
```

Oxide Layer Thicknesses (j): Compute the 1st, 2nd, and 3rd quartiles for thickness data. Interpret these values.

```
quantile(Thickness, probs=c(.25, .5, .75))
```

```
##    25%    50%    75%
## 90.725 91.800 92.650
```

```r
quantile(Thickness, probs=c(0.25, .50, .75), type=2)  #type 7 is default, type=2 is SAS default
```

```
##  25%  50%  75%
## 90.6 91.8 92.7
```

Oxide Layer Thicknesses Example (k): Compute and compare the range, IQR and Standard deviation for the thickness data. Interpret these values.

```r
range(Thickness)
```

```
## [1] 88.2 98.2
```

```r
IQR(Thickness, type=2)
```

```
## [1] 2.1
```

```r
quantile(Thickness, 0.75, type=2, names=FALSE)-quantile(Thickness, 0.25, type=2, names=FALSE)
```

```
## [1] 2.1
```

```r
mean(Thickness)
```

```
## [1] 92.06667
```

```r
sum((Thickness-mean(Thickness))^2) #sum of squared deviations
```

```
## [1] 101.24
```

```r
sqrt(sum((Thickness-mean(Thickness))^2)/17) #sample sd "long hand"
```

```
## [1] 2.440347
```

```r
sqrt(sum((Thickness-mean(Thickness))^2)/(length(Thickness)-1)) #sample sd "long hand"
```

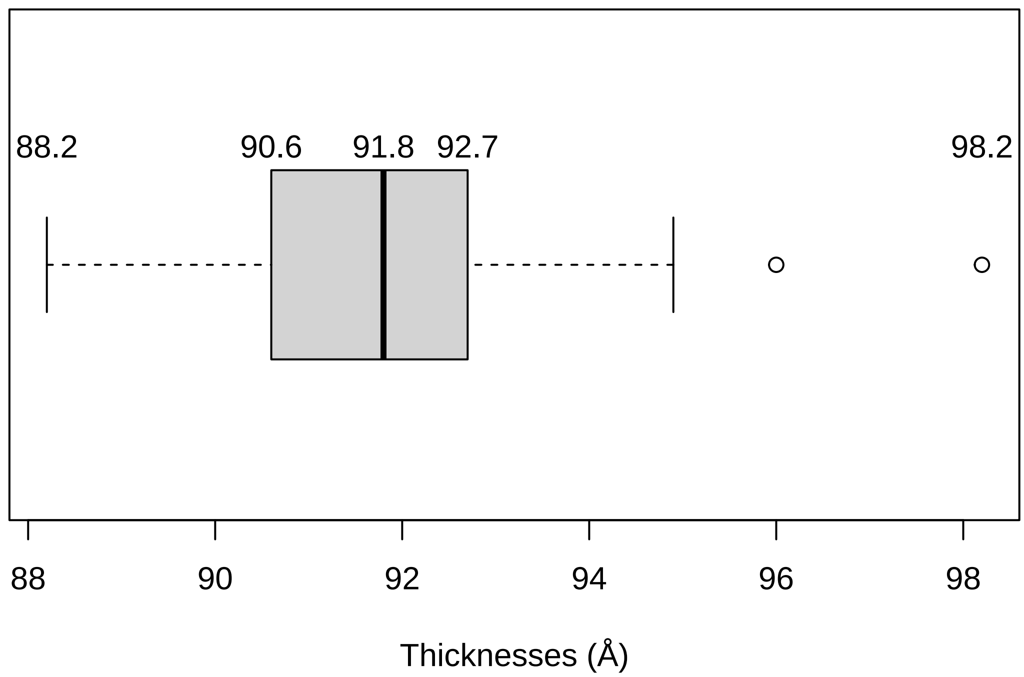
```
## [1] 2.440347
```

```r
sd(Thickness) #sample sd
```

```
## [1] 2.440347
```

Oxide Layer Thicknesses Example (l): Construct a boxplot for the thickness data

```r
boxplot(Thickness, horizontal=TRUE, xlab="Thicknesses (Å)")
text(x=fivenum(Thickness), labels=fivenum(Thickness), y=1.25)
```




```
quantile(Thickness, type=2)
```

```
##   0%  25%  50%  75% 100%
## 88.2 90.6 91.8 92.7 98.2
```

Oxide Layer Thicknesses m: Suppose the 98.2 had actually been recorded as 99.2. So the thickness values with the error are: 90.0, 92.2, 94.9, 92.7, 91.6, 88.2, 92.0, 99.2, 96.0, 91.1, 89.8, 91.5, 91.5, 90.6, 93.1, 88.9, 92.5, 92.4. How would Mean, Median, Range, IQR, and sample SD be affected by the error? How much would their value[s] change? How would the graphical summaries change?

```
Thickness_Error<-c(90.0, 92.2, 94.9, 92.7, 91.6, 88.2, 92.0, 99.2, 96.0, 91.1, 89.8, 91.5, 91.5, 90.6, 9
# or by copying and editing the original vector
Thickness_Error2<-Thickness
Thickness_Error2[18]<-99.2
Thickness_Error2
```

```
##  [1] 88.2 88.9 89.8 90.0 90.6 91.1 91.5 91.5 91.6 92.0 92.2 92.4 92.5 92.7 93.1
## [16] 94.9 96.0 99.2
```

```
mean(Thickness_Error) ; mean(Thickness_Error2)
```

```
## [1] 92.12222
```

```
## [1] 92.12222
```

```r
sum(Thickness_Error)  ; sum(Thickness_Error2)
```

```
## [1] 1658.2
```

```
## [1] 1658.2
```

```r
median(Thickness_Error)
```

```
## [1] 91.8
```

```r
quantile(Thickness_Error, type=2)
```

```
##   0%  25%  50%  75% 100%
## 88.2 90.6 91.8 92.7 99.2
```

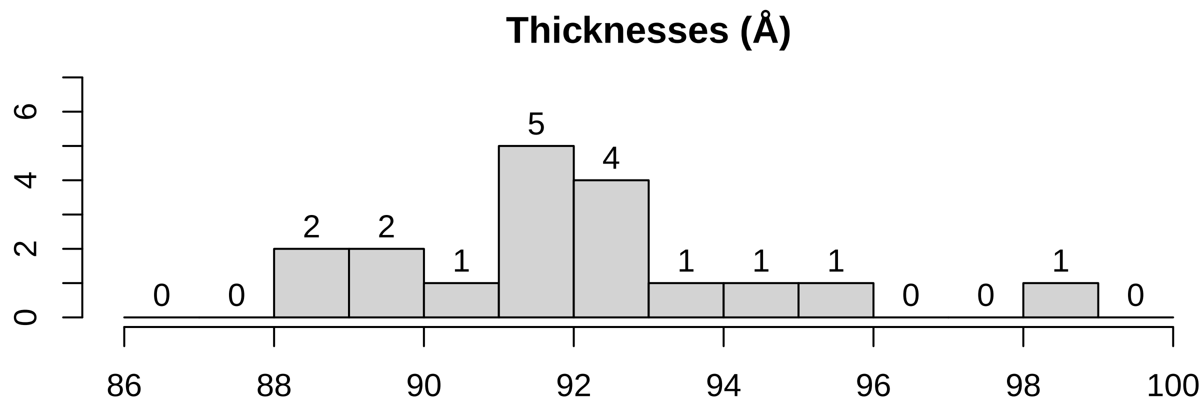
```r
range(Thickness_Error)
```

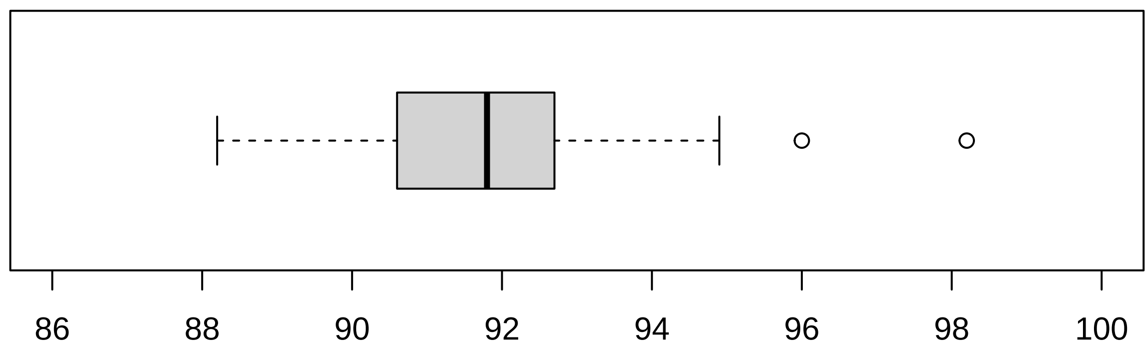
```
## [1] 88.2 99.2
```

```r
sd(Thickness_Error)
```

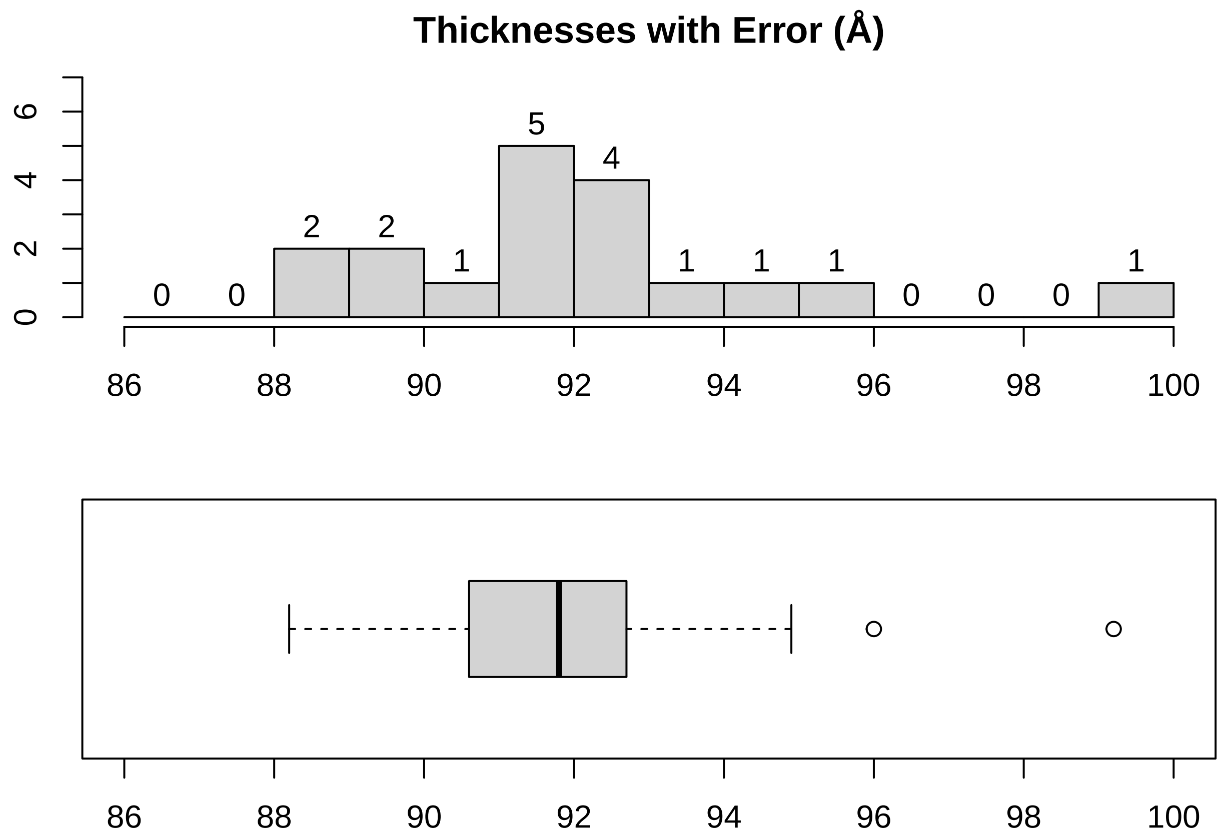
```
## [1] 2.59469
```

```r
#Graphs of correct data
par(mfrow=c(2,1), mar=c(2.5,2,2,1)) #This makes two rows of graphs in 1 column and sets the margins

hist(Thickness, labels=TRUE, ylim=c(0,7),
     breaks=seq(86,100,1), main="Thicknesses (Å)")
boxplot(Thickness, horizontal=TRUE, ylim=c(86,100))
```





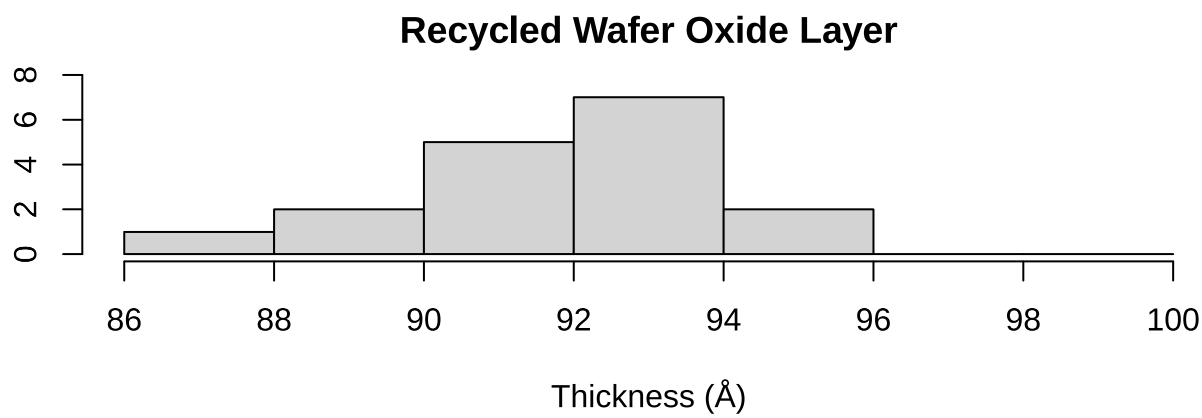

```r
#Graphs of data with error
hist(Thickness_Error, labels=TRUE, ylim=c(0,7),
     breaks=seq(86,100,1), main="Thicknesses with Error (Å)")
boxplot(Thickness_Error, horizontal=TRUE, ylim=c(86,100))
```

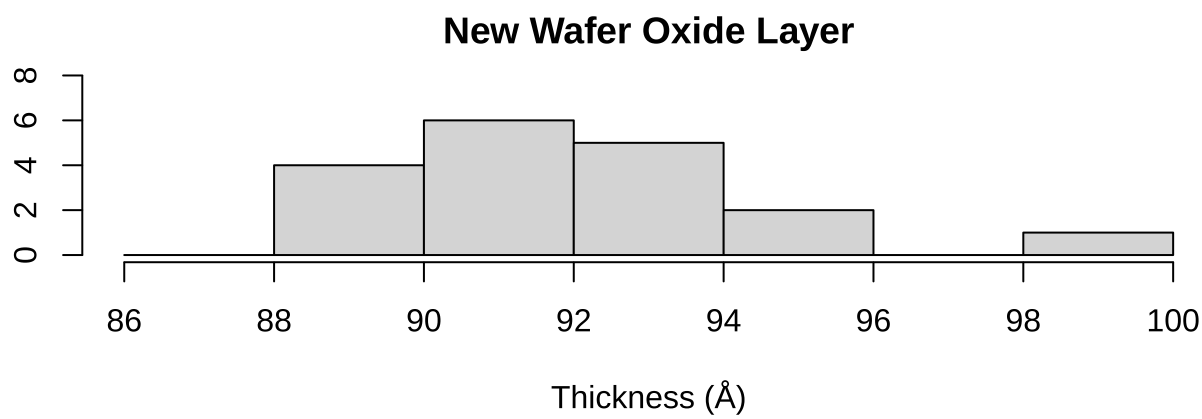



```r
par(mfrow=c(1,1), mar=c(5, 4.1, 4.1, 2.1)) #This resets the graphics window to 1 graph at a time and re
```

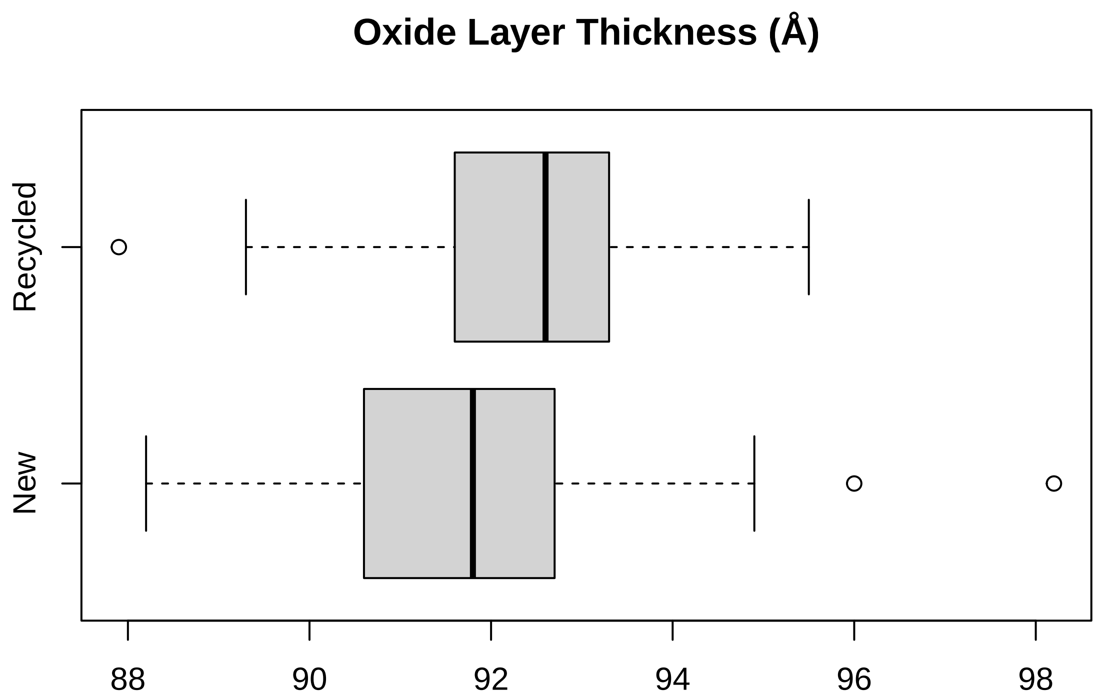
Thickness Example (n): comparing 2 samples

```r
Thickness_new<-Thickness
Thickness_rec<-c(91.8, 94.5, 93.9, 92.0, 89.9, 87.9, 92.8, 93.3, 92.6, 90.3, 92.8, 91.6, 92.7, 91.7, 89

par(mfrow=c(2,1), mar=c(4.2,2,2,1))

hist(Thickness_rec, ylim=c(0,8),
     breaks=seq(86,100,2), main="Recycled Wafer Oxide Layer",
     xlab="Thickness (Å)")

hist(Thickness_new, ylim=c(0,8),
     breaks=seq(86, 100, 2), main="New Wafer Oxide Layer",
     xlab="Thickness (Å)")
```






```r
par(mfrow=c(1,1), mar=c(5.1, 4.1, 4.1, 2.1))


boxplot(Thickness_new,Thickness_rec, horizontal=TRUE,
        names=c("New", "Recycled"),
        main="Oxide Layer Thickness (Å)")
```




```r
mean(Thickness_new); mean(Thickness_rec)
```

```
## [1] 92.06667

## [1] 92.12941
```

```r
median(Thickness_new); median(Thickness_rec)
```

```
## [1] 91.8

## [1] 92.6
```

```r
sd(Thickness_new); sd(Thickness_rec)
```

```
## [1] 2.440347

## [1] 1.934801
```

```r
IQR(Thickness_new, type=2); IQR(Thickness_rec, type=2)
```

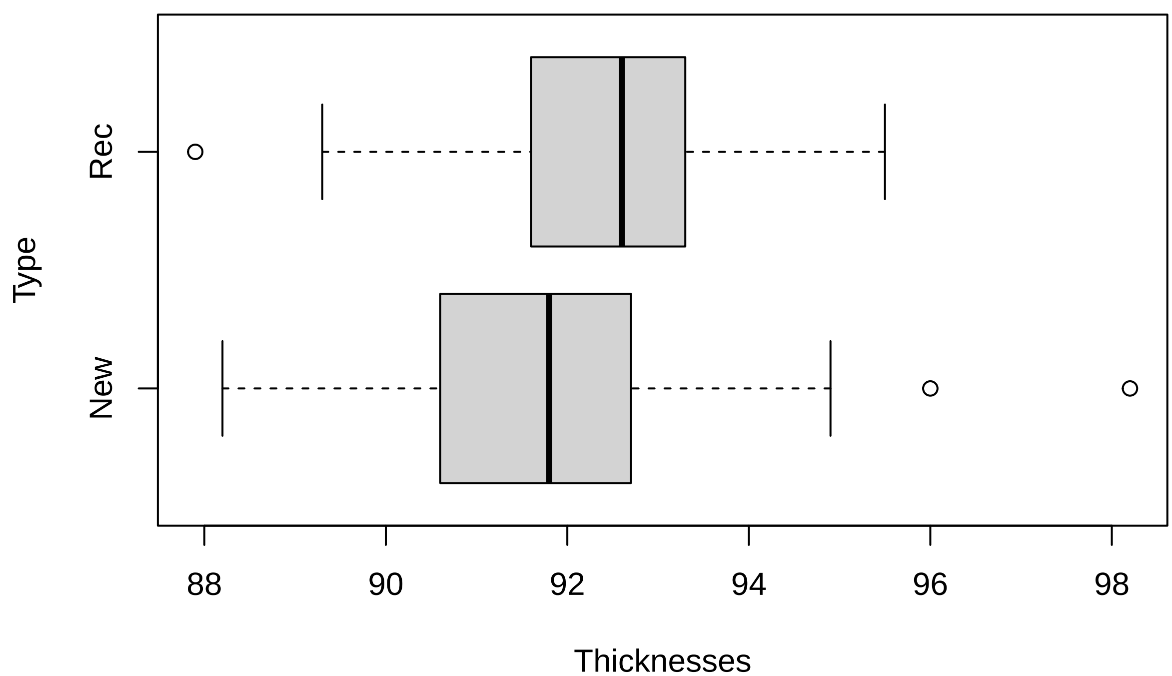
```
## [1] 2.1
```

```
## [1] 1.7
```

If Time: Thickness data from .csv file

(1) Navigate to folder that holds .Rmd and .csv file in Files pane
(2) Set folder in Files pane as working directory
(3) Read data into data frame using read.csv() command

```r
#Thicknesses<-c(Thickness_new, Thickness_rec)
#Type=c(rep("New", times=18), rep("Rec", 17))
#Thickness_data<-data.frame(Type, Thicknesses)
#write.csv(Thickness_data, "Thickness_Data.csv")

Thickness_Data<-read.csv("Thickness_Data.csv")


#boxplot code works nicely
boxplot(Thicknesses~Type, data=Thickness_Data,
        horizontal=TRUE)
```




```r
#other summaries and graphs in base R require you save off the vectors of data separately

#this gets a dataframe and then vector of thicknesses for New Wafers
New_df<-subset(Thickness_Data, Type=="New")
Thick_new<-New_df$Thicknesses
mean(Thick_new); sd(Thick_new)
```

```
## [1] 92.06667

## [1] 2.440347
```

```
#this gets a dataframe and then vector of thicknesses for Rec Wafers
Rec_df<-subset(Thickness_Data, Type=="Rec")
Thick_rec<-Rec_df$Thicknesses
mean(Thick_rec); sd(Thick_rec)
```

```
## [1] 92.12941

## [1] 1.934801
```

> Oxide Thickness example o: Summarize the observed sample according to the variable "Oxide Thickness Less than 90"

```
Thickness<90
```

```
##  [1]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE
```

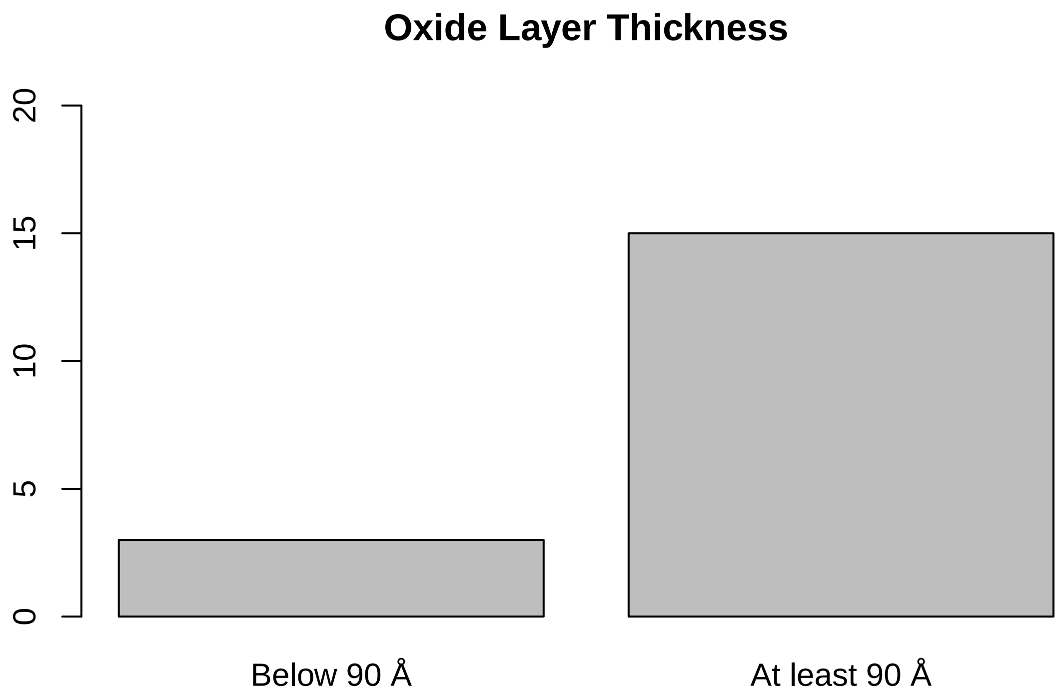
```
sum(Thickness<90)  #3 Trues (1) 15 Falses (0)
```

```
## [1] 3
```

```
sum(Thickness>=90)
```

```
## [1] 15
```

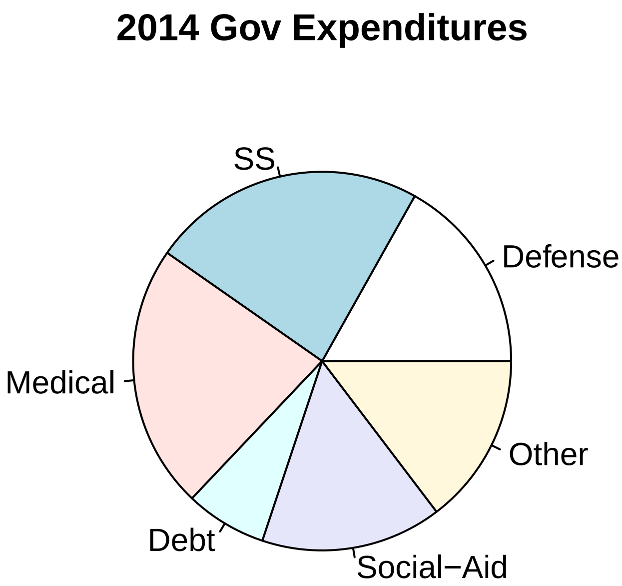
```
barplot(height=c(3,15), names=c("Below 90 Å", "At least 90 Å"), ylim=c(0,20), main="Oxide Layer Thickne
```



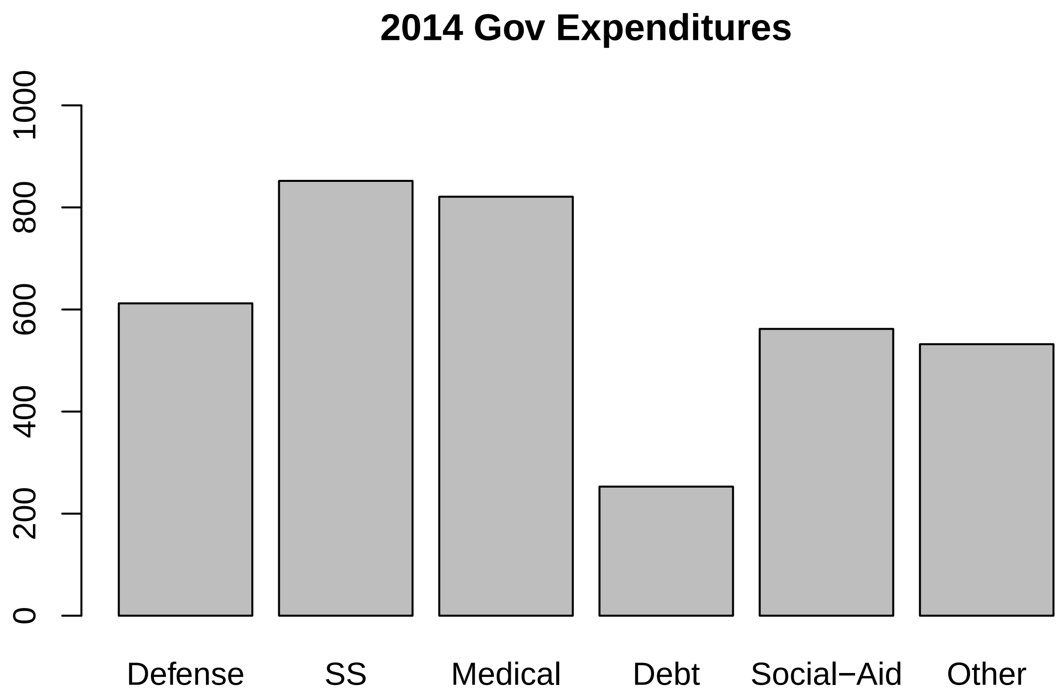
Goverment Spending

Gov Spending pie chart and bar chart.

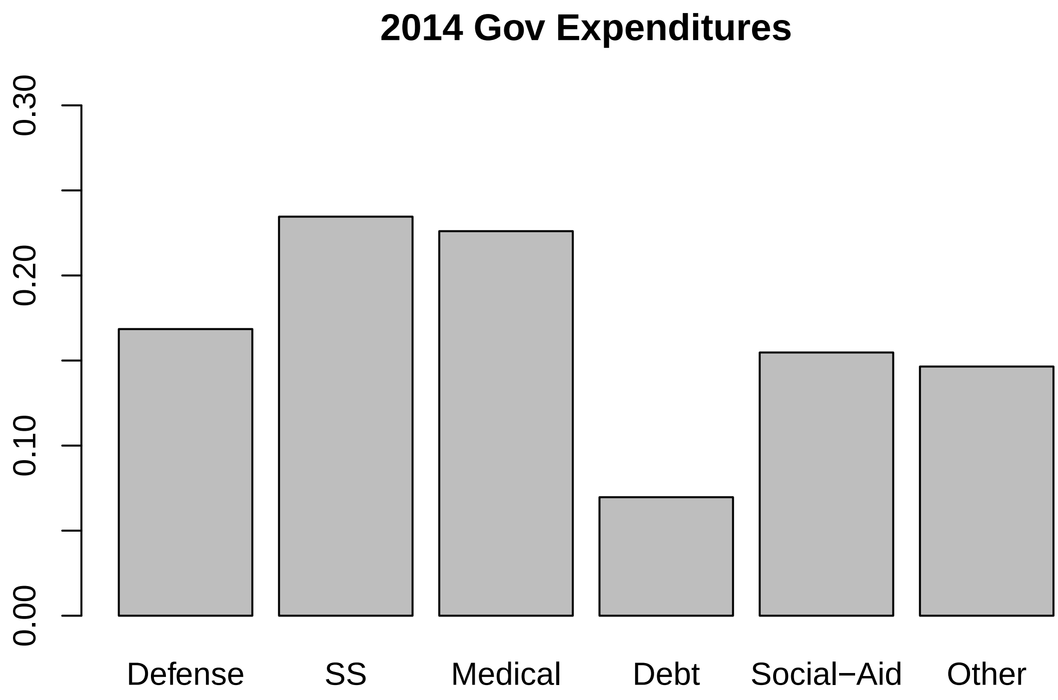
```
Dollars<-c(612, 852, 821, 253, 562, 532)
#Pie Chart
pie(Dollars, labels=c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Other"), main="2014 Gov Expendi
```



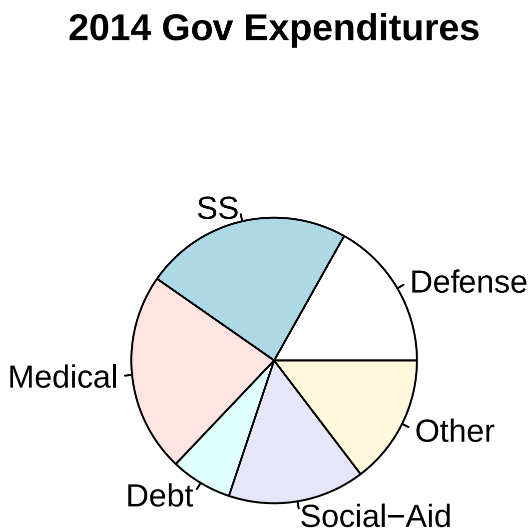

```
#BarPlot
barplot(height=Dollars, names.arg=c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Other"), main="20
```

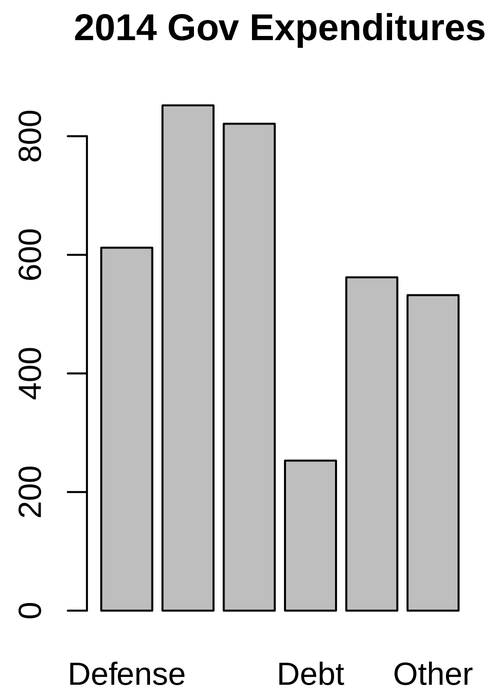



```r
barplot(height=Dollars/sum(Dollars), names.arg=c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Oth
```




```
par(mfrow=c(1,2))
pie(Dollars, labels=c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Other"), main="2014 Gov Expendi

barplot(height=Dollars, names.arg=c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Other"), main="20
```



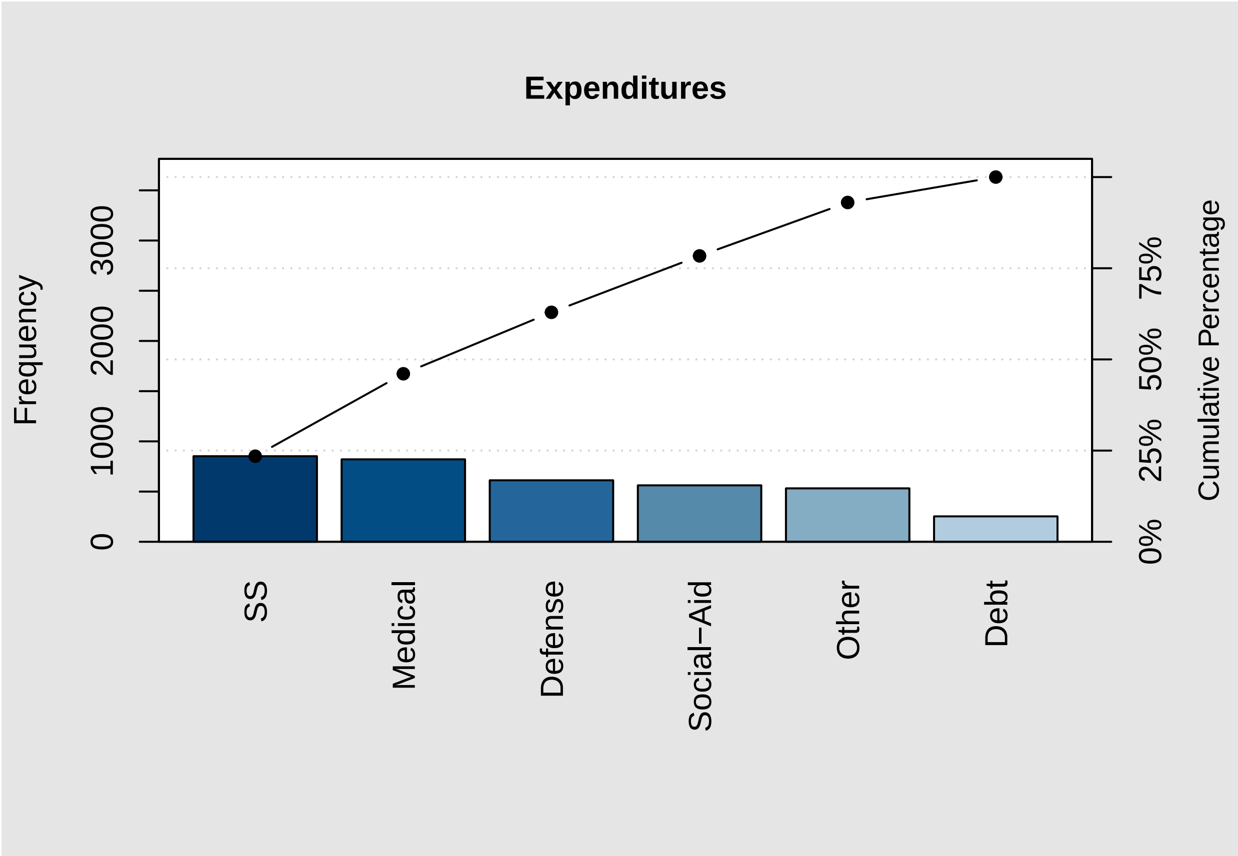



```r
par(mfrow=c(1,1))

#Pareto Chart
#Dollars<-c(612, 852, 821, 253, 562, 532)
names(Dollars)<-c("Defense", "SS", "Medical", "Debt", "Social-Aid", "Other")

#Load pareto.chart() function from qcc package
#install.packages("qcc")
library(qcc)
```

```
## Package 'qcc' version 2.7
```

```
## Type 'citation("qcc")' for citing this R package in publications.
```

```r
pareto.chart(Dollars, main="Expenditures")
```




```
##
## Pareto chart analysis for Dollars
##                Frequency   Cum.Freq.  Percentage Cum.Percent.
##   SS           852.000000  852.000000   23.458150    23.458150
##   Medical      821.000000 1673.000000   22.604626    46.062775
##   Defense      612.000000 2285.000000   16.850220    62.912996
##   Social-Aid   562.000000 2847.000000   15.473568    78.386564
##   Other        532.000000 3379.000000   14.647577    93.034141
##   Debt         253.000000 3632.000000    6.965859   100.000000
```