Graphical Displays

Grant B. Morgan
BaylorUniversity

August 23, 2017

This post contains code for generating graphical displays that are commonly used with data collected from experiments. The code below uses an SPSS dataset.

Make sure to have these packages installed: `foreign, ggplot2, reshape, plyr`.

I read the data into an R object that I have called "namegame".

```r
# Read the data into R
library(foreign)
namegame <- read.spss("/Users/grant_morgan/Box Sync/Teaching/EDP 6360 - Experimental Design/Da
    to.data.frame = TRUE)
```

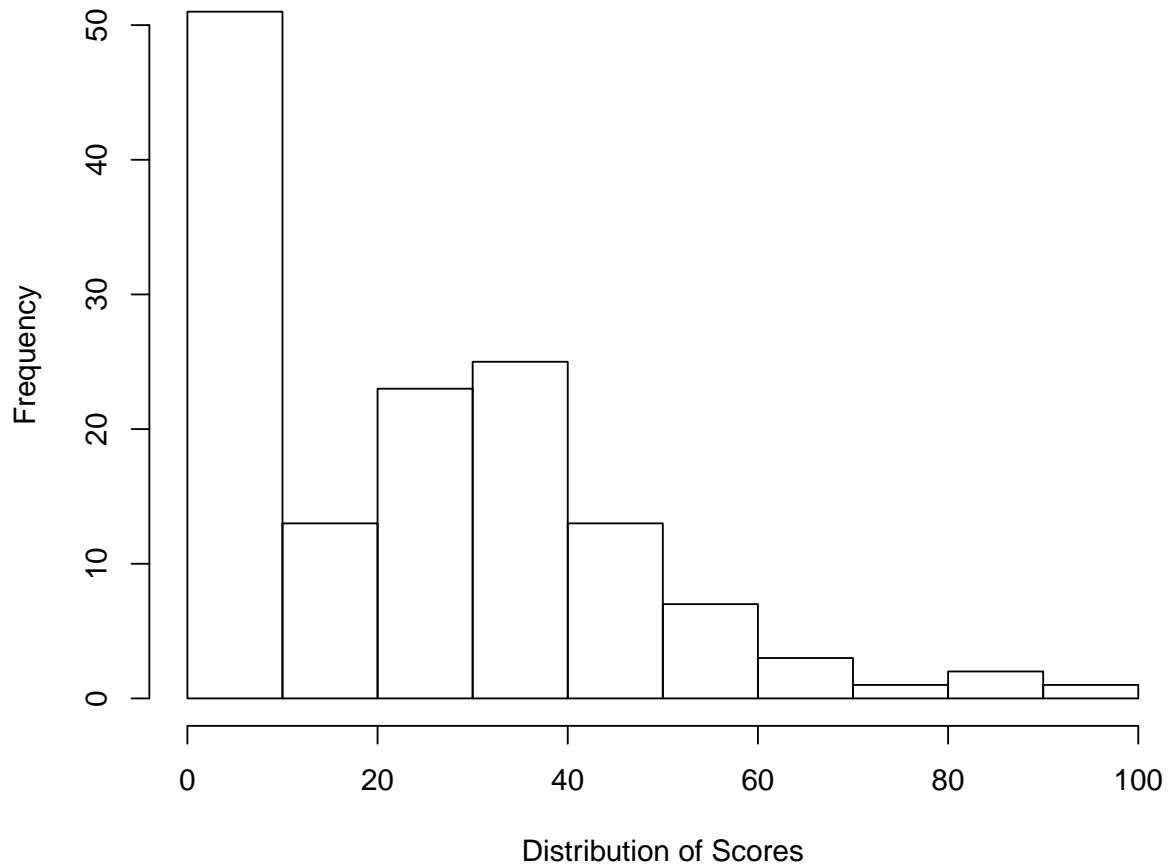Let's first take a look at the first few lines of the data set.

```r
head(namegame)
```

```
##   GROUP RECALL
## 1     1     24
## 2     1     43
## 3     1     38
## 4     1     65
## 5     1     35
## 6     1     15
```

Here I generate a histogram of the variable called "RECALL" in the "namegame" object. The dollar sign ($) is used to specify a variable within an object.

```r
hist(namegame$RECALL, main="Histogram of Scores",
    xlab="Distribution of Scores")
```
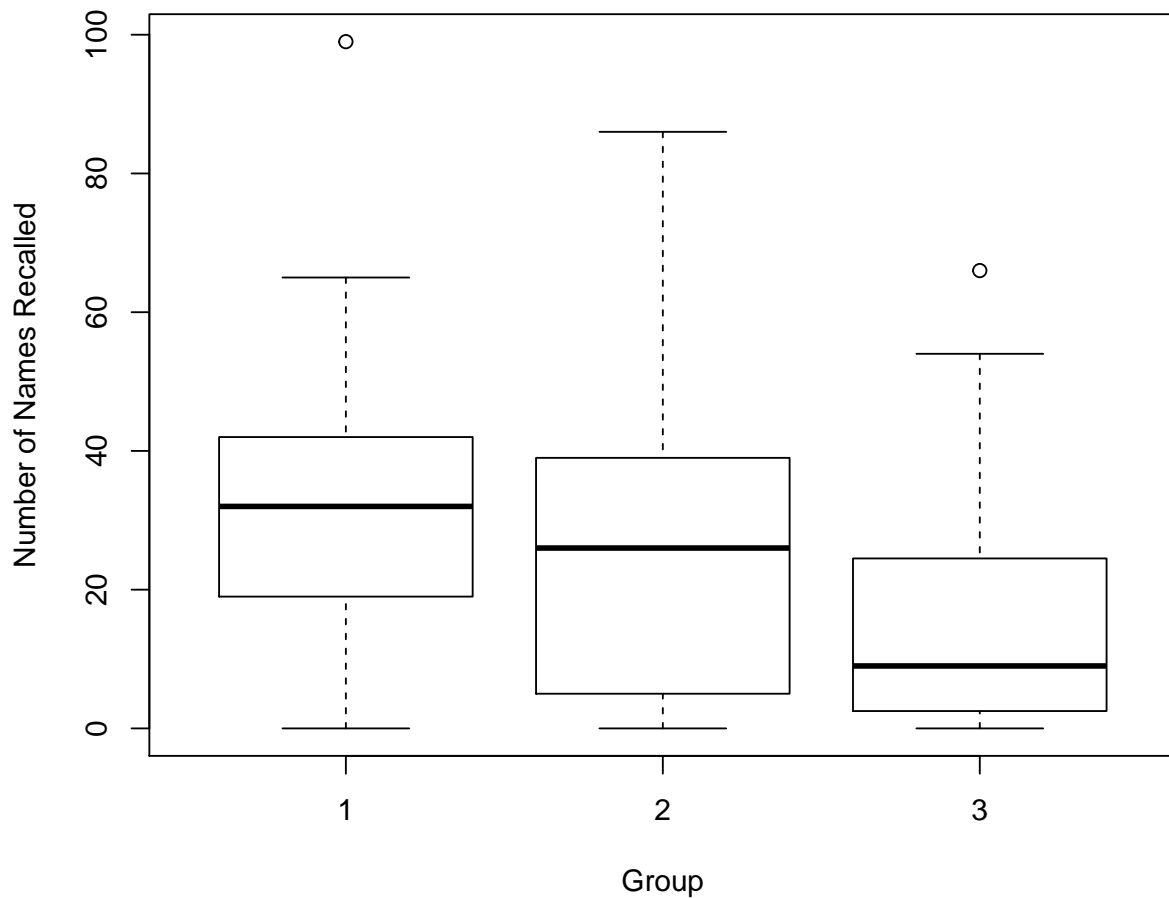
Histogram of Scores

Here I generate a side-by-side boxplot of the variable called "RECALL" in the "namegame" object for each "GROUP".

```
boxplot(namegame$RECALL ~ namegame$GROUP,
        data=namegame,
        main="Side-by-side Boxplot",
        xlab="Group",
        ylab="Number of Names Recalled")
```

Side–side Boxplot

Here I generate a line graphs of means of a variable across multiple time points.

```
library(ggplot2)
library(plyr)
library(reshape)
repeated<-read.spss("/Users/grant_morgan/Box Sync/Teaching/EDP 6360 - Experimental Design/Data/
        to.data.frame=TRUE)

repeat.long<-melt(repeated,
        id = "Person",
        measured=c("Time1", "Time2", "Time3", "Time4"))

names(repeat.long)<-c("Person", "Time", "Score")

t.plot <- ddply(repeat.long,
        .(Time),
        summarize,
        mean = mean(Score))
```

```
ggplot(data = t.plot, aes(x = Time, y = mean, group=1)) +
  geom_path() +
  geom_point(size = 3) +
  geom_line(size = 1) +
  ylab("Score") +
  xlab("Time")
```
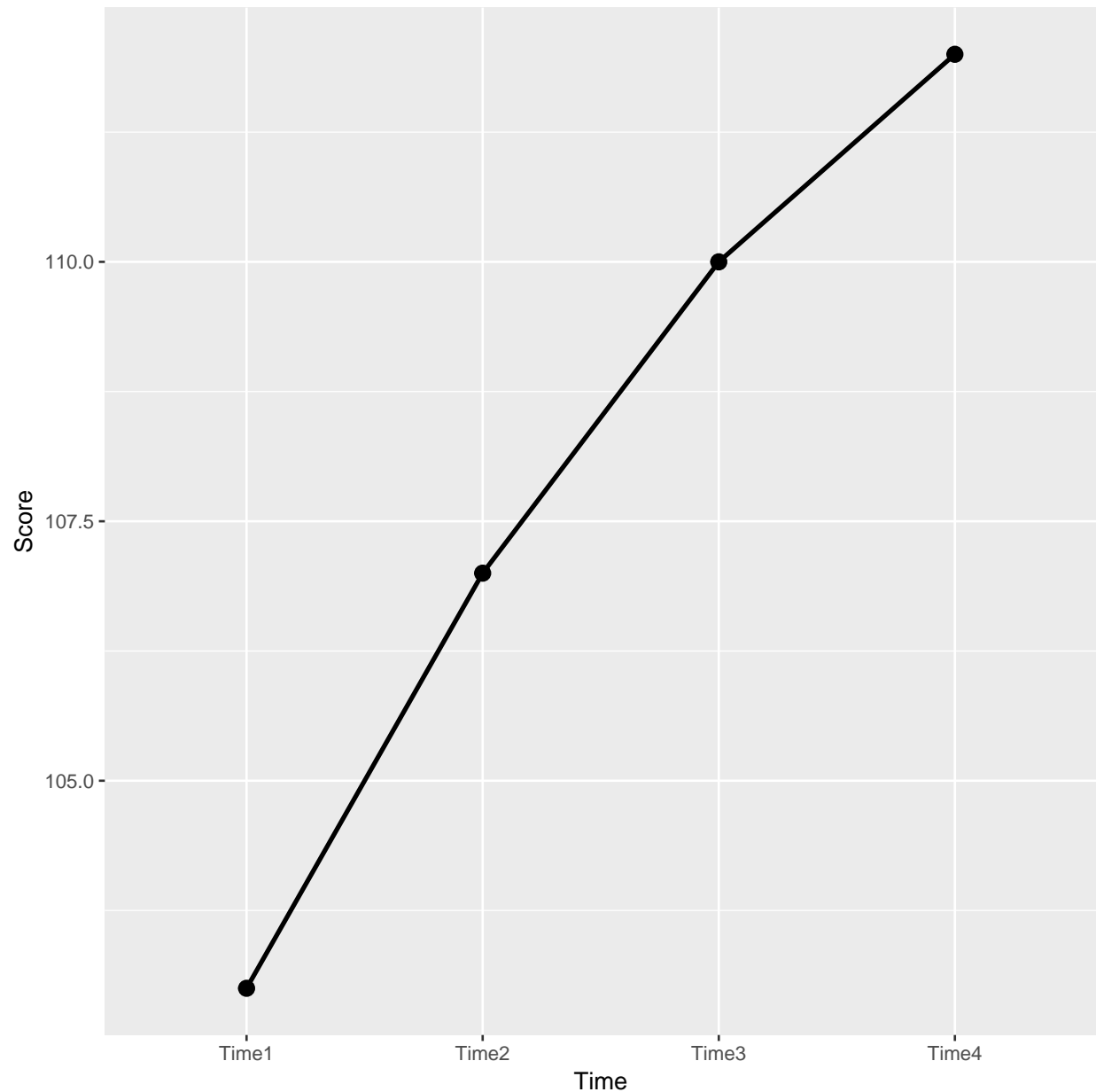

You may want to save the plots to your computer. It's easiest to do this with the png(), jpg(), or bmp(). Using the boxplot code above as an example, the plot can be saved in a png format, for example, using:

```
png("/Users/grant_morgan/Box Sync/Teaching/EDP 6360 - Experimental Design/Data/sbsboxplot.png")
        data=namegame,
        main="Side-by-side Boxplot",
        xlab="Group",
        ylab="Number of Names Recalled")
dev.off()
```