Boxplot Walkthrough in ggplot2

ISOM 675

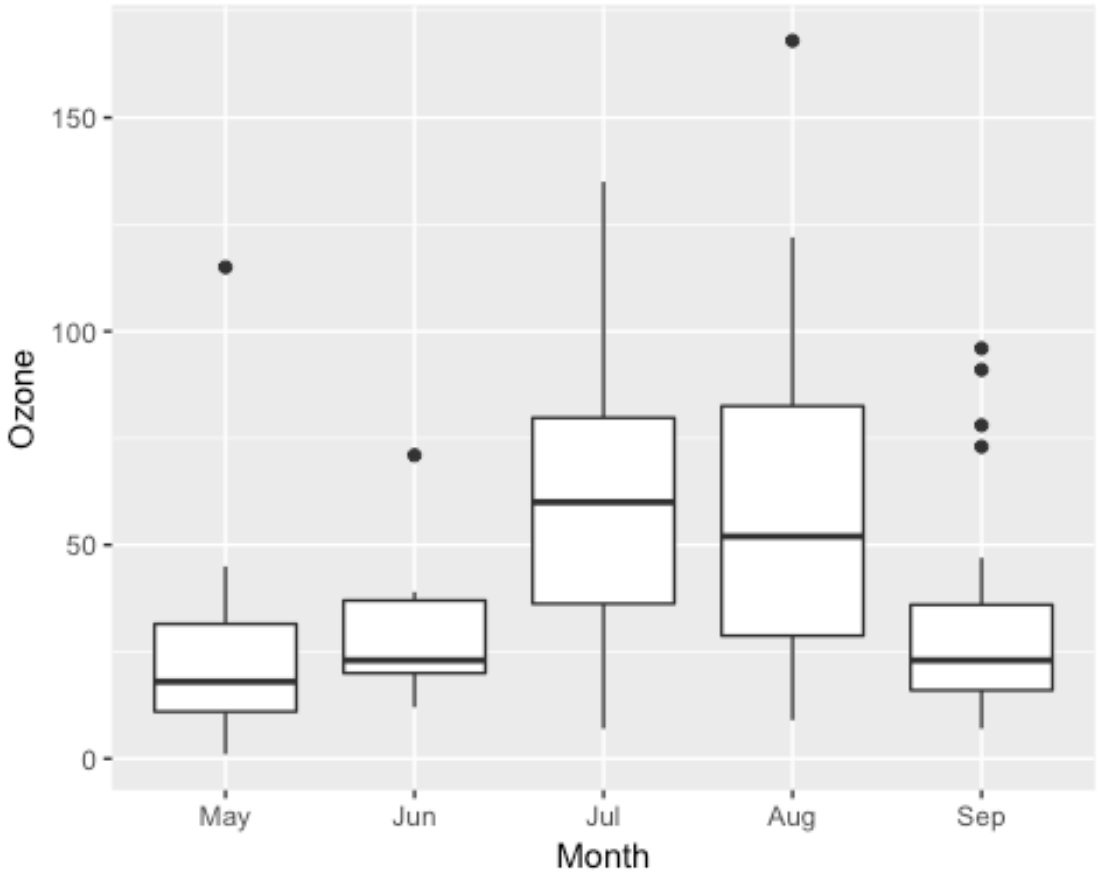
This is a brief tutorial on how to make and customize Tukey Box Plots (whiskers represent 1.5IQR and outliers are shown) with ggplot2. It is based on a logner tutorial from Jodie Burchell on her blog Standard Error. See the full tutorial here: http://t-redactyl.io/blog/2016/04/creating-plots-in-r-using-ggplot2-part-10-boxplots.html

Start by loading the needed libraries and data set. We'll use the airquality dataset from the datasets package, which contains daily air quality measures in New York from May-Sept in 1973. To see more details use ?airquality once you load the library. We'll also create a factor (categorical) variable from Month. This explicit casting to a factor makes our boxplot recognize Month as a category variable on not a simple string.

```
library(datasets)
library(ggplot2)

data(airquality)
airquality$Month <- factor(airquality$Month,
                           labels = c("May", "Jun", "Jul", "Aug", "Sep"))
```

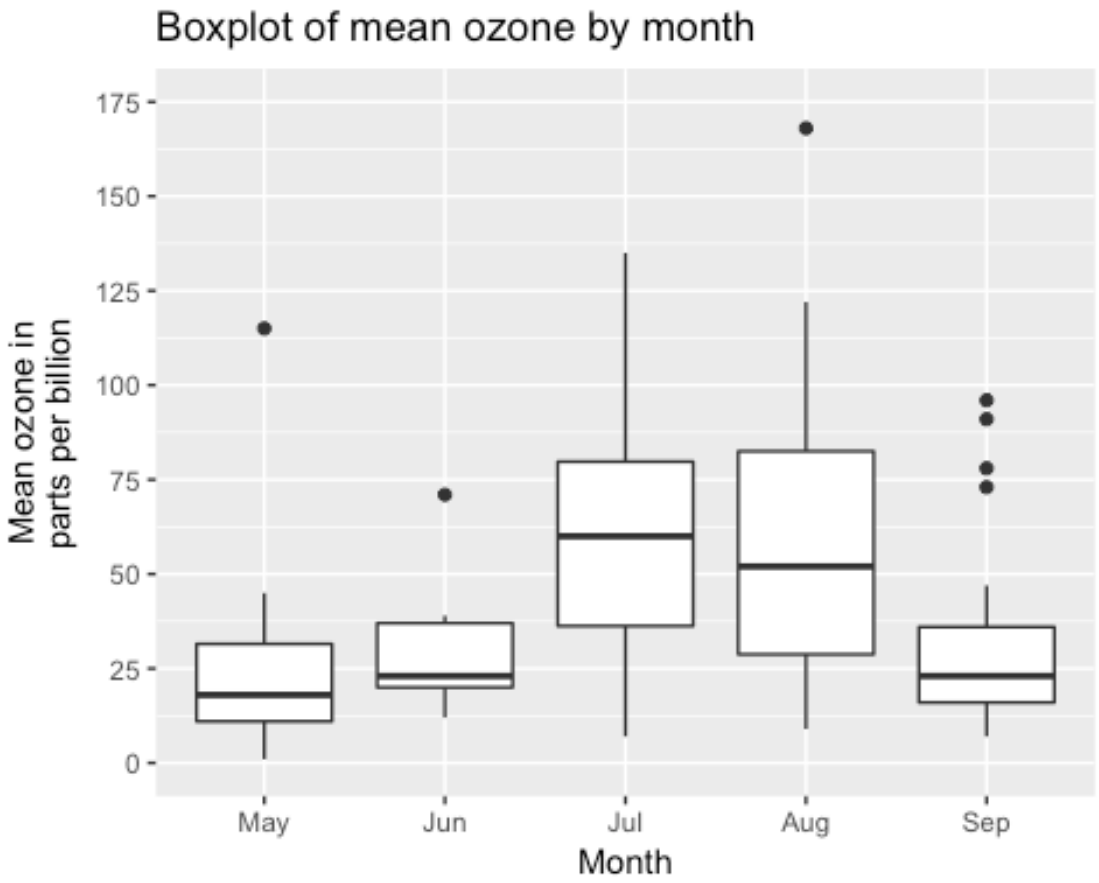
A basic boxplot is very simply to specify in ggplot2. We just need to specify the dataset, the x and y mappings (in this case we want to show the distribution of Ozone levels by month) and then call geom_boxplot().

```
p1 <- ggplot(airquality, aes(x = Month, y = Ozone)) +
        geom_boxplot()
p1

## Warning: Removed 37 rows containing non-finite values (stat_boxplot).
```



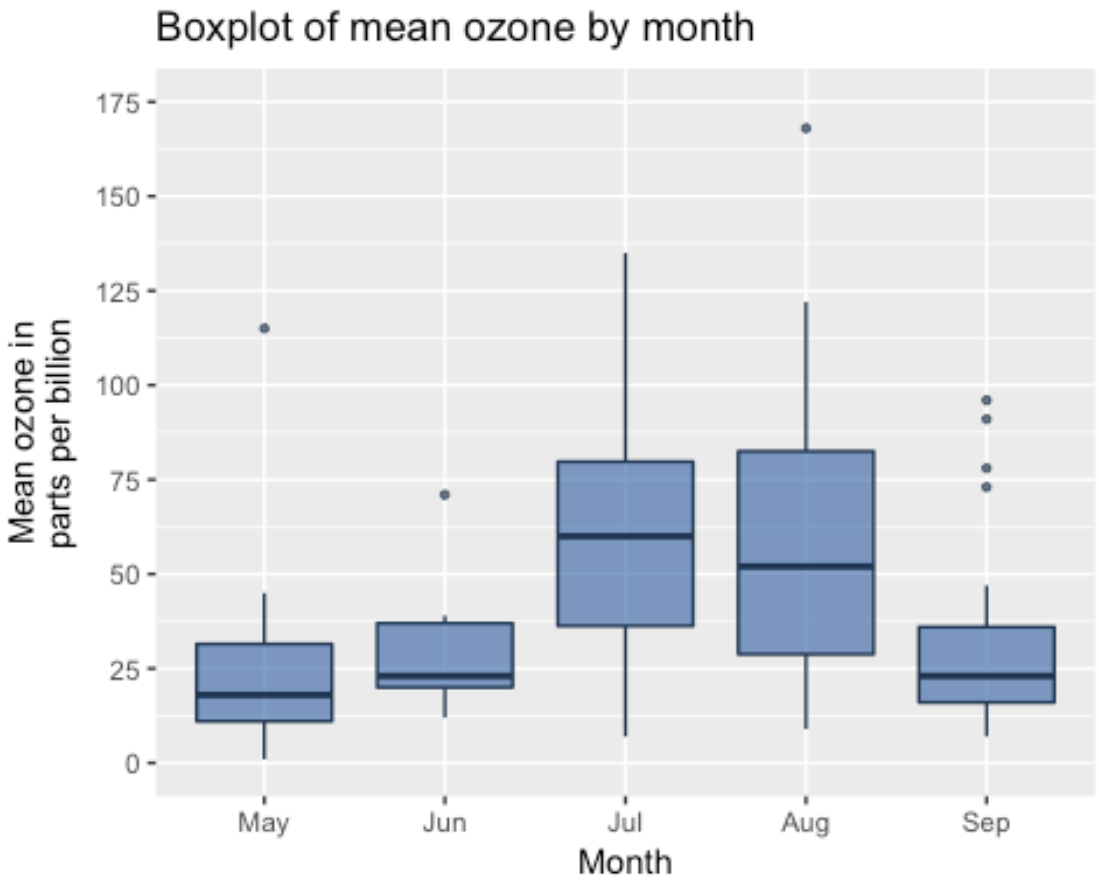
We can change axis tick marks and titles using scale_ options. And we can add a title to the chart with ggtitle(). Note, putting () around a line prints the result to the screen.

```
(p1 <- p1 + scale_y_continuous(name = "Mean ozone in\nparts per billion",
                               breaks = seq(0, 175, 25),
                               limits=c(0, 175)) +
        ggtitle("Boxplot of mean ozone by month"))

## Warning: Removed 37 rows containing non-finite values (stat_boxplot).
```



We can also adjust the color and formatting of the box elements themselves. The fill, colour (or color), and alpha values set the colors for the box. Outlier. options set formatting for outlier points. To see all of the boxplot options use ?geom_boxplot() once ggplot2 library is loaded. Note, we can define variables and use those in our plot attributes.

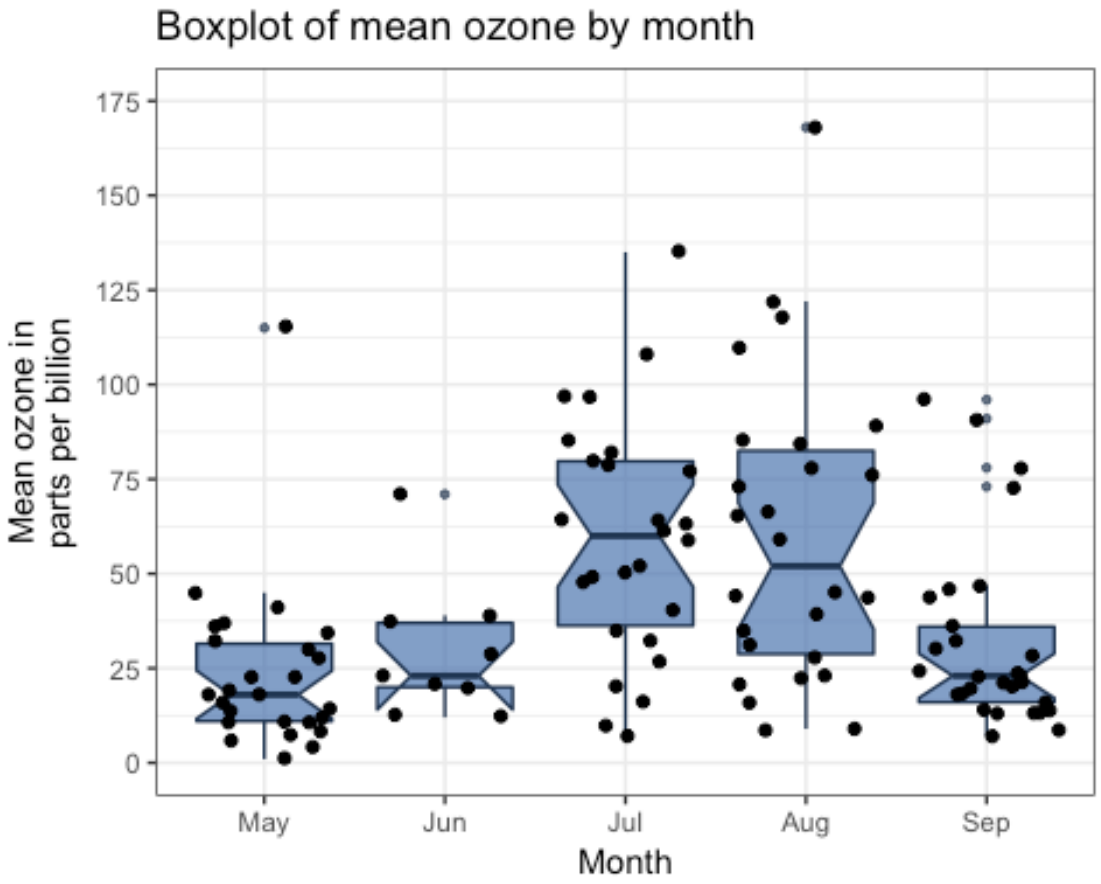
```
fill <- "#4271AE"  # Hex color code
line <- "#1F3552"  # Hex color code
# see Hex Color Picker:https://www.w3schools.com/colors/colors_picker.asp

(p1 <- ggplot(airquality, aes(x = Month, y = Ozone)) +
        geom_boxplot(fill = fill, colour = line, alpha = 0.7,
                     outlier.colour = "#1F3552", outlier.shape = 20) +
        scale_y_continuous(name = "Mean ozone in\nparts per billion",
                           breaks = seq(0, 175, 25),
                           limits=c(0, 175)) +
        scale_x_discrete(name = "Month") +
        ggtitle("Boxplot of mean ozone by month"))

## Warning: Removed 37 rows containing non-finite values (stat_boxplot).
```



We can also add a notch to the boxplot to emphasize the median, add the actual data points to the plot and make sure they don't overlap too much using geom_jitter() (see the help for more on this geom), and we can add themes to the plot as with any ggplot2 plot. You can see on our graph that the box for June looks a bit weird due to the very small gap between the 25th percentile and the median. Sometimes boxplots look better without a notch, depending on the underlying data distribution.

```
p1 <- ggplot(airquality, aes(x = Month, y = Ozone)) +
        geom_boxplot(fill = fill, colour = line, notch=TRUE, alpha = 0.7,
                     outlier.colour = "#1F3552", outlier.shape = 20) +
        scale_y_continuous(name = "Mean ozone in\nparts per billion",
                           breaks = seq(0, 175, 25),
                           limits=c(0, 175)) +
        scale_x_discrete(name = "Month") +
        ggtitle("Boxplot of mean ozone by month") + geom_jitter() + theme_bw(
)

p1

## Warning: Removed 37 rows containing non-finite values (stat_boxplot).

## notch went outside hinges. Try setting notch=FALSE.
```

```
## Warning: Removed 37 rows containing missing values (geom_point).
```



Finally, let's add another variable to our plot to further group our boxplots by nominal values.

We first need to do a little data wrangling. In order to make the graphs a bit clearer, we've kept only months "July", "Aug" and "Sep" in a new dataset airquality_trimmed. We've also mean-split Temp so that this is also categorical, and made it into a new labelled factor variable called Temp.f.

```
airquality_trimmed <- airquality[which(airquality$Month == "Jul" |
                                       airquality$Month == "Aug" |
                                       airquality$Month == "Sep"), ]
airquality_trimmed$Temp.f <- factor(ifelse(airquality_trimmed$Temp > mean(air
quality_trimmed$Temp), 1, 0), labels = c("Low temp", "High temp"))
```

Now create the boxplot graph. Firstly, in the ggplot function, we add a fill = Temp.f argument to aes (Make sure you know what this is doing!). Secondly, we customize the colours of the boxes by adding the scale_fill_brewer to the plot from the RColorBrewer package. We can also change the position of the legend by adding the legend.position = "bottom" argument to the theme option, which moves the legend under the plot. Finally, we can fix the legend title by adding the labs(fill = "Temperature") option to the plot. Try removing some of these options and see how it changes the final plot.

```r
#instal RColorBrewer if needed
library(RColorBrewer)

p1 <- ggplot(airquality_trimmed, aes(x = Month, y = Ozone, fill = Temp.f)) +
        geom_boxplot(alpha=0.7) +
        scale_y_continuous(name = "Mean ozone in\nparts per billion",
                           breaks = seq(0, 175, 25),
                           limits=c(0, 175)) +
        scale_x_discrete(name = "Month") +
        ggtitle("Boxplot of mean ozone by month") +
        theme_bw() +
        theme(plot.title = element_text(size = 14, family = "Tahoma", face =
"bold"),
              text = element_text(size = 12, family = "Tahoma"),
              axis.title = element_text(face="bold"),
              axis.text.x=element_text(size = 11),
              legend.position = "bottom") +
        scale_fill_brewer(palette = "Accent") +
        labs(fill = "Temperature")
p1
```

```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```