Post01 Sara Nam: Data visualization with ggplot2: Area Graphs, Density Plots, Heat Maps

Introduction

Invented by Hadley Wickham and Winston Chang, ggplot2 is one of the most downloaded packages in R. Not only is it a clean, efficient way of graphing, but it provides countless of options that expands the limits of graphing to produce graphs that are not only informative but aesthetically pleasing to the eye.

While common uses of ggplot2 include scatterplots, box plots, bar graphs, etc., ggplot2 can also produce aesthic looking area graphs, density graphs, and even heatmaps. We will explore these three types of graphs as well as the various functions to help visualize aspects of these graphs.

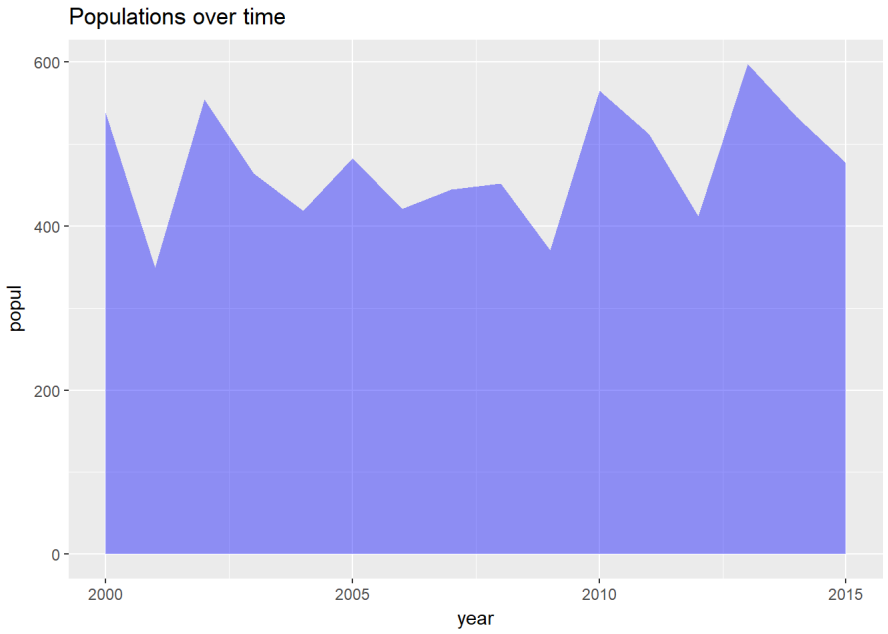
Preparing to graph

```
#Don't forget to load ggplot2!
library(ggplot2)
data("diamonds") #for later

#Creating a data frame with population data from five areas from the year 2000 to 2015.
set.seed(123)

df <- data.frame(area = rep(c("zone1", "zone2", "zone3", "zone4", "zone5"), each=16),
    year = (2000:2015),
    popul = round(c(rnorm(16, 2000, sd=500),
                rnorm(16,mean= 500, sd=77),
                rnorm(16, mean= 3000, sd=600),
                rnorm(16, mean = 6000, sd=900),
                rnorm(16, mean = 1000, sd=300)
                    )))
```

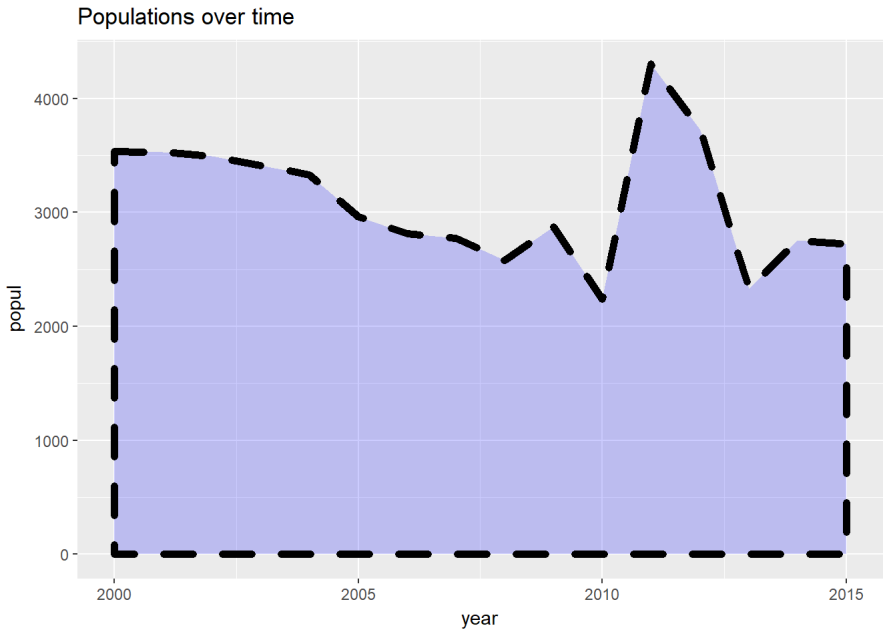
Area graphs

Visually, simple area graphs are basically line graphs with the area shaded underneath. Using ggplot, we can produce an area graph by using just one group of data from the data set and “geom_area”. Area graphs are commonly used to show trends rather than the specific values.

```
ggplot(df[df$area=="zone2", ], aes(year, popul)) +
  geom_area(fill="blue", alpha=0.4)  +
  ggtitle("Populations over time")
```




```
#alpha controls the transparency

ggplot(df[df$area=="zone3", ], aes(year, popul)) +
  geom_area(fill="blue", alpha=0.2, color="black", linetype="dashed", size=2)+
  ggtitle("Populations over time")
```

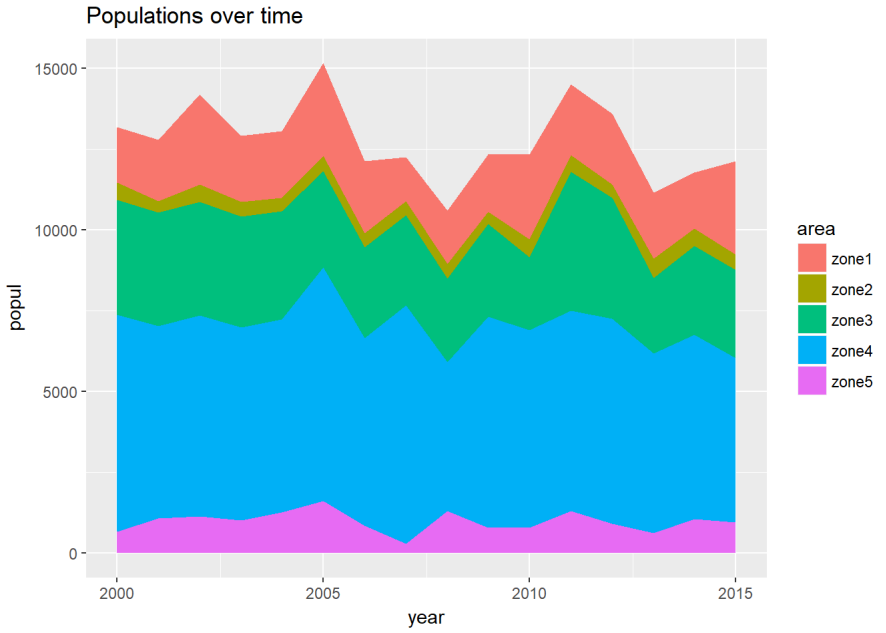


As shown above, we can specify some of the aestheics of the graph to make it look more visually appealing. For example, 'alpha' can control for transparency, and 'fill' can control the color of the area. We can also use 'linetype,' 'size,' and 'color,' to alter how the line representing the data is portrayed.

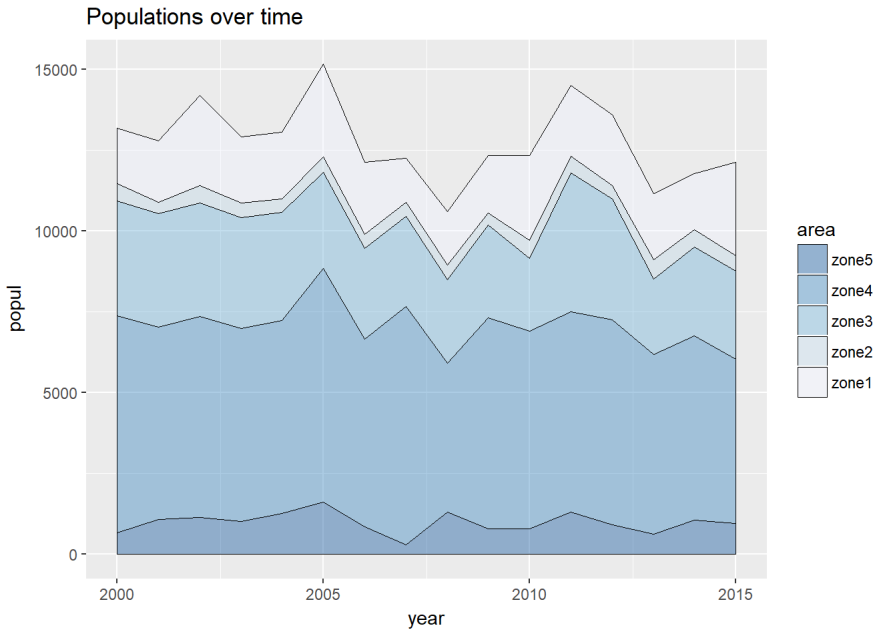
Stacked Area Graphs

Stacked area graphs are area graphs with multiple groups of data represented together, with each data set starting off from where the previous data set was plotted. These graphs can be useful when comparing data sets over a period of interval.

```
ggplot(df, aes(x= year, y = popul, fill=area)) +
  geom_area()+
  ggtitle("Populations over time")
```




```
#Playing with colors
ggplot(df, aes(year, popul, fill=area)) +
  geom_area(color="black", size=0.2, alpha=0.4) +
  scale_fill_brewer(palette="Blues",
                    breaks=rev(levels(df$area)))+
  ggtitle("Populations over time")
```

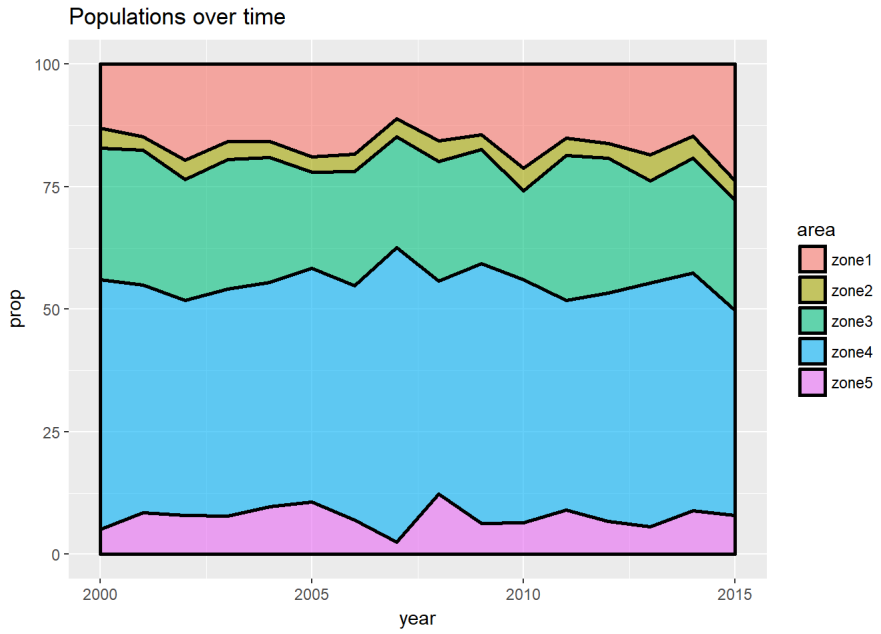



```
#Changing the order of the data fram will change stacking order
```

Proportional Stacked Area Graph

Stacked area graphs can also be proportional, as in the example below, where the population is represented in percentage of the total population over the years for each area.

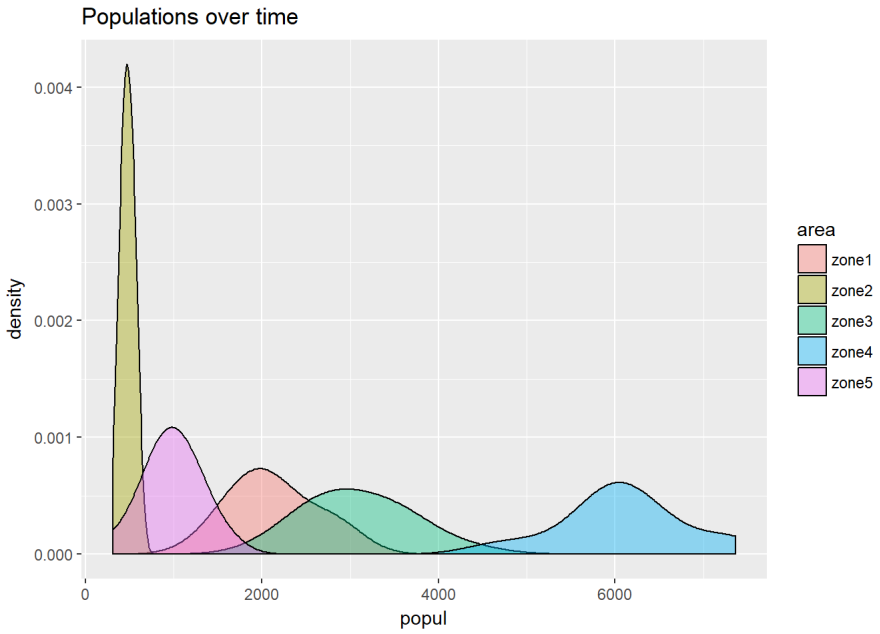
```
func = function(vec) {
  as.numeric(vec[3])/sum(df$popul[df$year ==vec[2]]) * 100
}
df$prop = apply(df, 1, func)

ggplot(df, aes(year, prop, fill=area)) +
  geom_area(alpha=0.6, size=1, color="black")+
  ggtitle("Populations over time")
```



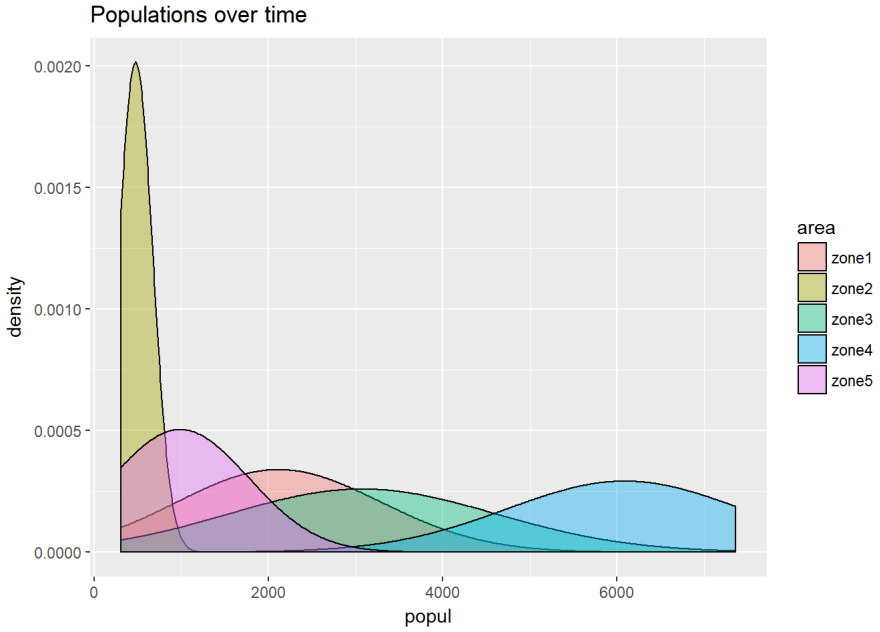
Density Plot

Density plots are also another great way of representing the distribution of data over a continuous interval. However, notice the difference in y-axis of the density plot from the area graphs. Density plots can help visualize where values are concentrated as well. It can be helpful to think of density plots as smoothed hsitograms. The bandwidth, used with 'adust,' can control how smooth the data looks.

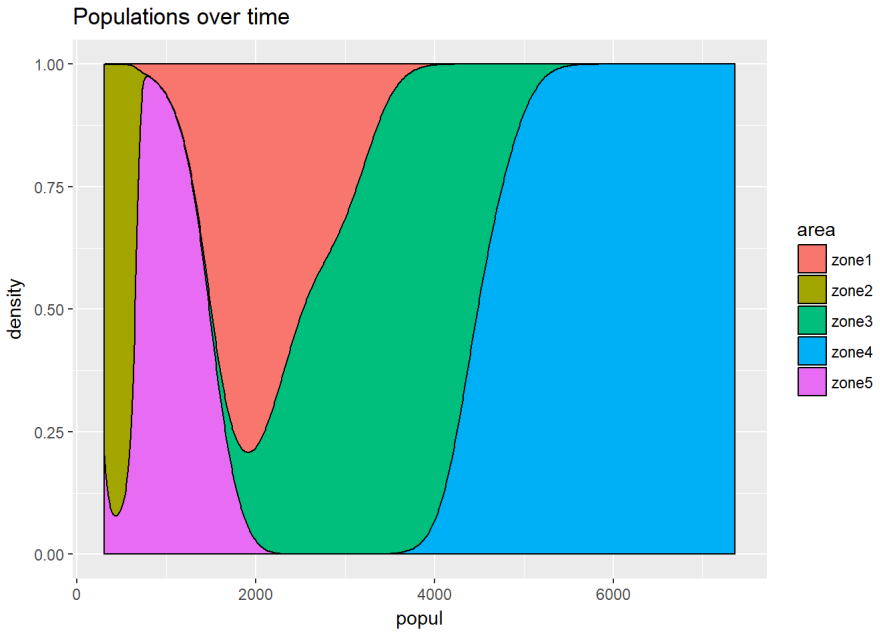
```
#Density plots with different adjust values
ggplot(df, aes(popul, group=area, fill=area))+
  geom_density(adjust=1.5, alpha=0.4)+
  ggtitle("Populations over time")
```




```
ggplot(df, aes(popul, group=area, fill=area))+
  geom_density(adjust=5, alpha=0.4)+
  ggtitle("Populations over time")
```

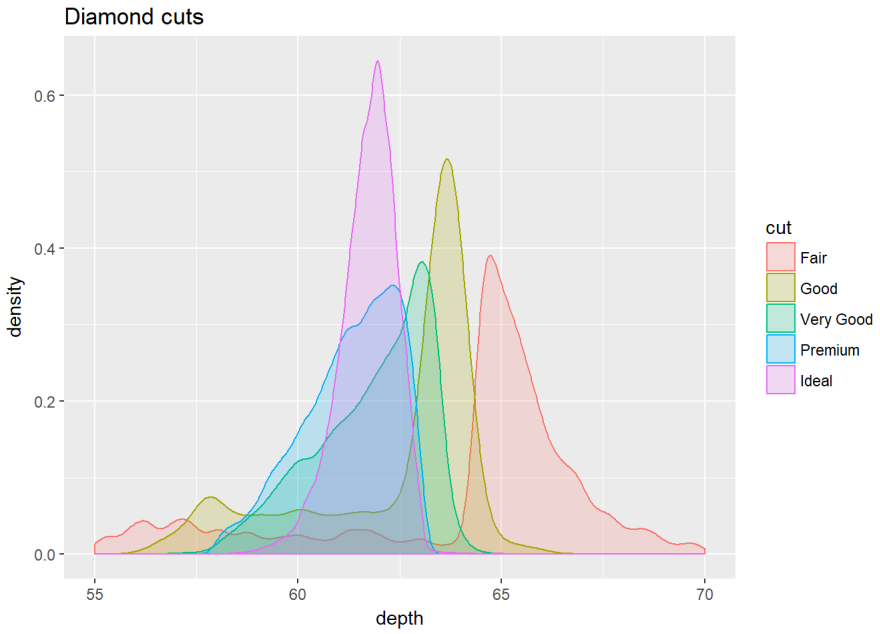



```
#Stacked Density plot
ggplot(df, aes(popul, group=area, fill=area))+
  geom_density(adjust=1.5, position="fill")+
  ggtitle("Populations over time")
```




```
#stacked density plots using R's dataset 'diamonds'
ggplot(diamonds, aes(depth, fill=cut, color=cut)) +
  geom_density(alpha=0.2) +
  xlim(55, 70)+
  ggtitle("Diamond cuts")
```

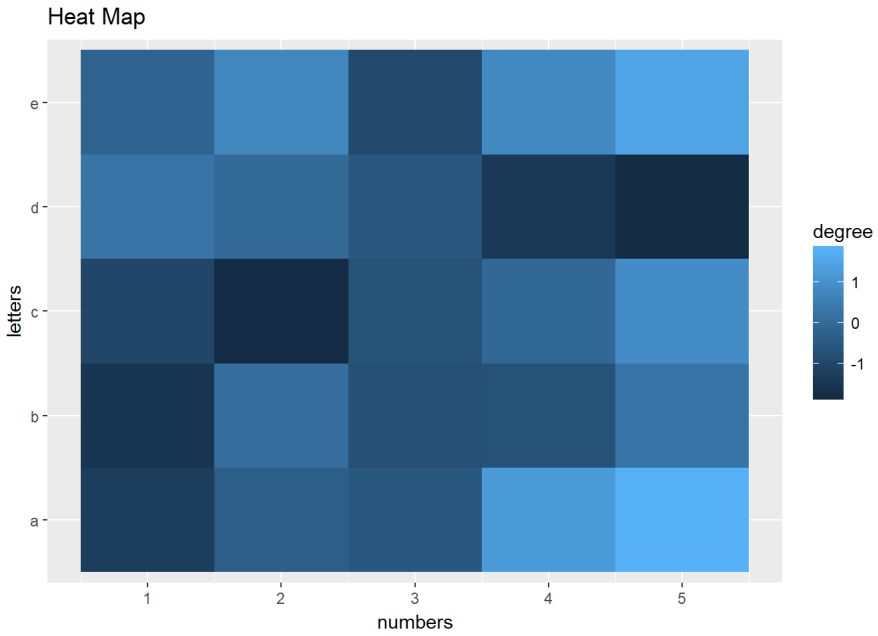
```
## Warning: Removed 45 rows containing non-finite values (stat_density).
```



Heat Maps

Finally, we'll take a look at heat maps, which uses geom_tile(). The example below shows a very simplified heat map. While you can't tell the actual data, heat maps can be great for visualizing data for comparison, and it can also be useful in observing patterns and variance.

```
heat <- expand.grid(
  numbers = 1:5,
  letters = c("a", "b", "c", "d", "e")
)

set.seed(30)
heat$degree <- rnorm(nrow(heat))

ggplot(heat, aes(x=numbers, y=letters)) +
  geom_tile(aes(fill=degree)) + ggtitle("Heat Map")
```



Conclusion/Take Away!

With the numerous graphing options provided in ggplot2, it is easy to get overwhelmed and stick to the basic plotting (such as bar graphs, histograms, line graphs). While these basic plots are also great for showcasing the data, there are more options out there that can also be great at representing the data in an easy to view and efficient manner. Some of these options include area graphs, density plots, and heat maps! Not only are they visually appealing, but they are also great for visualizing the data. Area graphs and density plots are especially useful in representing data over intervals, such as time, and heat maps are a great way to observe patterns, correlations, and variances in the data. Ggplot2 is an amazing resource for aesthetic visual representation that can help us process our data in an efficient manner.

References

1. r4stats
2. rgraph gallery
3. sthda
4. sharpsightlabs
5. ggplot2
6. datavixcatalogue
7. rstudio
8. Histograms and Density Plots