**15.3.6**

## Create Heatmap Plots

**Jeremy** had been looking at this fuel efficiency data for a bit when suddenly, inspiration struck: He could use a heatmap to visualize the average highway fuel efficiency across the type of vehicle class from 1999 to 2008. Heatmaps are a useful way to see intensity across time—and this type of chart could go over really well in a high-level presentation, as it's a good way to see a lot of information at once. Time to get coding!

Heatmap plots help visualize the relationship between one continuous numerical variable and two other variables (categorical or numerical). Heatmaps display numerical values as colors on a two-dimensional grid so that value clusters and trends are readily identifiable. For example, if we want to visualize the average highway fuel efficiency across the type of vehicle class from 1999 to 2008, our R code would look as follows:

```
> mpg_summary <- mpg %>% group_by(class,year) %>% summarize(Mean_Hwy=mean(hwy), .groups = 'keep') #create sum

> plt <- ggplot(mpg_summary, aes(x=class,y=factor(year),fill=Mean_Hwy))

> plt + geom_tile() + labs(x="Vehicle Class",y="Vehicle Year",fill="Mean Highway (MPG)") #create heatmap with
```
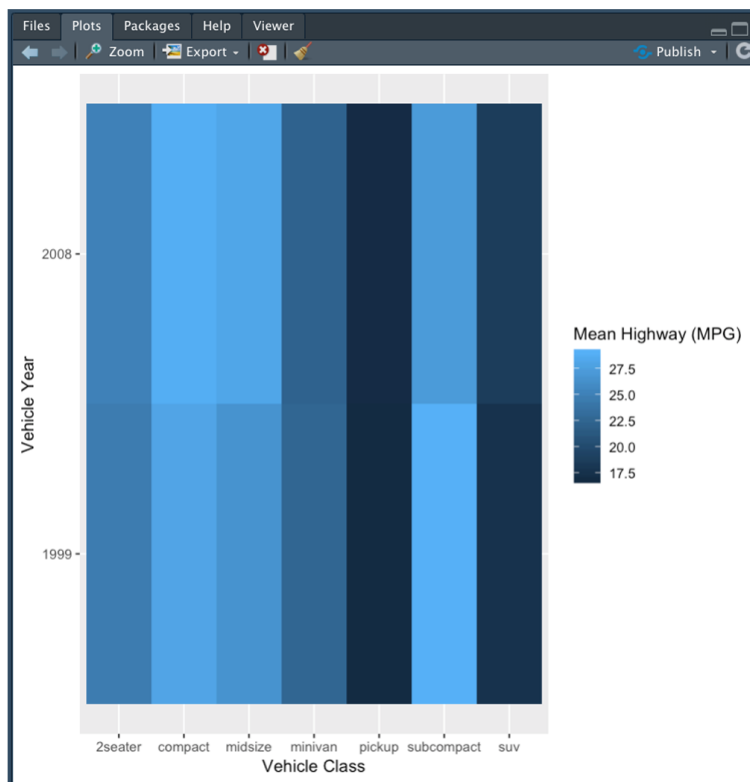
Our heatmap shows that the majority of the vehicle classes experienced an average improvement in highway fuel efficiency from 1999 to 2008. Unlike our previous ggplot visualizations, heatmaps are used to look at large trends in a dataset. Therefore, we can use heatmaps to visualize variables with a large number of values/categories. For example, if we want to look at the difference in average highway fuel efficiency across each vehicle model from 1999 to 2008, our R code would look as follows:

```
> mpg_summary <- mpg %>% group_by(model,year) %>% summarize(Mean_Hwy=mean(hwy), .groups = 'keep') #create sum
> plt <- ggplot(mpg_summary, aes(x=model,y=factor(year),fill=Mean_Hwy)) #import dataset into ggplot2
> plt + geom_tile() + labs(x="Model",y="Vehicle Year",fill="Mean Highway (MPG)") + #add heatmap with labels >
```
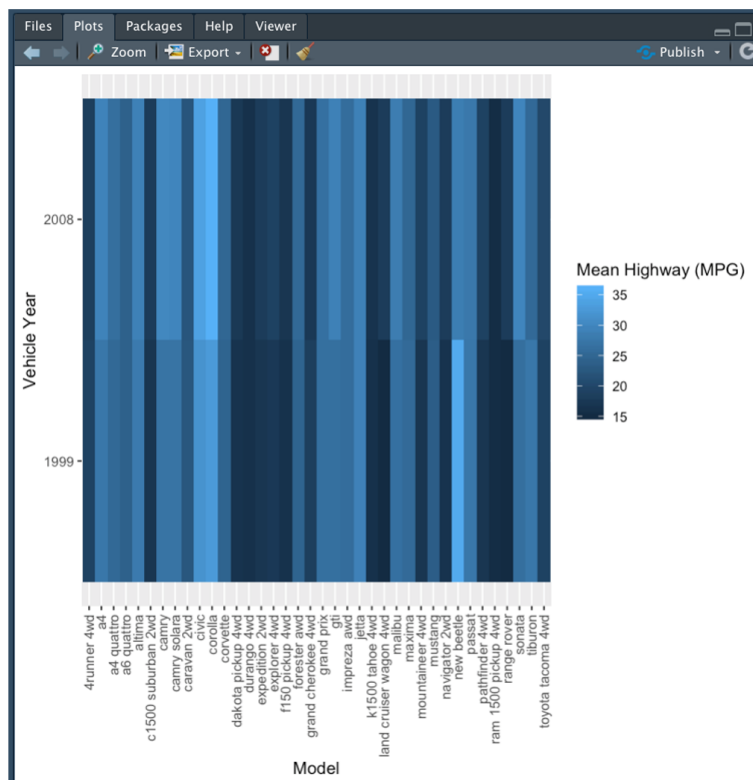
**NOTE**

When using variables with a number of different values (categories, levels, etc.), you may want to adjust the angle of your text from 45 to 90 degrees. If you angle your text, you may need to also adjust your `hjust` and `vjust` arguments to ensure your labels line up with your tick marks.

Although boxplots and heatmaps are two of the more common advanced visualizations used in data science, there are a number of more specific ggplot2 visualization functions that can be used on an individual basis.

The ggplot2 documentation sufficiently describes how to implement each function, but does not provide logic or guidance on how to select a visualization for your data. Thankfully, there are many cheat sheets available that provide guidance on what visualizations to use given the dimensions and data types you wish to use. One of the most popular is the **RStudio Data Visualization ggplot2 Cheat Sheet** **(https://github.com/rstudio/cheatsheets/blob/master/data-visualization.pdf)** , which is used to help data scientists determine what functions to use to generate an appropriate visualization for their analysis.