Week 3: Data Visualization

MTH 365: Introduction to Data Science

2024-09-03

Recommended Reading:

- *Modern Data Science with R* Ch. 2: Data Visualization
- *Modern Data Science with R* Ch. 3: A Grammar for Graphics

`ggplot2`

In the lecture, we showed that statistics alone may lead to a misunderstanding of the data. Therefore, when working with new data, we should always make some visualizations to help us understand the data. A common way for plotting in R today is through `ggplot2`.

`ggplot2` is an R package (located in `tidyverse`) for "decoratively creating graphics"

- https://ggplot2.tidyverse.org/reference/

```r
library(tidyverse)
```

Example: Hate crimes and income inequality

A FiveThirtyEight article published in 2017 claimed that higher rates of hate crimes were tied to greater income inequality.

https://fivethirtyeight.com/features/higher-rates-of-hate-crimes-are-tied-to-income-inequality/

- FiveThirtyEight publishes their data sets - let's investigate for ourselves.
- Data set is posted in BlueLine. Download this data set, and save it to your computer.

Follow these steps to read the data into RStudio:

1. In the Environment tab, click "Import Dataset". Since this is a CSV document, you want to import a text file.

2. Navigate to your CSV data set. Make sure that the first row contains column names.
3. Import the data.

Another way to do this:

1. Put the data file next to the Rmd file.
2. In the console, print`read.csv(hate.crimes.csv)` See ?read.csv for more information about this function

```
hate_crimes <- read.csv("./data/hate_crimes.csv")
glimpse(hate_crimes)
head(hate_crimes)
```

A simple plot?

Basic Format of a Plot:

`data and aesthetics + plot type + options`

The data, aesthetics, and plot type are necessary to create a plot. For example, below is what happens when we just specify the data and aesthetics.

```
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi))
```

Variable type

After specifying the data and aesthetics, we need to decide the plot type. In order to do that, we need to know the variable type(s). There are two different ways to distinguish the variables.

By function:

1. response variable
2. explanatory variable

By value type:

1. continuous variable
2. categorical variable

Type of plots

1. Visualize one continous variable.

Usually for the response variable using histograms and density plots

(a) Histograms

```
#Histogram (default bin number)
ggplot(hate_crimes, aes(x=median_income)) +
  geom_histogram()

#Histogram (change number of bins)
ggplot(hate_crimes, aes(x=median_income)) +
  geom_histogram(bins = 10)
```

(b) Density Plots

```
## Density Plot
ggplot(hate_crimes, aes(x=median_income)) +
  geom_density()

## Density Plot (add a fill color and transparency with alpha)
ggplot(hate_crimes, aes(x=median_income)) +
  geom_density(fill = "blue", alpha = 0.5)
```

(c) Box Plots

What is the difference between box-plot and histogram/density plot? Box-plot shows the median but not the distribution.

```
#Box plot horizontally
ggplot(hate_crimes, aes(x=median_income)) +
  geom_boxplot()

#Box plot vertically
ggplot(hate_crimes, aes(y=median_income)) +
  geom_boxplot()
```

2. Visualize one categorical variable, usually for the response variable using a bar-plot

```
# Bar Plot with bars groups on x-axis
ggplot(hate_crimes, aes(x=region)) +
  geom_bar()

# Bar Plot with bars groups on y-axis
# changing x=region to y=region is equivalent
ggplot(hate_crimes, aes(x=region)) +
  geom_bar() +
  coord_flip()
```

3. Visualize two continous variables.

Focus on showing the relation between them. Can be response variable + explanatroy variable. Can also be explanatory variable + explanatory variable.

(a) Scatterplots

```
#Scatterplot with points
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_point()

#Scatterplot with state labels (text) instead of points
#Be careful with clutter
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_text(aes(label=state))
```

(b) Line plots and Smooth Line Plots to connect the points in the scatterplot.

The Smooth Line Plots help show the trend due to smoothness

```
#Line plot - connects all points
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_line()

#Smooth Plot show general trend (GAM Model)
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_smooth()
```

```r
#Smooth Plot showing linear trend (linear regression)
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_smooth(method = "lm")

#Can combine to show points and smooth lines in same graph
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi)) +
  geom_point +
  geom_smooth(method = "lm")
```

4. Visualize one continuous variable and one categorical variable (Multiple groups)

Sometimes we want to compare the variable(s) across multiple groups. eg: compare median income across different region. Which plots can compare multiple group?

Theses are called side-by-side plots.

```r
#Density Plot for each region
ggplot(hate_crimes, aes(x=median_income, group = region)) +
  geom_density()

#Boxplot for each region
ggplot(hate_crimes, aes(y=median_income, group = region)) +
  geom_boxplot()
```

Wait, how can I know which group is which group?

Include options like color and size

```r
#Same boxplot as above, colored by region
ggplot(hate_crimes, aes(y=median_income, group = region)) +
  geom_boxplot(aes(color = region))

#Same density plot as above, colored by region
ggplot(hate_crimes, aes(x=median_income, group = region)) +
  geom_density(aes(color = region))

#Scatterplot from before, colored by region
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi, group = region)) +
  geom_point(aes(color = region))
```

```
#Scatterplot from before, colored by region, dot size by unemployment
ggplot(hate_crimes, aes(x=median_income, y=hate_crimes_fbi, group = region)) +
  geom_point(aes(color = region, size = unemployment))
```

Adding plot title and changing axis titles

Add x-axis, y-axis labels and title

```
#Same plot as previous with better labels/title
ggplot(hate_crimes, aes(x=gini_index, y=hate_crimes_fbi)) +
  geom_point(aes(color=region, size=unemployment)) +
  xlab('Gini Index') +
  ylab('Hate Crimes (FBI, Sept. 2016)') +
  ggtitle('The relation between Gini Index and Hate Crime rate in 2016')
```

Faceting by groups

Instead of putting all groups information into one page, you can do by each panel.

```
#Facet by region into one row
#nrow = 2 gives a 2x2 layout of plots

ggplot(hate_crimes, aes(x=gini_index, y=hate_crimes_fbi)) +
  geom_point(aes(color=region, size=unemployment)) +
  xlab('Gini Index') +
  ylab('Hate Crimes (FBI, Sept. 2016)') +
  ggtitle('The relation between Gini Index and Hate Crime rate in 2016')+
  facet_wrap(~region, nrow=1)
```

Try it for yourself

1. Suppose we are interested in the unemployment rate and want to see its distribution.

```
ggplot(hate_crimes, aes(x=unemployment)) +
  geom_histogram()
```

2. Suppose we want to show the relation between unemployment rate and median income.

```
ggplot(hate_crimes, aes(x=unemployment, y = median_income)) +
  geom_point()
```

3. Report an approximate median for the unemployment rate.

```
ggplot(hate_crimes, aes(y=unemployment)) +
  geom_boxplot()
```

4. Show the unemployment rate across different region. Use color to indicate different regions.

```
ggplot(hate_crimes, aes(y = unemployment, x = region)) +
  geom_boxplot(aes(color = region))
```

5. Show the relation between unemployment rate and FBI hate crime rate. Use size to indicate gini index. Make sure to include axis labels and title.

```
ggplot(hate_crimes, aes(x=unemployment, y=hate_crimes_fbi)) +
  geom_point(aes(color = region, size=gini_index)) +
  xlab('Unemployment rate') +
  ylab('Hate Crimes (FBI, Sept. 2016)') +
  ggtitle('The relation between unemplyment rate and Hate Crime rate in 2016')
```

6. Plot the distribution of gini index and put differnet region on different panel.

```
ggplot(hate_crimes, aes(x = gini_index)) +
  geom_histogram() +
  facet_wrap(~region)
```

Weather patterns

The data set `Weather` contains data on weather-related variables for several world cities.

```
#install.packages(mosaicData)
library(mosaicData)
data(Weather)
glimpse(Weather)
```

(a). We want to study the average temperature. First, we want to see its distribution. How would we do this?

```r
# Histogram
ggplot(Weather, aes(x=avg_temp)) +
  geom_histogram()
#Density Plot
ggplot(Weather, aes(x=avg_temp)) +
  geom_density()
#Box Plot
ggplot(Weather, aes(y=avg_temp)) +
  geom_boxplot()
```

(b). Next, we want to study the distribution of average temperature across different cities. How would we do this?

```r
#Same as above (boxplot) but color/group by city
#coloud also use density plot
ggplot(Weather, aes(y=avg_temp, group=city)) +
  geom_boxplot(aes(color=city))
#City labels now also found on the x axis
ggplot(Weather, aes(y=avg_temp, x = city, group=city)) +
  geom_boxplot(aes(color=city))
#Removes legend from above cause no longer necessary
ggplot(Weather, aes(y=avg_temp, x = city, group=city)) +
  geom_boxplot(aes(color=city)) + theme(legend.position = "none")
```

(c). The average temperature may be related to date. How to show the relation between temperature and date?

```r
#Scatterplot of data vs avg_temp (can also use smooth plot)
#But smooth plot misses out on some interesting patterns
ggplot(Weather, aes(x=date, y=avg_temp)) +
  geom_point()

#add point and smooth plot
ggplot(Weather, aes(x=date, y=avg_temp)) +
  geom_point() +
  geom_smooth()
```

(d). Maybe different city has totally different trend on average temperature (ie. relationship between temperature and date grouped by city).

```r
#color point by city
#see some obvious trends
ggplot(Weather, aes(x=date, y=avg_temp)) +
  geom_point(aes(color=city))
```

(e). What if we only care about one city? Show the relationship between temperature and date for Beijing only.

```r
#Filter so only beijing. Can do this for different cities
Beijing <- Weather %>% filter(city=='Beijing')

#scatterplot for just beijing
ggplot(Beijing, aes(x=date, y=avg_temp)) +
  geom_point()

#smooth plot for just Beijing
ggplot(Beijing, aes(x=date, y=avg_temp)) +
  geom_smooth()

#combining both plots into one
ggplot(Beijing, aes(x=date, y=avg_temp)) +
  geom_smooth() +
  geom_point()
```

(f). Instead of the date, we can also use the month. Show the relationship between month and temperature (for Beijing).

```r
#scatterplot of month vs temperature
ggplot(Beijing, aes(y=avg_temp, x = month)) +
  geom_point()

#use as.factor if want every month to be labeled on plot
ggplot(Beijing, aes(y=avg_temp, x = as.factor(month))) +
  geom_point()
```

(g). Show the distribution of average temperature for the Beijing data set by month.

```r
#density plot for average temp by month
#month as factor so we have different colors
#use alpha otherwise density plots overlap
ggplot(Beijing, aes(x=avg_temp, group=month)) +
  geom_density(aes(color=as.factor(month), fill=as.factor(month)), alpha=0.5)
```

```r
#faceting by month since lots of overlapping plots
ggplot(Beijing, aes(x=avg_temp, group=month)) +
  geom_density(aes(color=as.factor(month), fill=as.factor(month)), alpha=0.5) +
  facet_wrap(~month, nrow=3)
```

(h). Show the relationship between the low temperature and the high temperature colored by month

```r
#scatterplot where points are colored by month
#could also facet
ggplot(Beijing, aes(x=low_temp, y=high_temp)) +
  geom_point(aes(color=as.factor(month)))
```