|  |
| --- |
| Here’s my belated Statistics Sunday post, using a cool technique I just learned about: gghighlight. This R package works with ggplot2 to highlight a subset of data. To demonstrate, I’ll use a dataset I analyzed  Gghighlight package  HIGHLIGHTING YOUR DATA IN R: THE OLD SCHOOL WAY  To implement this idea, we don't need any fancy packages other than ggplot2.  The steps are simple:   1. Using ggplot2, create a plot with your full data set in grey. 2. Create a new data frame that has been subset to only include the data which you would like to highlight. 3. Add the highlighted data on to your plot created in step 1.  Set the color to something other than grey. 4. Celebrate!   **Example**  For our example, we are going to examine the crime incident dataset from Seattle 911 Calls on data.gov.  Note that I have covered this data set through multiple blog posts already such as map plots in R and time based heat maps.  **Set Up R**  In terms of setting up the R working environment, we have a couple of options open to us.  We can use something like R Studio for a local analytics on our personal computer.  Or we can use a free, hosted, multi-language collaboration environment like Watson Studio.  If you'd like to get started with R in IBM Watson Studio, please have a look at the tutorial I wrote.  **Install and Load Libraries**  install.packages("lubridate")  install.packages("ggplot2")  install.packages("ggmap")  install.packages("data.table")  install.packages("ggrepel")  install.packages("dplyr")  install.packages("magrittr")  library(lubridate)  library(ggplot2)  library(ggmap)  library(dplyr)  library(data.table)  library(ggrepel)  library(magrittr)  **Download the Data**  incidents= fread('https://raw.githubusercontent.com/lgellis/MiscTutorial/master/ggmap/i2Sample.csv', stringsAsFactors = FALSE)  str(incidents) attach(incidents)  # Create some color variables for graphing later  custGrey = "#A9A9A9"  #add year to the incidents data frame  incidents$ymd <-mdy\_hms(Event.Clearance.Date)  incidents$month <- lubridate::month(incidents$ymd)  incidents$year <- year(incidents$ymd)  incidents$wday <- lubridate::wday(incidents$ymd, label = TRUE)  incidents$hour <- hour(incidents$ymd)  #Create a more manageable data frame with only 2017 data  i2 <- incidents[year>=2017, ]  #Only include complete cases  i2[complete.cases(i2), ]  attach(i2)  head(i2)  **Create a basic time series plot showing the count of 911 event types by month.**  #Group the data into a new data frame which has the count of events per month by subgroup  groupSummaries <- i2 %>%    group\_by(month, Event.Clearance.SubGroup) %>%    summarize(N = length(Event.Clearance.SubGroup))  #View the new data set  head(groupSummaries, n=100)  attach(groupSummaries)  #Graph the data set through ggplot 2  ggplot(groupSummaries, aes(x=month, y=N, color=Event.Clearance.SubGroup) )+    geom\_line() +    theme(legend.position="bottom",legend.text=element\_text(size=7),          legend.title = element\_blank()) +    scale\_x\_discrete(name ="Month",                   limits=c(3,6,9,12))  Highlight your Data in R with ggplot    **Create a Graph Highlighting Data with a Max Month Count of 95 or Greater**  # Create a data frame with only events types that have had a peak of 95 calls in a month or more  groupSummariesF <- groupSummaries %>%    group\_by(Event.Clearance.SubGroup) %>%    filter(max(N) > 95) %>%    ungroup()  head(groupSummariesF)  # Create a layered plot with one layer of grey data for the full data set and one layer of color data for the subset data set  ggplot() +    geom\_line(aes(month, N, group = Event.Clearance.SubGroup),              data = groupSummaries, colour = alpha("grey", 0.7)) +    geom\_line(aes(month, N, group = Event.Clearance.SubGroup, colour = Event.Clearance.SubGroup),              data = groupSummariesF) +    scale\_x\_discrete(name ="Month",                     limits=c(3,6,9,12)) +    theme(legend.position="bottom",legend.text=element\_text(size=7),          legend.title = element\_blank())  Highlight your Data in R with ggplot    One of the great things about the "old school way" of doing this type of highlighting is that it can be done with presumably every extension to the ggplot2 package.  For example, you can use this same technique to highlight with the ggmap package.   The code for these graphs is incredibly simple and has been included in my github repo.  Highlight your Data in R with ggplot and ggmap  HIGHLIGHTING YOUR DATA IN R: THE NEW SCHOOL WAY  While the above methodology is quite easy, it can be a bit of a pain at times to create and add the new data frame.  Further, you have to tinker more with the labelling to really call out the highlighted data points.  Thanks to Hiroaki Yutani, we now have the gghighlight package which does most of the work for us with a small function call!!   Please note that a lot of this code was created by looking at examples on her introduction document.  The new school way is even simplier:   1. Using ggplot2, create a plot with your full data set. 2. Add the gghighlight() function to your plot with the conditions set to identify your subset. 3. Celebrate! This was one less step AND we got labels!   **Example**  For our first example, we are going to create the same time series graph from above.  However, we are going to perform the highlighting with gghighlight vs manual layering.  # Install the gghighlight package  install.packages("gghighlight")  library(gghighlight)  # Create the highlighted graph  ggplot(groupSummaries, aes(month, N, colour = Event.Clearance.SubGroup)) +    geom\_line() +    gghighlight(max(N) > 95,  label\_key = Event.Clearance.SubGroup) +    scale\_x\_discrete(name ="Month",                     limits=c(3,6,9,12))  Highlight your Data in R with gghighlight  **More Examples**  Well that was so easy, we are going to try a few more ggmap plot types to see how we fare.  Below show both a scatterplot and histogram chart.  # Try a scatterplot chart  ggplot(groupSummaries, aes(month, N, colour = Event.Clearance.SubGroup, use\_group\_by=FALSE)) +    geom\_point() +    gghighlight(N > 200,  label\_key = Event.Clearance.SubGroup) +    scale\_x\_discrete(name ="Month",                     limits=c(3,6,9,12))  # Try a histogram chart  ggplot(groupSummaries, aes(N, fill = Event.Clearance.SubGroup)) +    geom\_histogram() +    theme(legend.position="bottom",legend.text=element\_text(size=7),          legend.title = element\_blank()) +    gghighlight(N > 100,  label\_key = Event.Clearance.SubGroup, use\_group\_by = FALSE) +    facet\_wrap(~ Event.Clearance.SubGroup)  Highlight your Data in R with gghighlight  Highlight your Data in R with gghighlight  setwd("~/R") library(tidyverse)  books<-read\_csv("2017\_books.csv", col\_names = TRUE)  ## Warning: Duplicated column names deduplicated: 'Author' => 'Author\_1' [13]  ## Parsed with column specification: ## cols( ## .default = col\_integer(), ## Title = col\_character(), ## Author = col\_character(), ## G\_Rating = col\_double(), ## Started = col\_character(), ## Finished = col\_character() ## )  ## See spec(...) for full column specifications.  One analysis I conducted with this dataset was to look at the correlation between book length (number of pages) and read time (number of days it took to read the book). We can also generate a scatterplot to visualize this relationship.  cor.test(books$Pages, books$Read\_Time)  ##  ## Pearson's product-moment correlation ##  ## data: books$Pages and books$Read\_Time ## t = 3.1396, df = 51, p-value = 0.002812 ## alternative hypothesis: true correlation is not equal to 0 ## 95 percent confidence interval: ## 0.1482981 0.6067498 ## sample estimates: ## cor  ## 0.4024597  scatter <- ggplot(books, aes(Pages, Read\_Time)) +  geom\_point(size = 3) +  theme\_classic() +  labs(title = "Relationship Between Reading Time and Page Length") +  ylab("Read Time (in days)") +  xlab("Number of Pages") +  theme(legend.position="none",plot.title=element\_text(hjust=0.5))    There’s a significant positive correlation here, meaning the longer books take more days to read. It’s a moderate correlation, and there are certainly other variables that may explain why a book took longer to read. For instance, nonfiction books may take longer. Books read in October or November (while I was gearing up for and participating in NaNoWriMo, respectively) may also take longer, since I had less spare time to read. I can conduct regressions and other analyses to examine which variables impact read time, but one of the most important parts of sharing results is creating good data visualizations. How can I show the impact these other variables have on read time in an understandable and visually appealing way?  gghighlight will let me draw attention to different parts of the plot. For example, I can ask gghighlight to draw attention to books that took longer than a certain amount of time to read, and I can even ask it to label those books.  library(gghighlight)  scatter + gghighlight(Read\_Time > 14) +  geom\_label(aes(label = Title),  hjust = 1,  vjust = 1,  fill = "blue",  color = "white",  alpha = 0.5) |

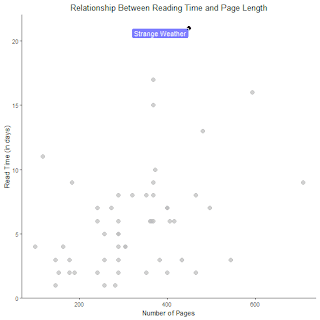
Here, the gghighlight function identifies the subset (books that took more than 2 weeks to read) and labels those books with the Title variable. Three of the four books with long read time values are non-fiction, and one was read for a course I took, so reading followed a set schedule. But the fourth is a fiction book, which took over 20 days to read. Let’s see how month impacts reading time, by highlighting books read in November. To do that, I’ll need to alter my dataset somewhat. The dataset contains a starting date and finish date, which were read in as characters. I need to convert those to dates and pull out the month variable to create my indicator.

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

books$Started <- mdy(books$Started)  
books$Start\_Month <- month(books$Started)  
books$Month <- ifelse(books$Start\_Month > 10 & books$Start\_Month < 12, books$Month <- 1,  
 books$Month <- 0)  
scatter + gghighlight(books$Month == 1) +  
 geom\_label(aes(label = Title), hjust = 1, vjust = 1, fill = "blue", color = "white", alpha = 0.5)

[](https://i0.wp.com/2.bp.blogspot.com/-CHyZ17fDkWo/W2ncu6eYNbI/AAAAAAAANYs/mPH6cmNgLu88tYOTttaK-yGNEl9xxlnOgCLcBGAs/s1600/unnamed-chunk-4-1.png?ssl=1)

The book with the longest read time was, in fact, read during November, when I was spending most of my time writing.