C7M4L2A1: You and GGPlot

Background for this activity

In this activity, you'll review a scenario, and use ggplot2 to quickly create data visualizations that allow you to explore your data and gain new insights. You will learn more about basic ggplot2 syntax and data visualization in R.

Throughout this activity, you will also have the opportunity to practice writing your own code by making changes to the code chunks yourself. If you encounter an error or get stuck, you can always check the Lesson2_GGPlot_Solutions .rmd file in the Solutions folder under Week 4 for the complete, correct code.

The Scenario

In this scenario, you are a junior data analyst working for a hotel booking company. You have cleaned and manipulated your data, and gotten some initial insights you would like to share. Now, you are going to create some simple data visualizations with the `ggplot2` package. You will use basic `ggplot2` syntax and troubleshoot some common errors you might encounter.

Step 1: Import your data

In the chunk below, you will use the `read_csv()` function to import data from a .csv in the project folder called "hotel_bookings.csv" and save it as a data frame called `hotel_bookings`:

If this line causes an error, copy in the line setwd("/cloud/project/Course 7/Week 4") before it.

```
hotel_bookings <- read.csv("hotel_bookings.csv")
```

Step 2: Look at a sample of your data

Use the `head()` function to preview your data:

```
head(hotel_bookings)
```

```
##          hotel is_canceled lead_time arrival_date_year arrival_date_month
## 1 Resort Hotel           0       342              2015               July
## 2 Resort Hotel           0       737              2015               July
## 3 Resort Hotel           0         7              2015               July
## 4 Resort Hotel           0        13              2015               July
## 5 Resort Hotel           0        14              2015               July
## 6 Resort Hotel           0        14              2015               July
##   arrival_date_week_number arrival_date_day_of_month stays_in_weekend_nights
## 1                       27                         1                       0
## 2                       27                         1                       0
## 3                       27                         1                       0
## 4                       27                         1                       0
## 5                       27                         1                       0
## 6                       27                         1                       0
##   stays_in_week_nights adults children babies meal country market_segment
## 1                    0      2        0      0   BB     PRT         Direct
## 2                    0      2        0      0   BB     PRT         Direct
```

```
## 3                   1      1        0      0   BB     GBR         Direct
## 4                   1      1        0      0   BB     GBR      Corporate
## 5                   2      2        0      0   BB     GBR      Online TA
## 6                   2      2        0      0   BB     GBR      Online TA
##   distribution_channel is_repeated_guest previous_cancellations
## 1               Direct                 0                      0
## 2               Direct                 0                      0
## 3               Direct                 0                      0
## 4            Corporate                 0                      0
## 5                TA/TO                 0                      0
## 6                TA/TO                 0                      0
##   previous_bookings_not_canceled reserved_room_type assigned_room_type
## 1                              0                 C                 C
## 2                              0                 C                 C
## 3                              0                 A                 C
## 4                              0                 A                 A
## 5                              0                 A                 A
## 6                              0                 A                 A
##   booking_changes deposit_type agent company days_in_waiting_list customer_type
## 1               3   No Deposit  NULL    NULL                    0     Transient
## 2               4   No Deposit  NULL    NULL                    0     Transient
## 3               0   No Deposit  NULL    NULL                    0     Transient
## 4               0   No Deposit   304    NULL                    0     Transient
## 5               0   No Deposit   240    NULL                    0     Transient
## 6               0   No Deposit   240    NULL                    0     Transient
##   adr required_car_parking_spaces total_of_special_requests reservation_status
## 1   0                            0                         0          Check-Out
## 2   0                            0                         0          Check-Out
## 3  75                            0                         0          Check-Out
## 4  75                            0                         0          Check-Out
## 5  98                            0                         1          Check-Out
## 6  98                            0                         1          Check-Out
##   reservation_status_date
## 1              2015-07-01
## 2              2015-07-01
## 3              2015-07-02
## 4              2015-07-02
## 5              2015-07-03
## 6              2015-07-03
```

You can also use `colnames()` to get the names of all the columns in your data set. Run the code chunk below to find out the column names in this data set:

```
colnames(hotel_bookings)
```

```
##  [1] "hotel"                          "is_canceled"
##  [3] "lead_time"                      "arrival_date_year"
##  [5] "arrival_date_month"             "arrival_date_week_number"
##  [7] "arrival_date_day_of_month"      "stays_in_weekend_nights"
##  [9] "stays_in_week_nights"           "adults"
## [11] "children"                       "babies"
## [13] "meal"                           "country"
## [15] "market_segment"                 "distribution_channel"
## [17] "is_repeated_guest"              "previous_cancellations"
## [19] "previous_bookings_not_canceled" "reserved_room_type"
```

```
## [21] "assigned_room_type"             "booking_changes"
## [23] "deposit_type"                   "agent"
## [25] "company"                        "days_in_waiting_list"
## [27] "customer_type"                  "adr"
## [29] "required_car_parking_spaces"    "total_of_special_requests"
## [31] "reservation_status"             "reservation_status_date"
```

Step 3: Install and load the 'ggplot2' package

If you haven't already installed and loaded the `ggplot2` package, you will need to do that before you can use the `ggplot()` function.

Run the code chunk below to install and load `ggplot2`. This may take a few minutes.

Step 4: Begin creating a plot

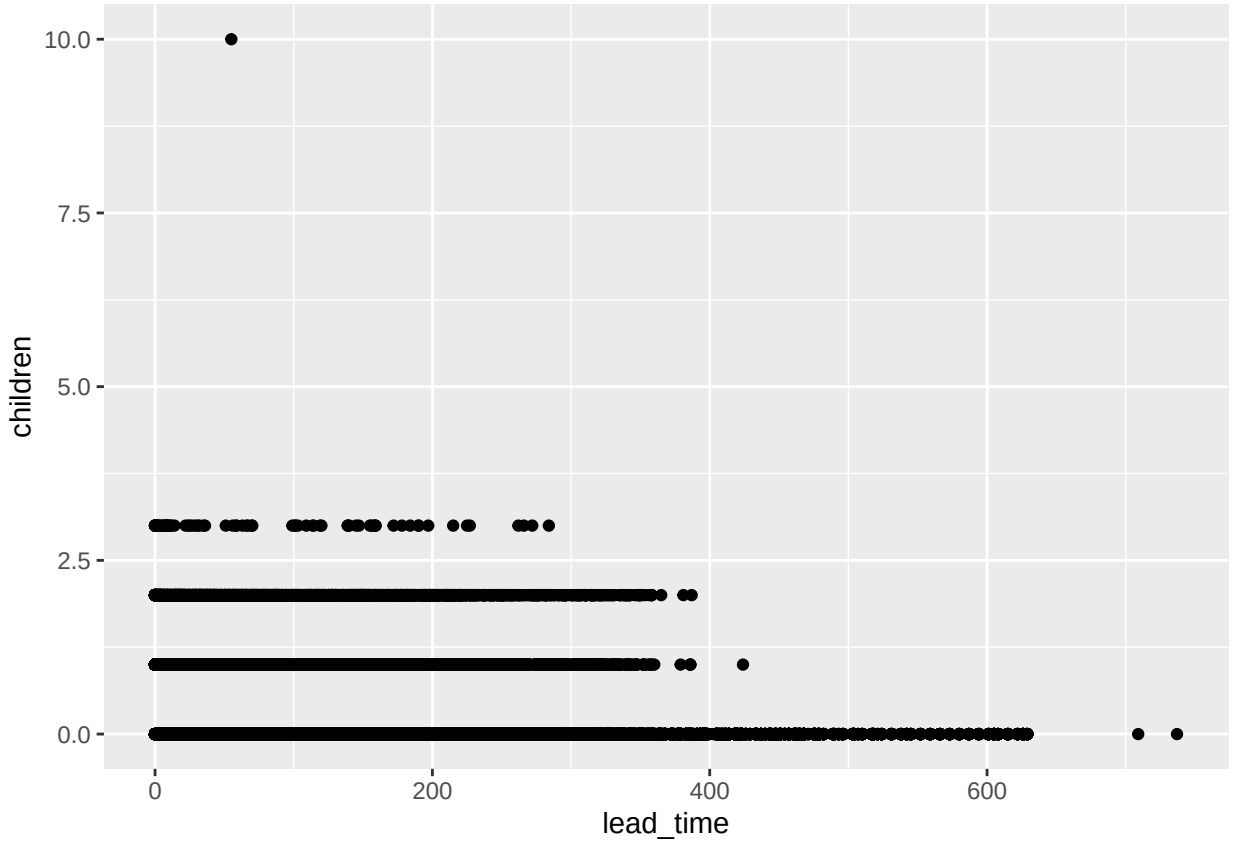
A stakeholder tells you, "I want to target people who book early, and I have a hypothesis that people with children have to book in advance."

When you start to explore the data, it doesn't show what you would expect. That is why you decide to create a visualization to see how true that statement is– or isn't.

You can use `ggplot2` to do this. Try running the code below:

```
ggplot(data = hotel_bookings) +
  geom_point(mapping = aes(x = lead_time, y = children))
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



The geom_point() function uses points to create a scatterplot. Scatterplots are useful for showing the relationship between two numeric variables. In this case, the code maps the variable 'lead_time' to the x-axis and the variable 'children' to the y-axis.

On the x-axis, the plot shows how far in advance a booking is made, with the bookings furthest to the right happening the most in advance. On the y-axis it shows how many children there are in a party.

The plot reveals that your stakeholder's hypothesis is incorrect. You report back to your stakeholder that many of the advanced bookings are being made by people with 0 children.

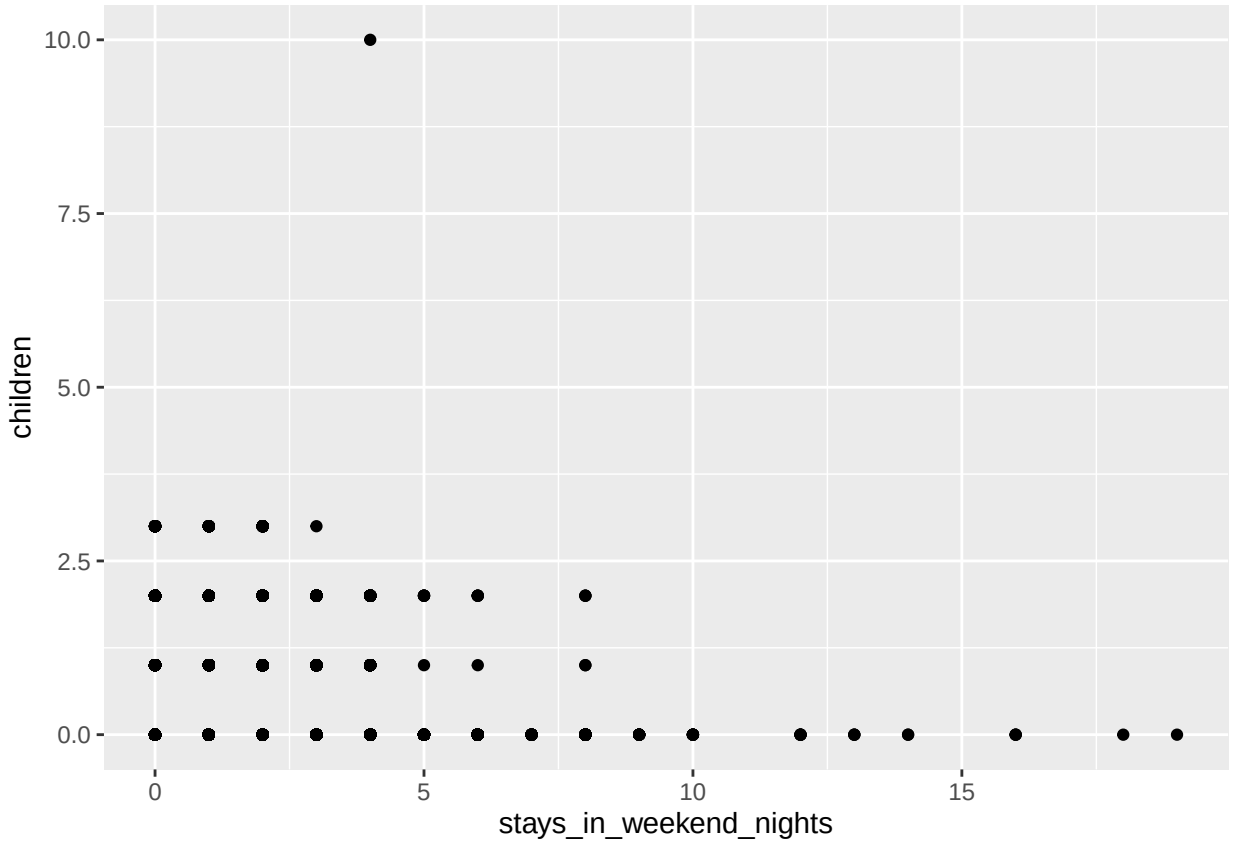
Step 5: Try it on your own

Next, your stakeholder says that she wants to increase weekend bookings, an important source of revenue for the hotel. Your stakeholder wants to know what group of guests book the most weekend nights in order to target that group in a new marketing campaign. She suggests that guests without children book the most weekend nights. Is this true?

Try mapping 'stays_in_weekend_nights' on the x-axis and 'children' on the y-axis by filling out the remainder of the code below.

```
ggplot(data = hotel_bookings) +
 geom_point(mapping = aes(x = stays_in_weekend_nights, y = children))
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



If you correctly enter this code, you should have a scatterplot with 'stays_in_weekend_nights' on the x-axis and 'children' on the y-axis.

What did you discover? Is your stakeholder correct?

What other types of plots could you use to show this relationship?

Remember, if you're having trouble filling out a code block, check the solutions document for this activity.

Activity Wrap Up

The `ggplot2` package allows you to quickly create data visualizations that can answer questions and give you insights about your data. Now that you are a little more familiar with the basic `ggplot2` syntax, you can practice these skills by modifying the code chunks in the rmd file, or use this code as a starting point in your own project console. With `ggplot2`, you will be able to create and share data visualizations without leaving your `R` console. You will learn more about `ggplot2` throughout this course and eventually create even more complex and beautiful visualizations!