Visualisation

2022-08-09

Import data

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

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
```

Install and load the 'ggplot2' package

install and load the `ggplot2` package.

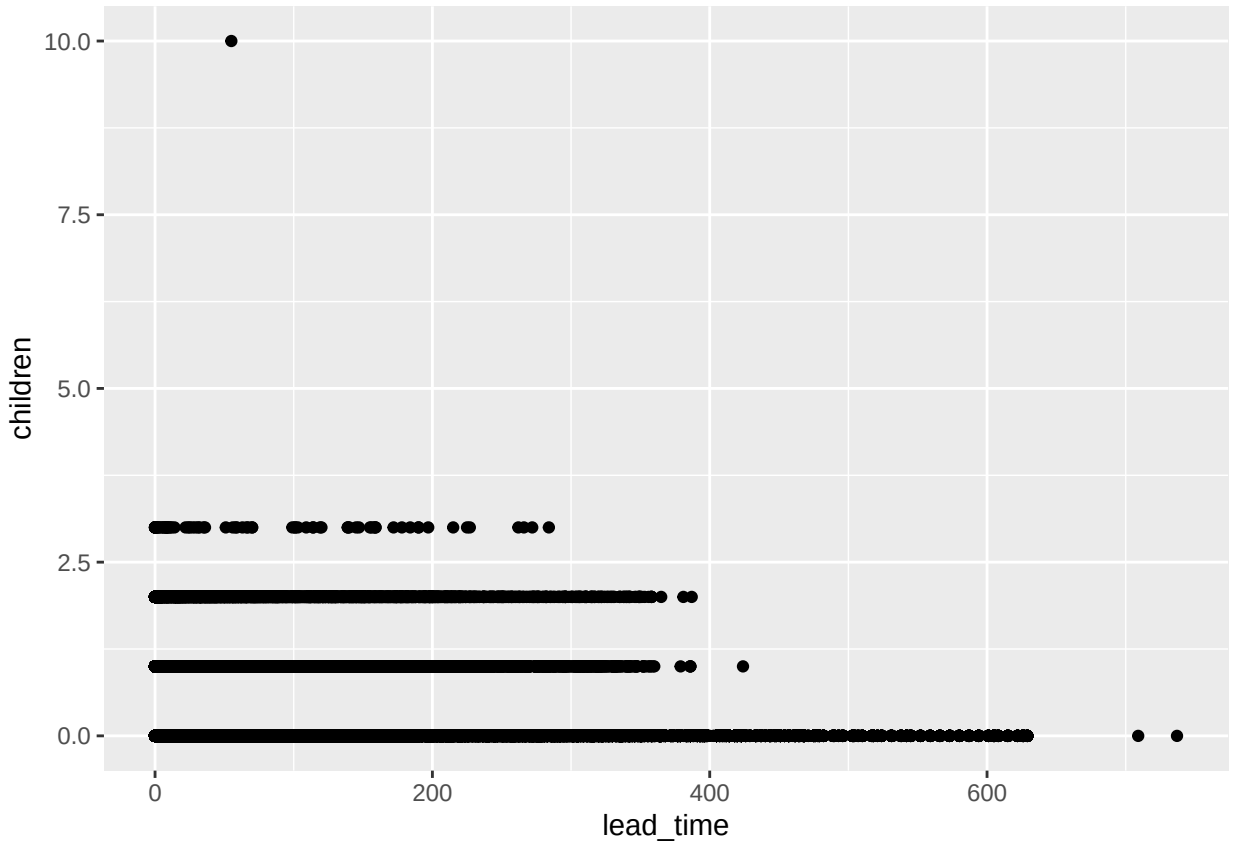
Run the code chunk below to install and load `ggplot2`. This may take a few minutes.

Creating a plot

I want to target people who book early, and I have a hypothesis that people with children have to book in advance.

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

```
## Warning: Removed 4 rows containing missing values (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.

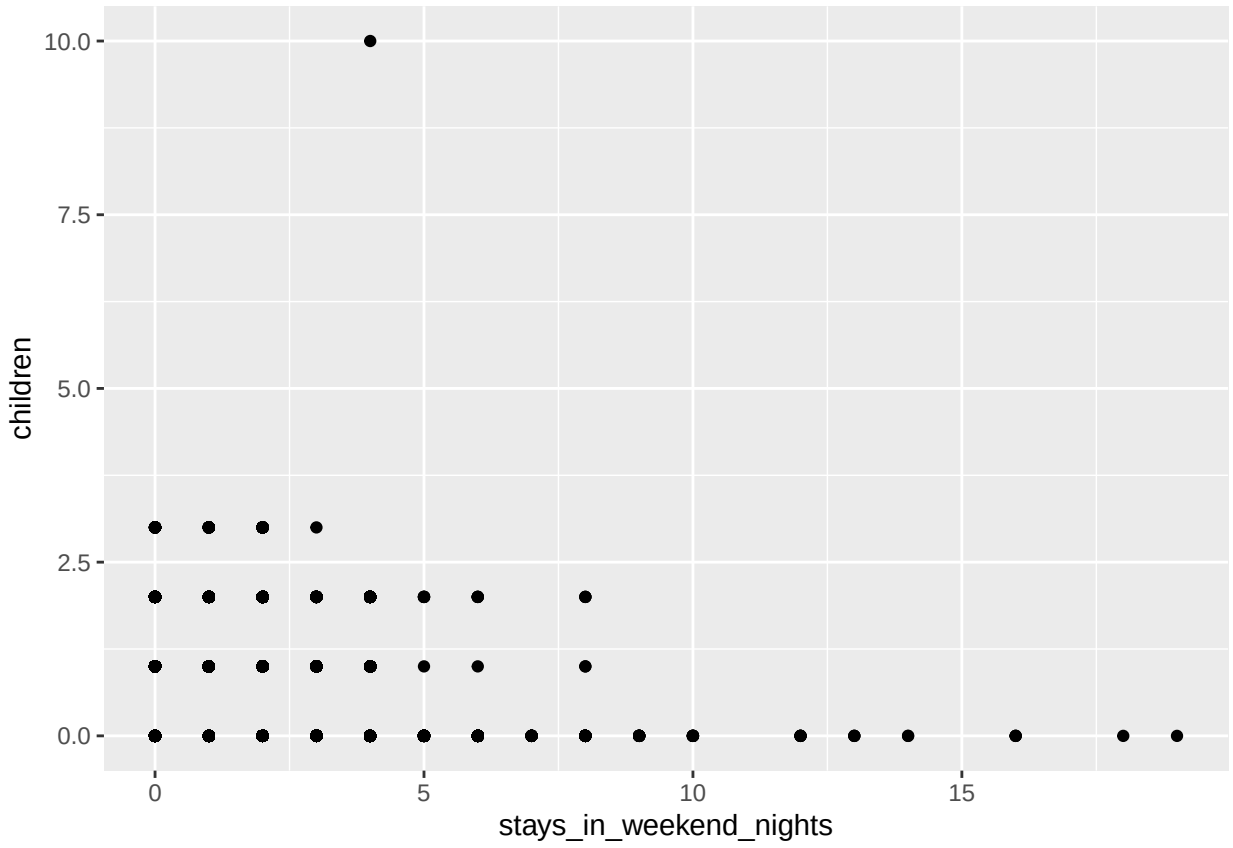
On the x-axis, the plot shows how far in advance a booking is made, with the bookings furthest to the right happening the most in advance. On the y-axis it shows how many children there are in a party.

The plot reveals that our hypothesis is incorrect. Many of the advanced bookings are being made by people with 0 children.

Next, we want to know what group of guests book the most weekend nights in order to target that group in a new marketing campaign.

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

```
## Warning: Removed 4 rows containing missing values (geom_point).
```

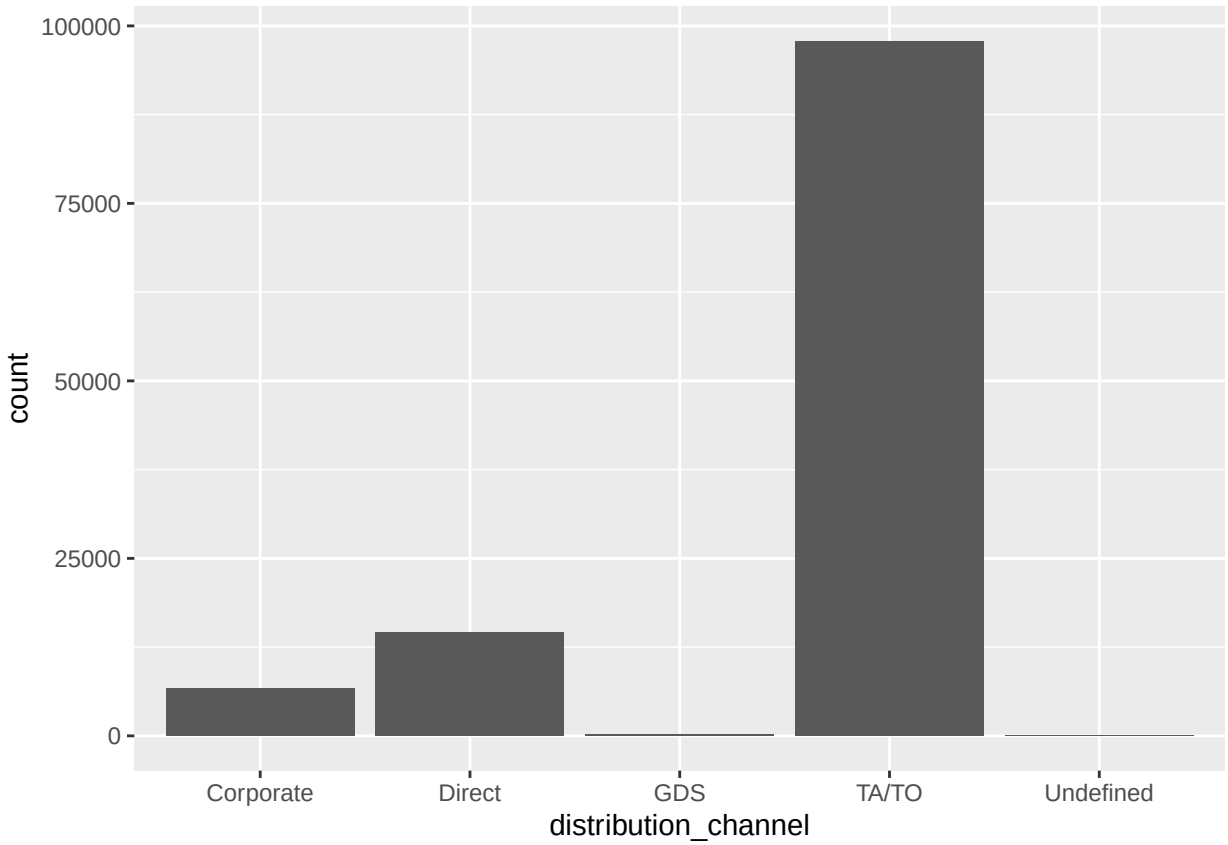


Making a Bar Chart

We need to know how many of the transactions are occurring for each different distribution type.

Previously, we used `geom_point` to make a scatter plot comparing lead time and number of children. Now, we will use `geom_bar` to make a bar chart in this code chunk:

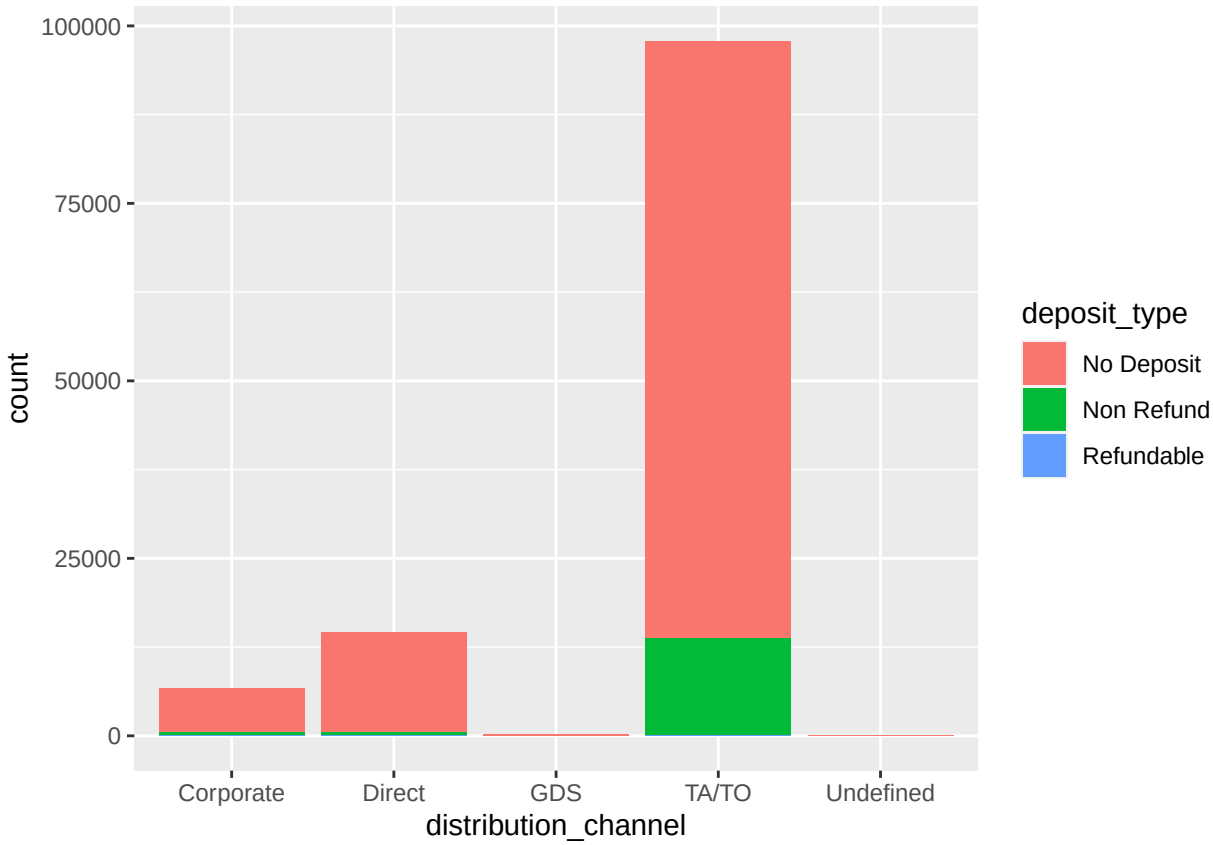
```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel))
```



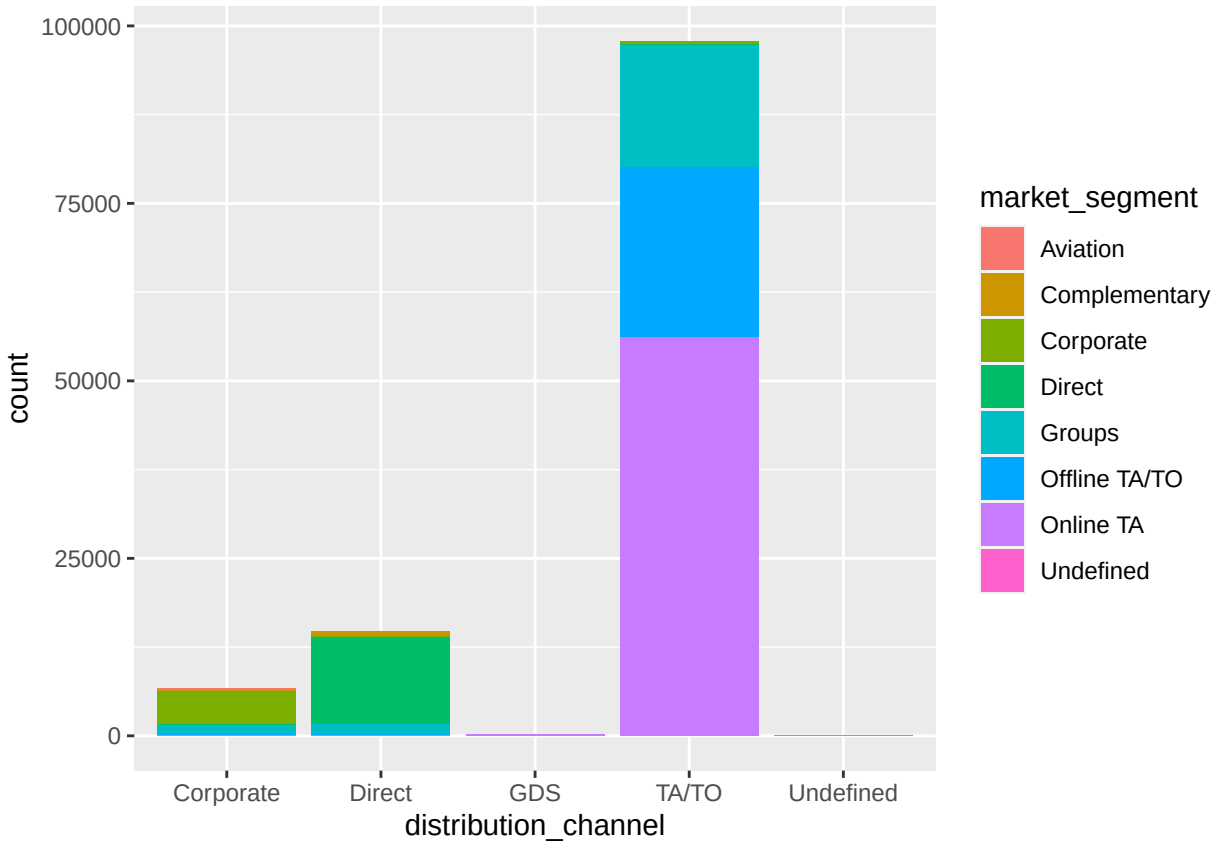
Diving deeper into bar charts

We want to know if the number of bookings for each distribution type is different depending on whether or not there was a deposit or what market segment they represent.

adding 'fill=deposit_type' after 'x = distribution_channel':



Now adding 'fill=market_segment' to this code chunk instead of 'fill=deposit_type':

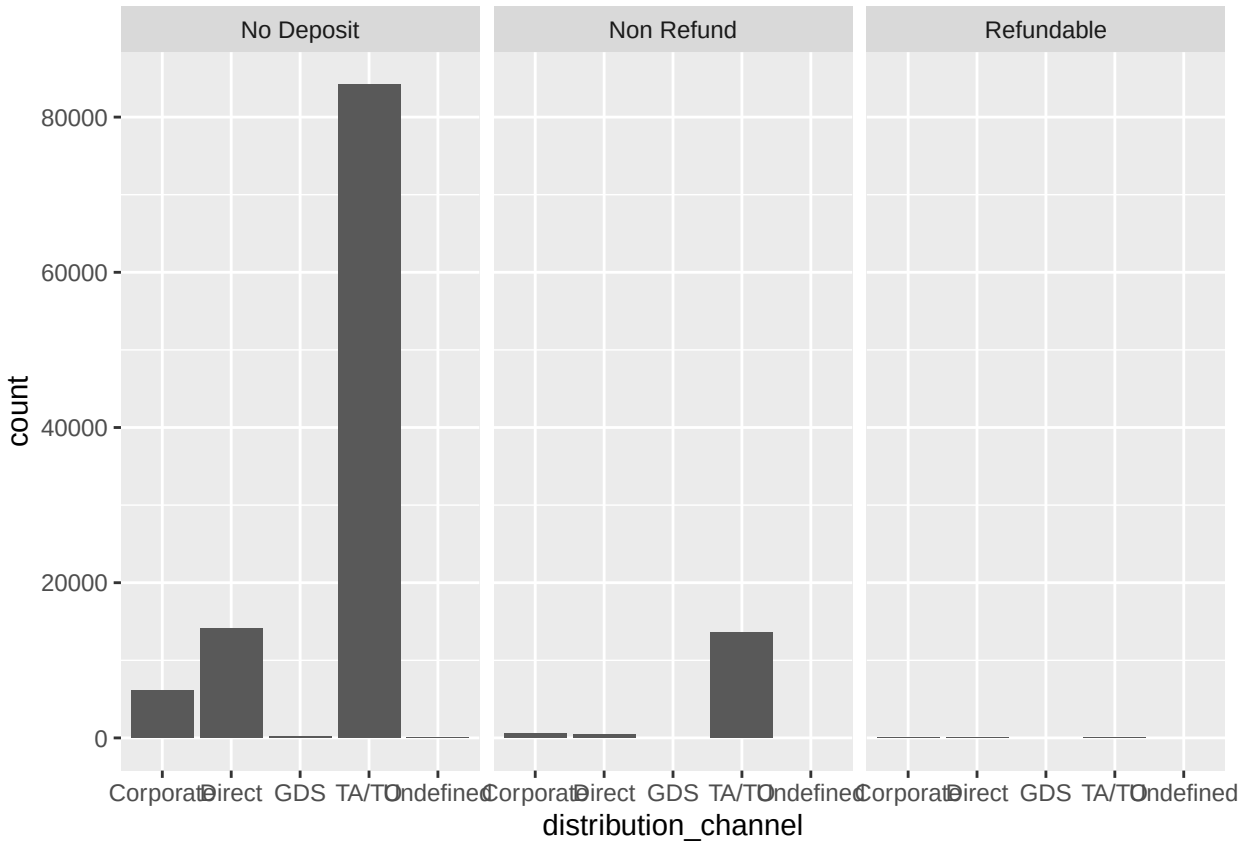


This bar chart is similar to the previous chart, except that 'market_segment' data is being recorded in the color-coded sections of each bar.

Facets galore

Create separate charts for each deposit type and market segment.

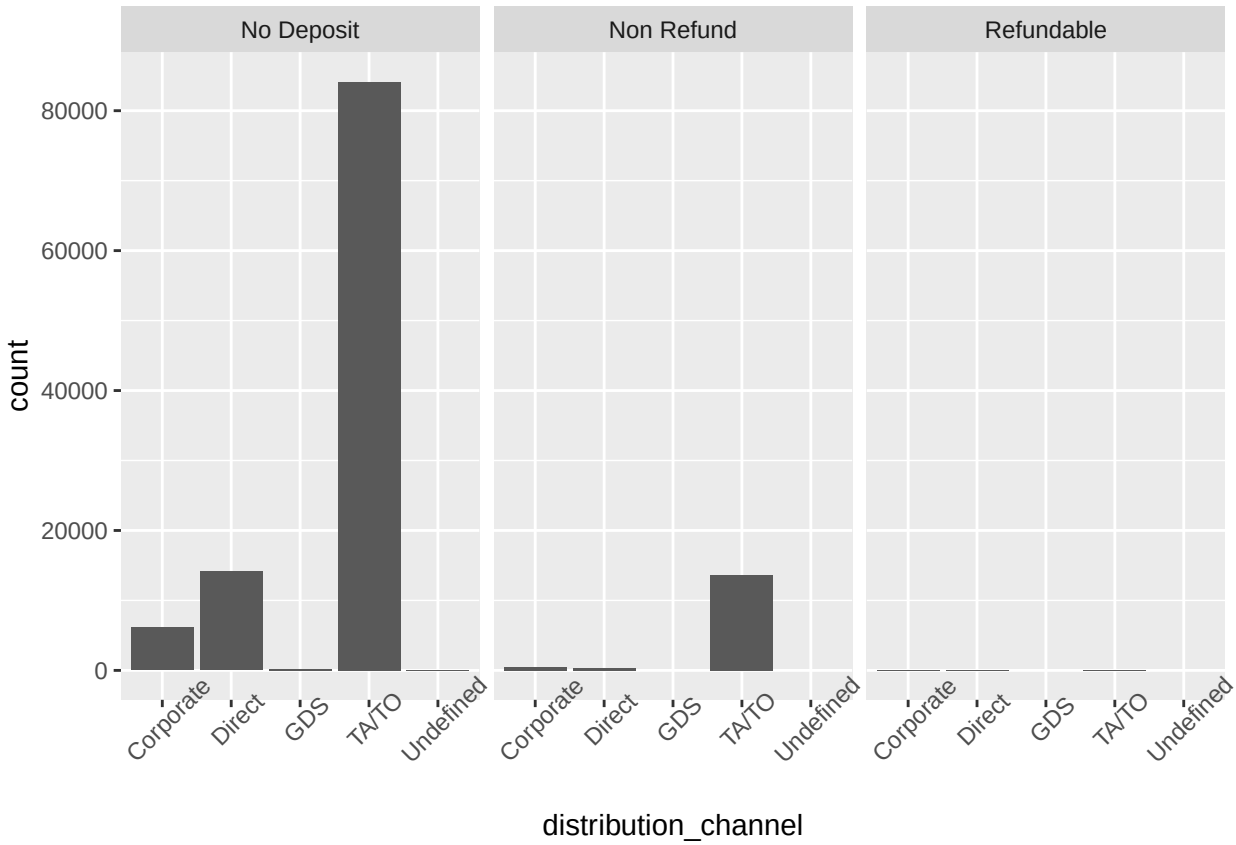
```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel)) +
  facet_wrap(~deposit_type)
```



This code chunk creates three bar charts for 'no_deposit', non_refund', and 'refundable' deposit types. You notice that it's hard to read the x-axis labels here, so you add one piece of code at the end that rotates the text to 45 degrees to make it easier to read.

Try it out below:

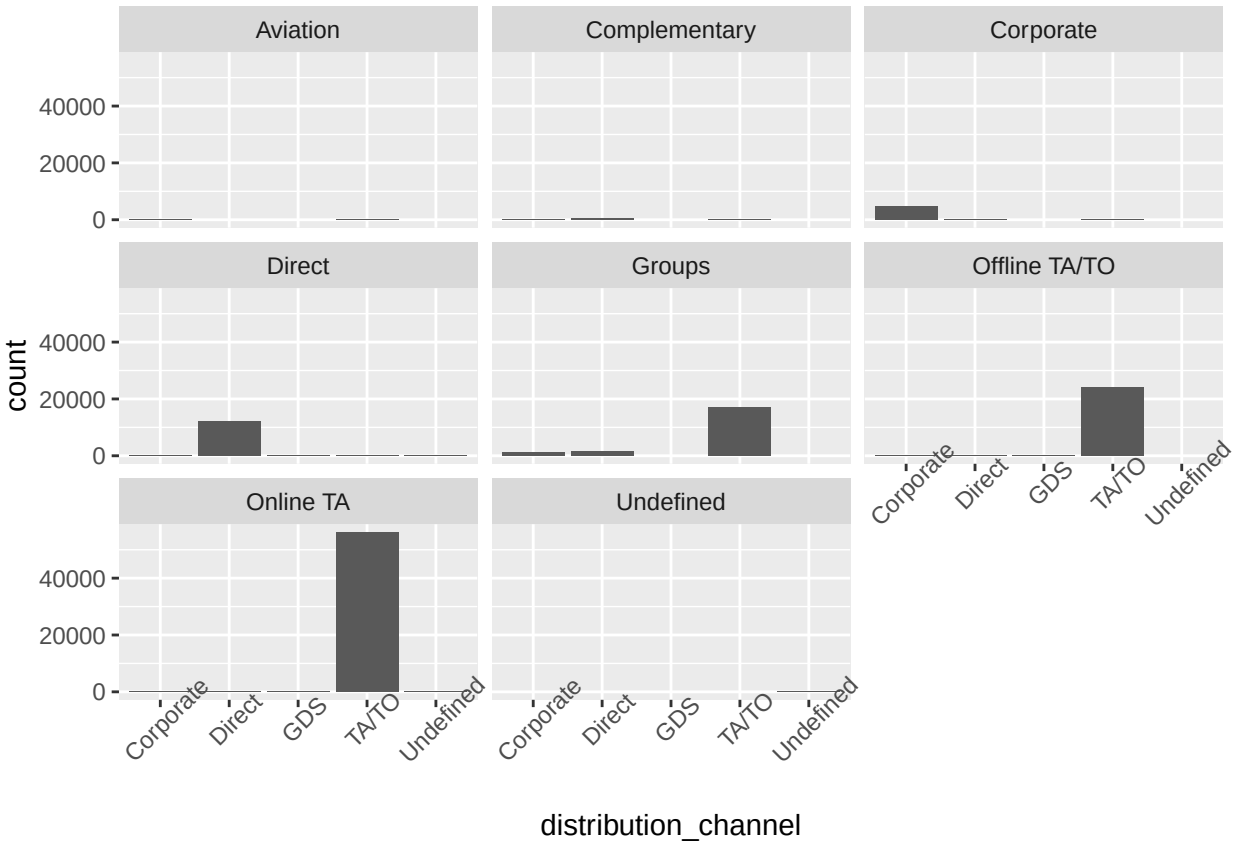
```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel)) +
  facet_wrap(~deposit_type) +
  theme(axis.text.x = element_text(angle = 45))
```



This code chunk creates a similar bar chart to the previous chunk, but now the labels on the x axis with the different distribution channels are clearer.

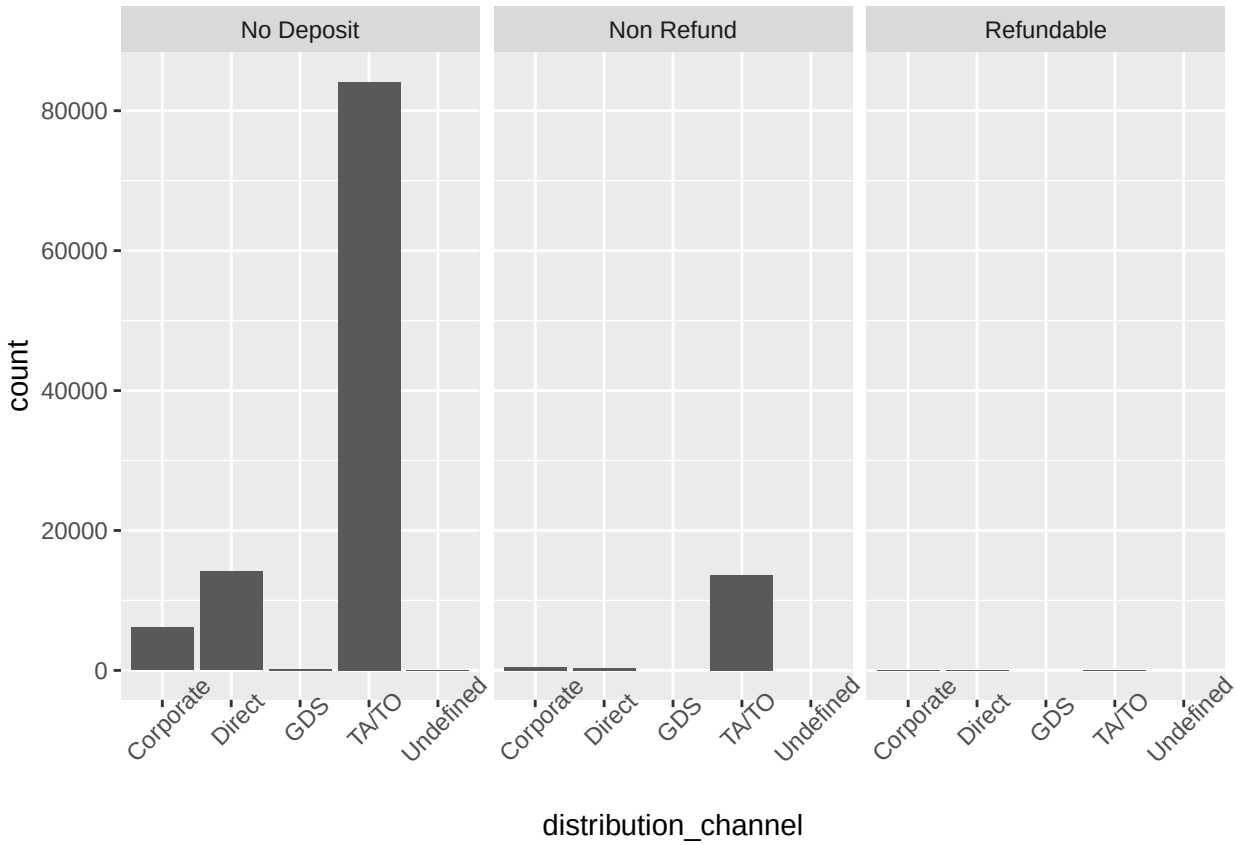
We can use the same syntax to create a different chart for each market segment:

```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel)) +
  facet_wrap(~market_segment) +
  theme(axis.text.x = element_text(angle = 45))
```



The `facet_grid` function does something similar. The main difference is that `facet_grid` will include plots even if they are empty. Run the code chunk below to check it out:

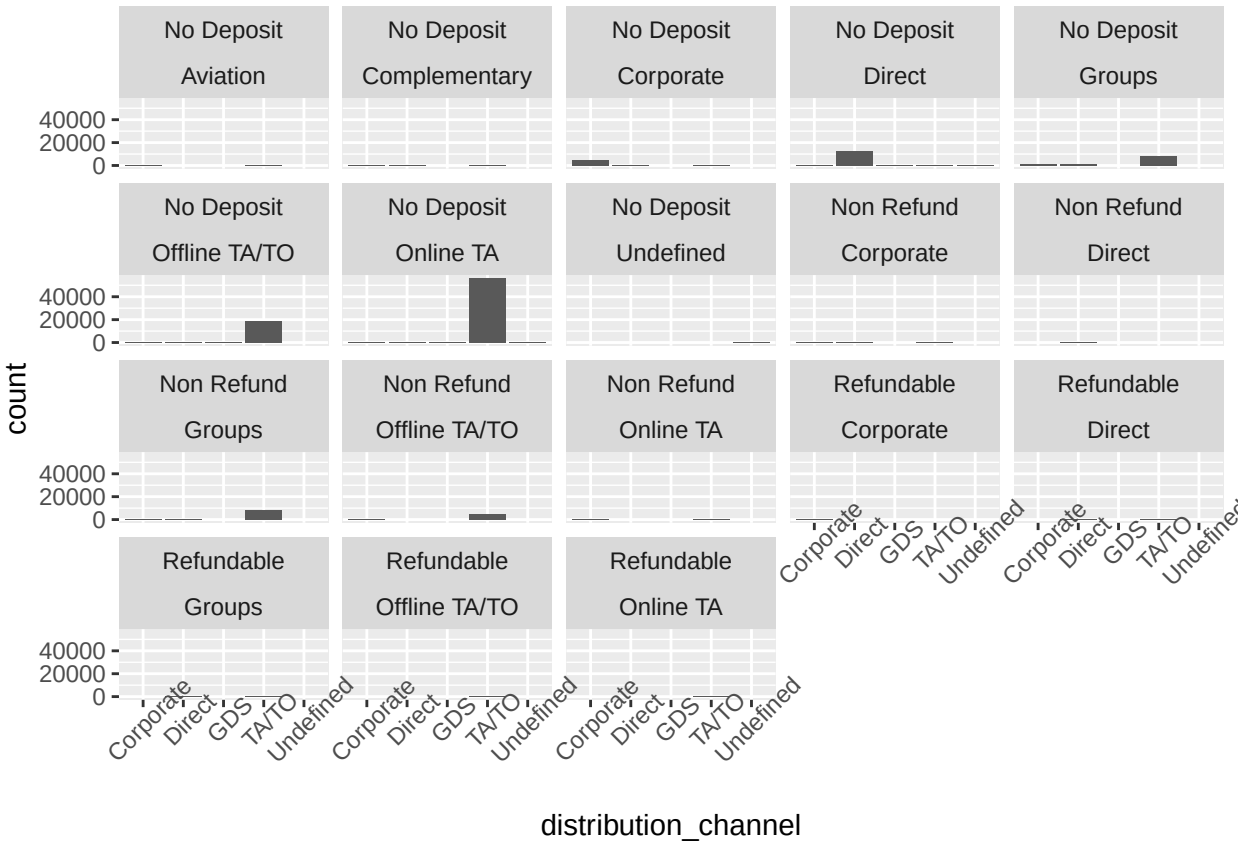
```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel)) +
  facet_grid(~deposit_type) +
  theme(axis.text.x = element_text(angle = 45))
```



Now, you could put all of this in one chart and explore the differences by deposit type and market segment.

Run the code chunk below to find out; notice how the ~ character is being used before the variables that the chart is being split by:

```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = distribution_channel)) +
  facet_wrap(~deposit_type~market_segment) +
  theme(axis.text.x = element_text(angle = 45))
```



Filtering

Install the `tidyverse` package.

```
install.packages('tidyverse')
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## v purrr   0.3.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

A plot that shows the relationship between lead time and guests traveling with children for online bookings at city hotels.

For the first step, we can use the `filter()` function to create a data set that only includes the data we want.

```
onlineta_city_hotels <- filter(hotel_bookings,
                           (hotel=="City Hotel" &
                             hotel_bookings$market_segment=="Online TA"))
```

Note that we can use the '&' character to demonstrate that we want two different conditions to be true. Also, we can use the '$' character to specify which column in the data frame 'hotel_bookings' you are referencing (for example, 'market_segment').

You name this data frame `onlineta_city_hotels_v2`:

```
onlineta_city_hotels_v2 <- hotel_bookings %>%
  filter(hotel=="City Hotel") %>%
  filter(market_segment=="Online TA")
```

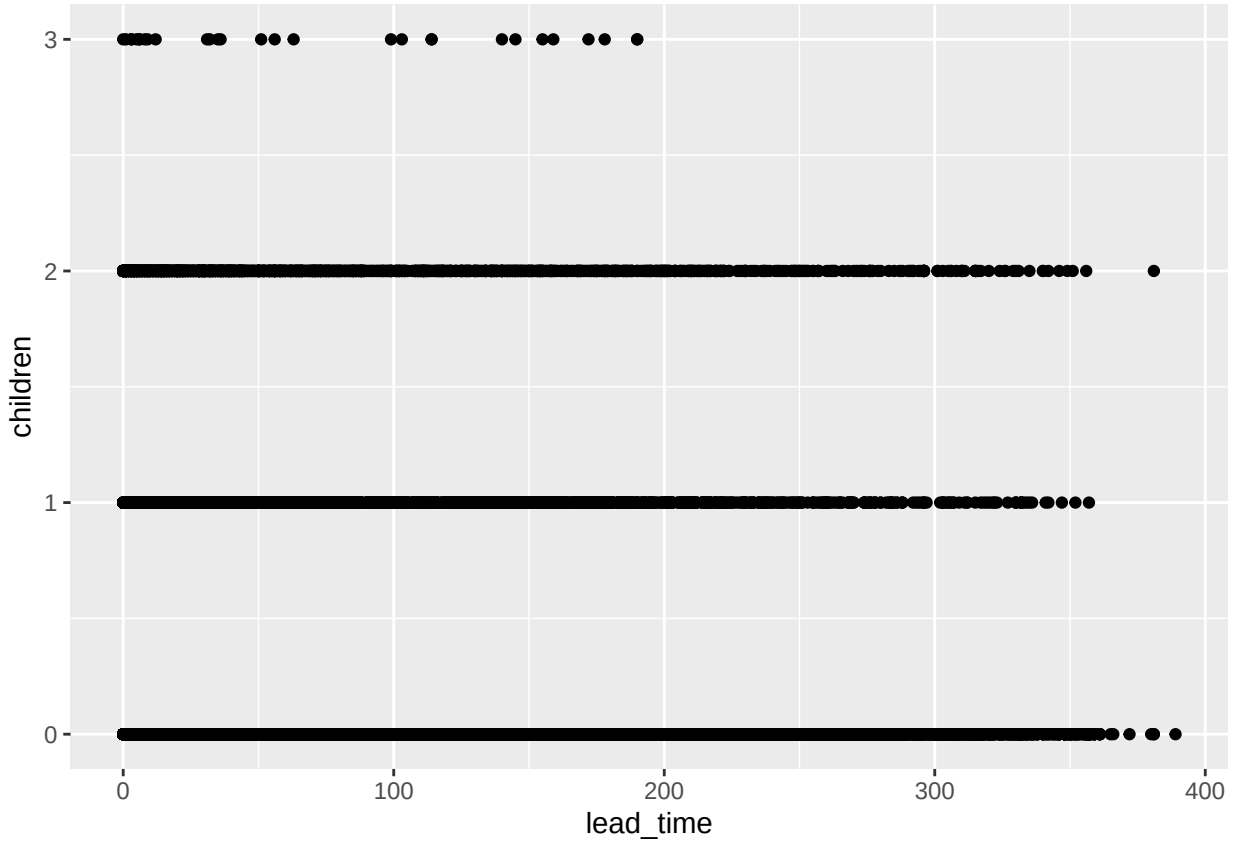
Notice how in the code chunk above, the %>% symbol is used to note the logical steps of this code. First, it starts with the name of the data frame, `onlineta_city_hotels_v2`, AND THEN it tells `R` to start with the original data frame `hotel_bookings`. Then it tells it to filter on the 'hotel' column; finally, it tells it to filter on the 'market_segment' column.

Use your new dataframe

Using the code for scatterplot, replace `variable_name` in the code chunk below with either `onlineta_city_hotels` or `onlineta_city_hotels_v2` to plot the data.

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

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



Based on your previous filter, this scatterplot shows data for online bookings for city hotels. The plot reveals that bookings with children tend to have a shorter lead time, and bookings with 3 children have a significantly shorter lead time (<200 days). So, promotions targeting families can be made closer to the valid booking dates.

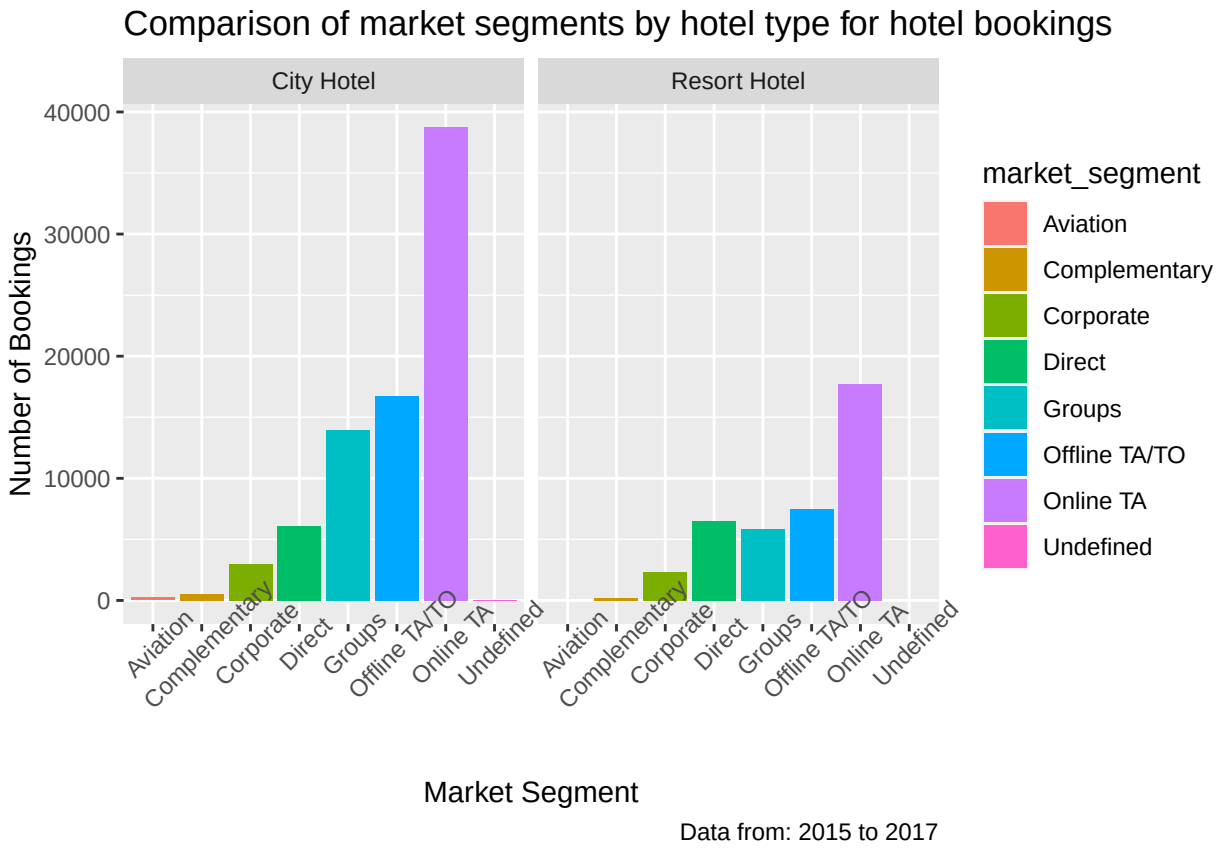
Annotating chart

Create a visualization that compares market segments between city hotels and resort hotels. This will help inform how the company targets promotions in the future.

```
mindate <- min(hotel_bookings$arrival_date_year)
```

```r
maxdate <-max(hotel_bookings$arrival_date_year)
```

```
ggplot(data = hotel_bookings) +
  geom_bar(mapping = aes(x = market_segment, fill = market_segment)) +
  facet_wrap(~hotel) +
  theme(axis.text.x = element_text(angle = 45)) +
  labs(title="Comparison of market segments by hotel type for hotel bookings",
       caption=paste0("Data from: ", mindate, " to ", maxdate),
       x="Market Segment",
       y="Number of Bookings")
```



Saving your chart

Now, it's time to save chart.

Use the `ggsave()` function to do just that! It will save your image as a 7x7 at the file path input by default, which makes it simple to export plots from R.

The `ggsave()` function in the code chunk below will save the last plot that was generated, so if you ran something after running the code chunk above, run that code chunk again.

Then run the following code chunk to save that plot as a .png file named `hotel_booking_chart`.

```
ggsave('hotel_booking_chart.png', width=7,
       height=7)
```