Spotify Exploratory Data Analysis

Seth Goldman

November 3, 2019

Overview

Requested my personal data from Spotify, which took a couple of days. The data was made available via a zip folder of JSON files. Documentation from Spotify

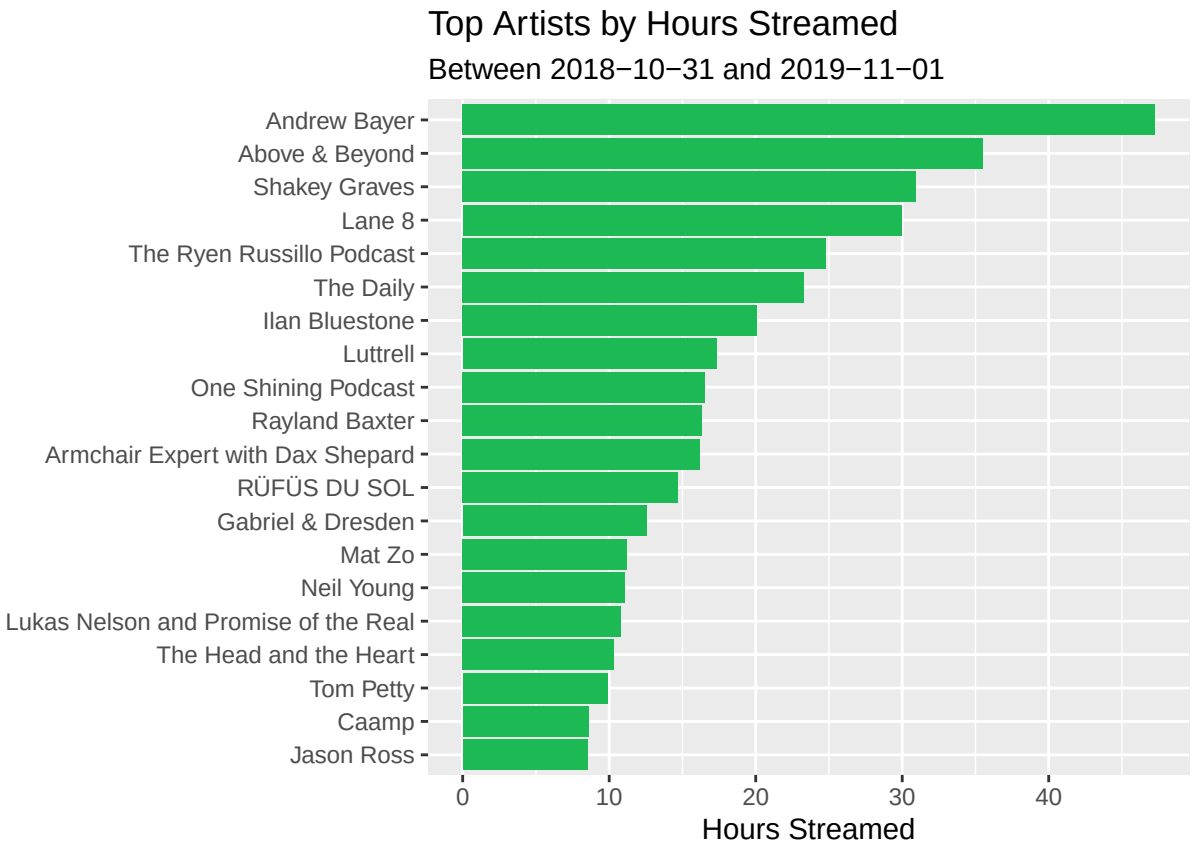
Summary Statistics

```
# Summary Statistics
skim(streaming)
```

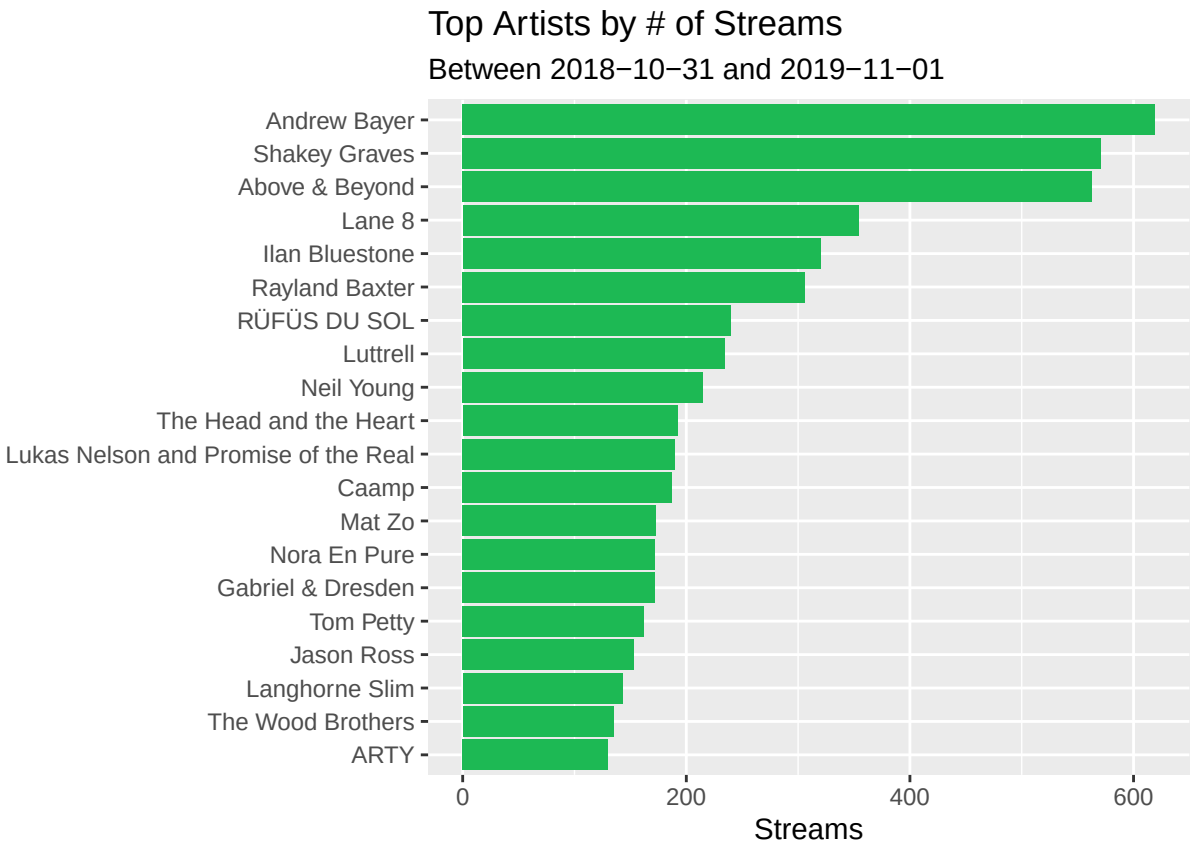
```
## Skim summary statistics
##  n obs: 16419
##  n variables: 4
##
## -- Variable type:character -------------------------------------
##    variable missing complete     n min max empty n_unique
##  artistName       0    16419 16419   2  47     0     1710
##   trackName       0    16419 16419   2 141     0     5442
##
## -- Variable type:integer ---------------------------------------
##  variable missing complete     n      mean        sd p0      p25    p50
##  msPlayed       0    16419 16419 228310.59 256874.55  0 166241.5 219374
##     p75    p100     hist
##  268271 7436082 <U+2587><U+2581><U+2581><U+2581><U+2581><U+2581><U+2581><U+2581>
##
## -- Variable type:POSIXct ---------------------------------------
##  variable missing complete     n        min        max     median n_unique
##   endTime       0    16419 16419 2018-10-31 2019-11-01 2019-03-06    14876
```

Most Played Artists

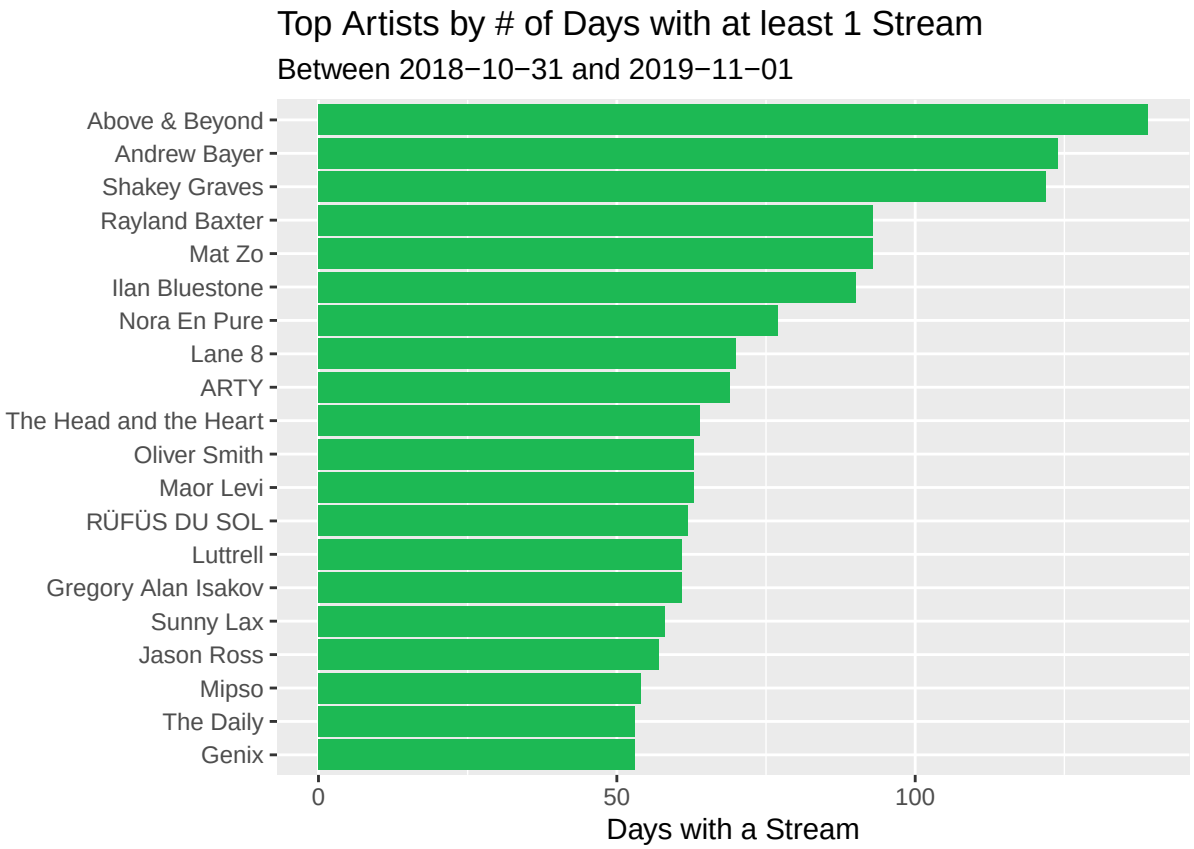
```
artist_summary <- streaming %>%
  group_by(artistName) %>%
  summarise("Streams" = n(),
            "Minutes Streamed" = round(sum(msPlayed)/60000,2),
            "Hours Streamed" = round(`Minutes Streamed`/60,2),
            "Days with a Stream" = n_distinct(as.Date(endTime)))


artist_summary %>%
  top_n(n = 20, wt = `Hours Streamed`) %>%
  ggplot(aes(x=reorder(artistName,`Hours Streamed`), y = `Hours Streamed`))+
  geom_bar(stat = "identity", fill = "#1DB954")+
  coord_flip() +
  xlab("")+
  ggtitle("Top Artists by Hours Streamed",
          subtitle = sub_title_text)
```




```
artist_summary %>%
  top_n(n = 20, wt = Streams) %>%
  ggplot(aes(x=reorder(artistName,Streams), y = Streams))+
  geom_bar(stat = "identity", fill = "#1DB954")+
  coord_flip() +
  xlab("")+
  ggtitle("Top Artists by # of Streams",
          subtitle = sub_title_text)
```

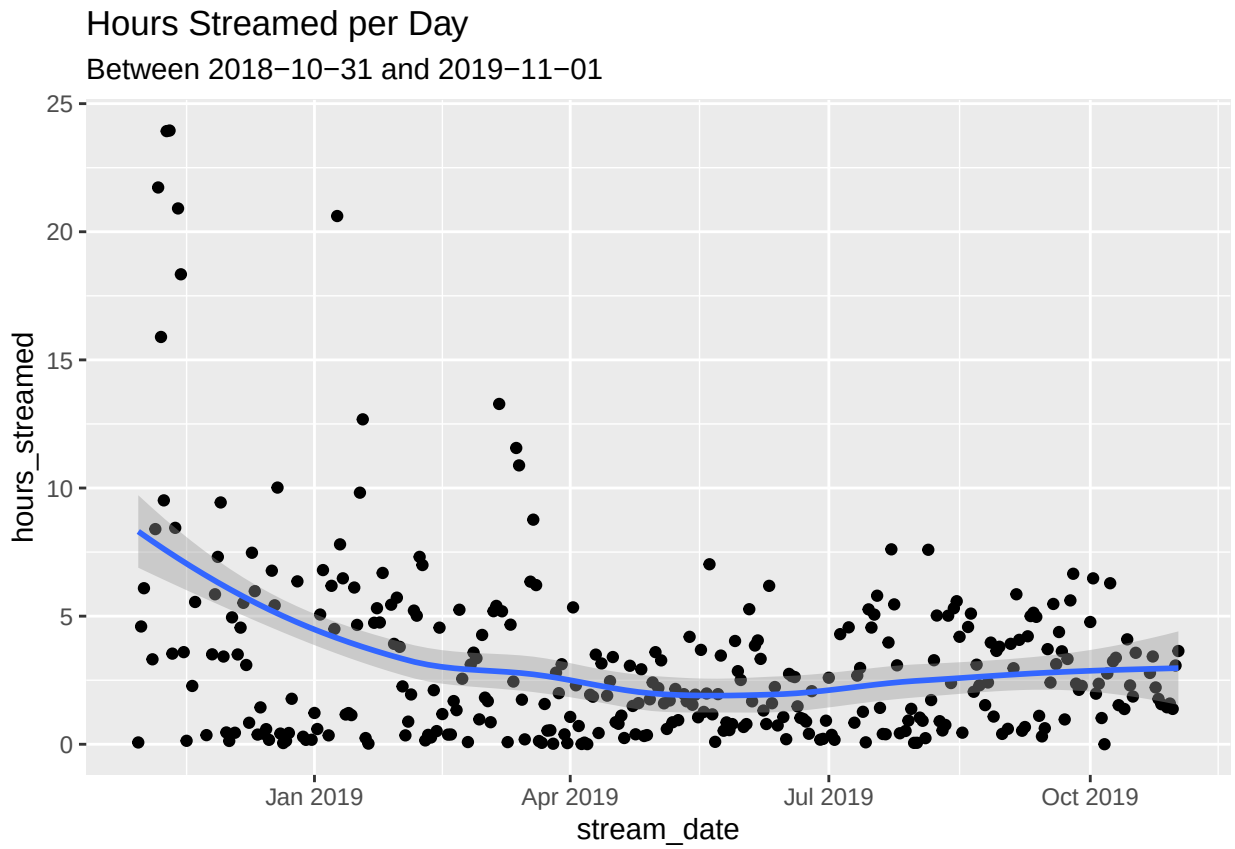



```
artist_summary %>%
  top_n(n = 20, wt = `Days with a Stream`) %>%
  ggplot(aes(x=reorder(artistName,`Days with a Stream`), y = `Days with a Stream`))+
  geom_bar(stat = "identity", fill = "#1DB954")+
  coord_flip() +
  xlab("")+
  ggtitle("Top Artists by # of Days with at least 1 Stream",
          subtitle = sub_title_text)
```



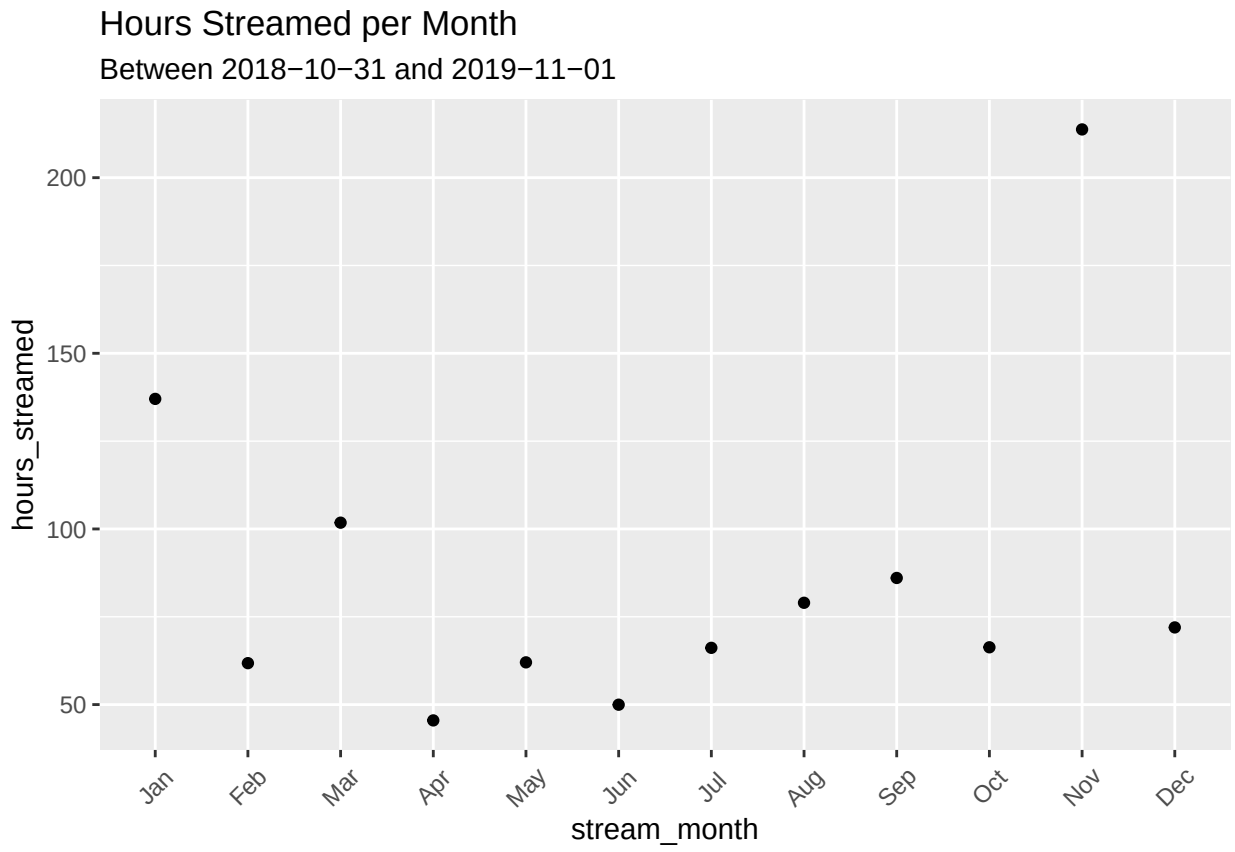

```
streaming %>%
  group_by("stream_date" = as.Date(endTime)) %>%
  summarise("stream_count" = n(),
            "minutes_streamed" = sum(msPlayed)/60000,
            "hours_streamed" = minutes_streamed / 60) %>%
  ggplot(aes(x=stream_date, y = hours_streamed))+
  geom_point()+
  geom_smooth()+
  ggtitle("Hours Streamed per Day",
          subtitle = sub_title_text)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

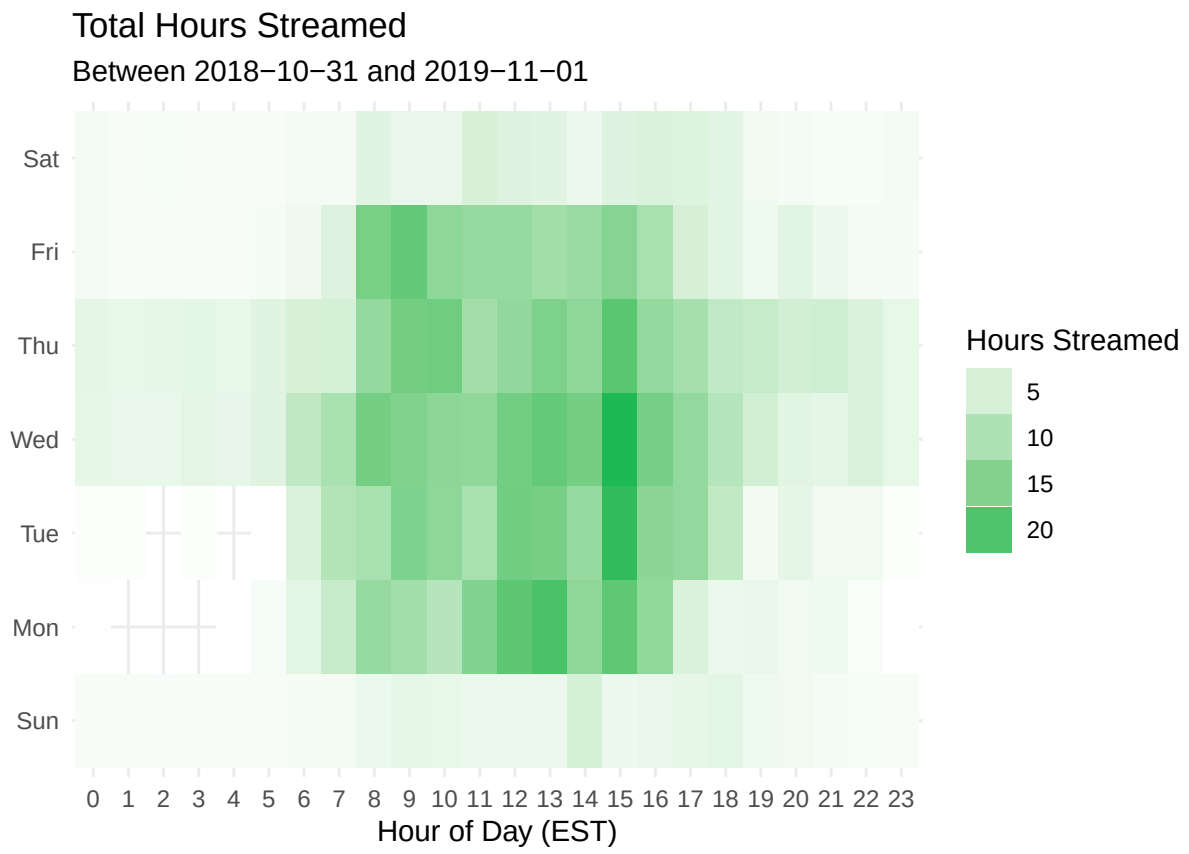



```
streaming %>%
  group_by("stream_month" = factor(month(as.Date(endTime),label = TRUE))) %>%
  summarise("stream_count" = n(),
            "minutes_streamed" = sum(msPlayed)/60000,
            "hours_streamed" = minutes_streamed / 60) %>%
  ggplot(aes(x=stream_month, y = hours_streamed))+
  geom_point()+
  geom_smooth()+
  ggtitle("Hours Streamed per Month",
          subtitle = sub_title_text)+
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

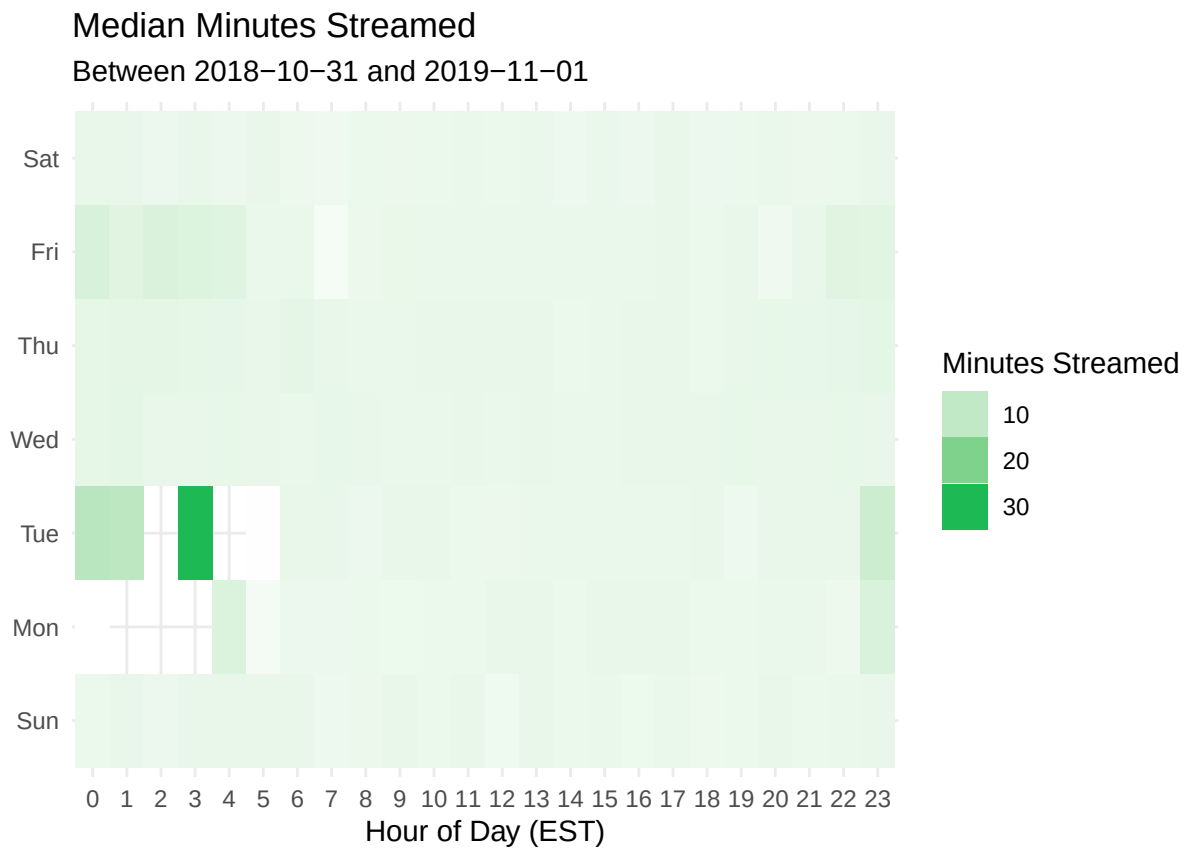
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
time_of_day_summary <-
  streaming %>%
  mutate("day_of_week" = wday(as.Date(endTime),label = TRUE),
         "hour_of_day" = hour(endTime)) %>%
  group_by(day_of_week, hour_of_day) %>%
  summarise("stream_count" = n(),
            "minutes_streamed" = sum(msPlayed)/60000,
            "hours_streamed" = minutes_streamed/60,
            "avg_minutes_streamed" = mean(msPlayed)/60000,
            "median_minutes_streamed" = median(msPlayed)/60000)



time_of_day_summary %>%
  ggplot(aes(x=factor(hour_of_day),y=day_of_week, fill = hours_streamed))+
  geom_tile()+
  scale_fill_gradient(low="white", high = "#1DB954")+
  xlab("Hour of Day (EST)")+
  ylab("")+
  ggtitle("Total Hours Streamed", subtitle = sub_title_text)+
  guides(fill=guide_legend(title="Hours Streamed")) +
  theme_bw()+
  theme_minimal()
```




```
time_of_day_summary %>%
  ggplot(aes(x=factor(hour_of_day),y=day_of_week, fill = median_minutes_streamed))+
  geom_tile()+
  scale_fill_gradient(low="white", high = "#1DB954")+
  xlab("Hour of Day (EST)")+
  ylab("")+
  ggtitle("Median Minutes Streamed", subtitle = sub_title_text)+
  guides(fill=guide_legend(title="Minutes Streamed")) +
  theme_bw()+
  theme_minimal()
```




```
time_of_day_summary %>%
  group_by("is_weekend" = ifelse(day_of_week %in% c("Sat","Sun"), TRUE, FALSE)) %>%
  summarise("stream_count" = sum(stream_count),
            "hours_streamed" = sum(hours_streamed))
```

```
## # A tibble: 2 x 3
##   is_weekend stream_count hours_streamed
##   <lgl>             <int>          <dbl>
## 1 FALSE             14395           934.
## 2 TRUE               2024           108.
```

```
streaming %>%
  group_by(as.Date(endTime)) %>%
  summarise( "hours_streamed" = sum(msPlayed)/60000/60) %>%
  top_n(n = 20,wt = hours_streamed) %>%
  arrange(desc(hours_streamed))
```

```
## # A tibble: 20 x 2
##    `as.Date(endTime)` hours_streamed
##    <date>                      <dbl>
##  1 2018-11-11                  23.9
##  2 2018-11-10                  23.9
##  3 2018-11-07                  21.7
##  4 2018-11-14                  20.9
##  5 2019-01-09                  20.6
##  6 2018-11-15                  18.3
##  7 2018-11-08                  15.9
```

```
##  8 2019-03-07                13.3
##  9 2019-01-18                12.7
## 10 2019-03-13                11.6
## 11 2019-03-14                10.9
## 12 2018-12-19                10.0
## 13 2019-01-17                 9.82
## 14 2018-11-09                 9.52
## 15 2018-11-29                 9.44
## 16 2019-03-19                 8.76
## 17 2018-11-13                 8.45
## 18 2018-11-06                 8.40
## 19 2019-01-10                 7.80
## 20 2019-07-23                 7.61
```