A Tidy Analysis of LAX Holiday Traffic Using Twitter and R

Greg Botwin

SatRday Los Angeles | Apr 6, 2019

About Me and This Short Talk

- Research Associate in the Translational Genomics Group at Cedars-Sinai
- Experience in biology and clinical trials, ~4 years ago failed repeatedly and miserably to learn **R** until I came across the opinionated **tidyverse**
- Mildly addicted to Twitter, lets connect @gjbotwin
- Nothing unites LA like a shared hatred for traffic
- Slides and code available at https://github.com/greg-botwin

```
install.packages("tidyverse", "rtweet")
```

Tidy Data Principles

1. Each variable must have its own column.
2. Each observation must have its own row.
3. Each value must have its own cell.

country	year	cases
Afghanistan	1999	745
Afghanistan	2000	2666
Brazil	1999	37737
Brazil	2000	80488
China	1999	212258
China	2000	213766

country	1999	2000
Afghanistan	745	2666
Brazil	37737	80488
China	212258	213766

table4

[1] Wickham, Hadley. "Tidy data." Journal of Statistical Software 59.10 (2014): 1-23.

Scrapping Twitter Data with rtweet

> #LAX TRAFFIC UPDATE
> (As of 10:00 PM)
> Upper Level : 42 min
> Lower Level: 20 min
> Time to Terminal 1:
> - From Sepulveda/Westchester: 10 min
> - From Century/405: 13 min
> - Via 105 from 405: 9 min#FlyLAX #HolidayTraffic
>
> — LAX Airport (@flyLAXairport) January 3, 2019

```
library(rtweet)
library(tidyverse)

# first time users will need to authenticate
#------ Sat Jan 12 18:08:41 2019 ------#
tl <- get_timeline(user = "flyLAXairport", n = 1000)
df <- tl %>%
  filter(str_detect(text, "LAX TRAFFIC UPDATE")) %>%
  select(text, created_at) %>%
  separate(text, into = paste0("line", seq(1:9)), sep = "\\n")
```

Make Data Tidy

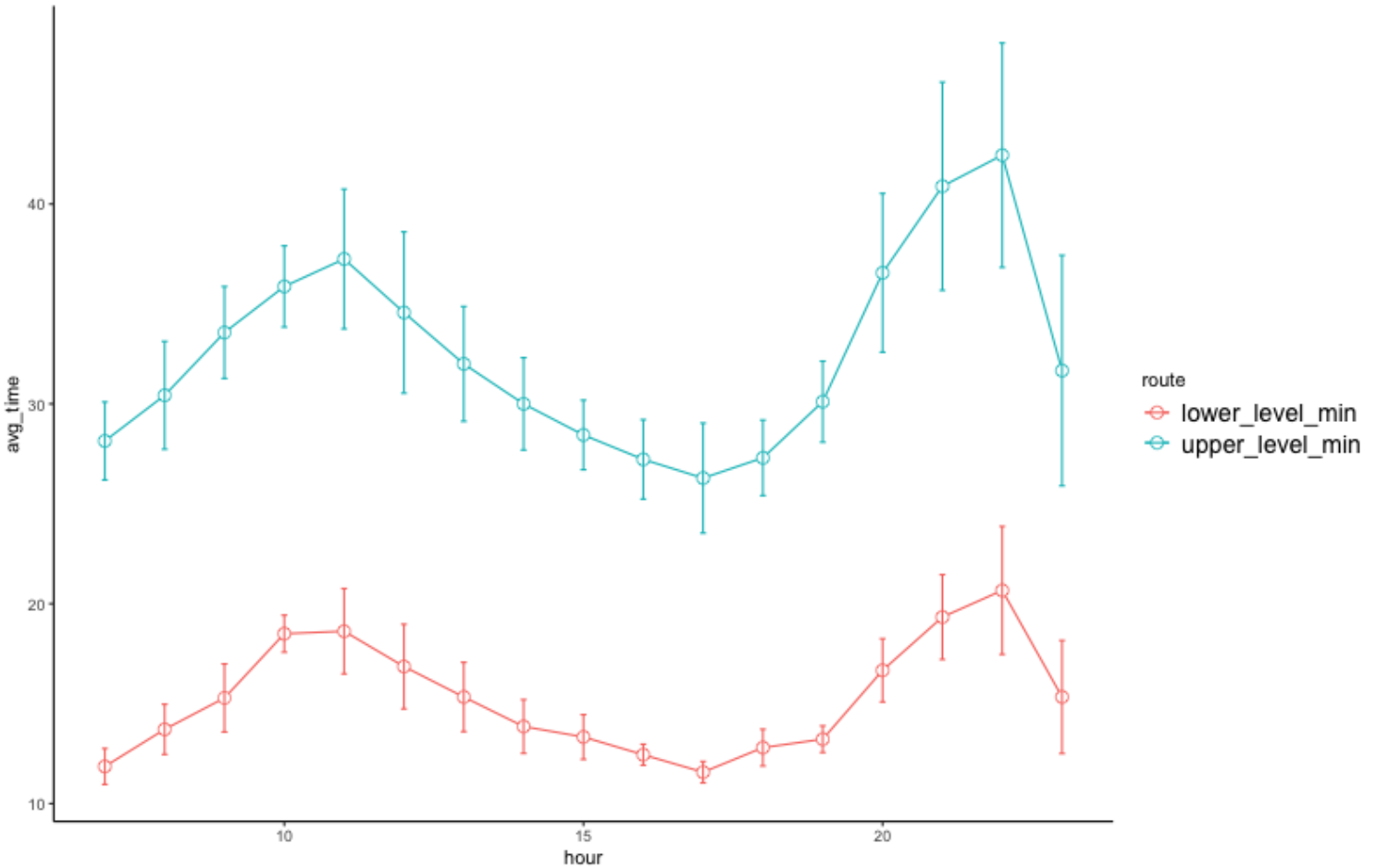
```
df <- df %>%
  mutate(day_of_wk = lubridate::wday(created_at, label = TRUE)) %>%
  mutate(day = date(created_at)) %>%
  mutate(hour = hour(created_at))%>%
  select(day_of_wk, day, hour, upper_level_min, lower_level_min, t1_s
         t1_cent_min, t1_105_min, created_at) %>%
  gather(key = "route", value = "time",-day_of_wk, -day, -hour, -crea
```

```
## # A tibble: 5 x 6
##   day_of_wk day         hour created_at          route            time
##   <ord>     <date>     <int> <dttm>              <chr>           <int>
## 1 Wed       2019-01-02    23 2019-01-02 23:02:54 upper_level_min    27
## 2 Wed       2019-01-02    22 2019-01-02 22:05:00 upper_level_min    42
## 3 Wed       2019-01-02    21 2019-01-02 21:01:10 upper_level_min    39
## 4 Wed       2019-01-02    20 2019-01-02 20:03:05 upper_level_min    37
## 5 Wed       2019-01-02    19 2019-01-02 19:08:33 upper_level_min    30
```

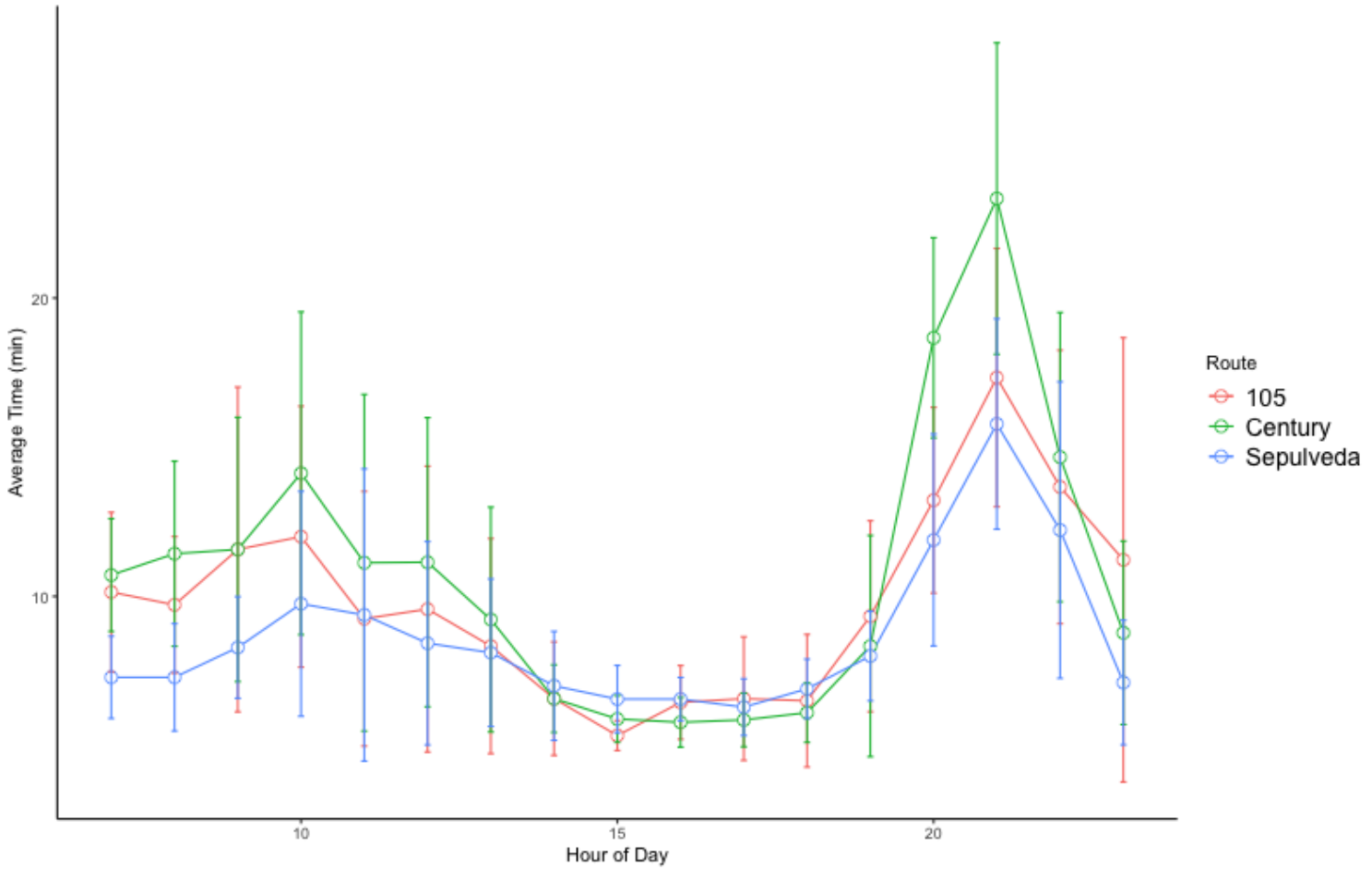
Time to Complete One Loop Around LAX

```
df %>%
  filter(route %in% c("upper_level_min", "lower_level_min")) %>%
  group_by(route, hour) %>%
  summarise(avg_time = mean(time),
            sd_time = sd(time)) %>%
  ggplot(aes(x = hour, y = avg_time, color = route)) +
  geom_line() +
  geom_point(size=3, shape=21, fill="white") +
  geom_errorbar(aes(ymin=avg_time - sd_time,
                    ymax=avg_time + sd_time), width=.1) +
  theme_classic()+
  theme(legend.text=element_text(size=14))
```

Time to Complete One Loop Around LAX



Time to Terminal 1 By Entrance Route

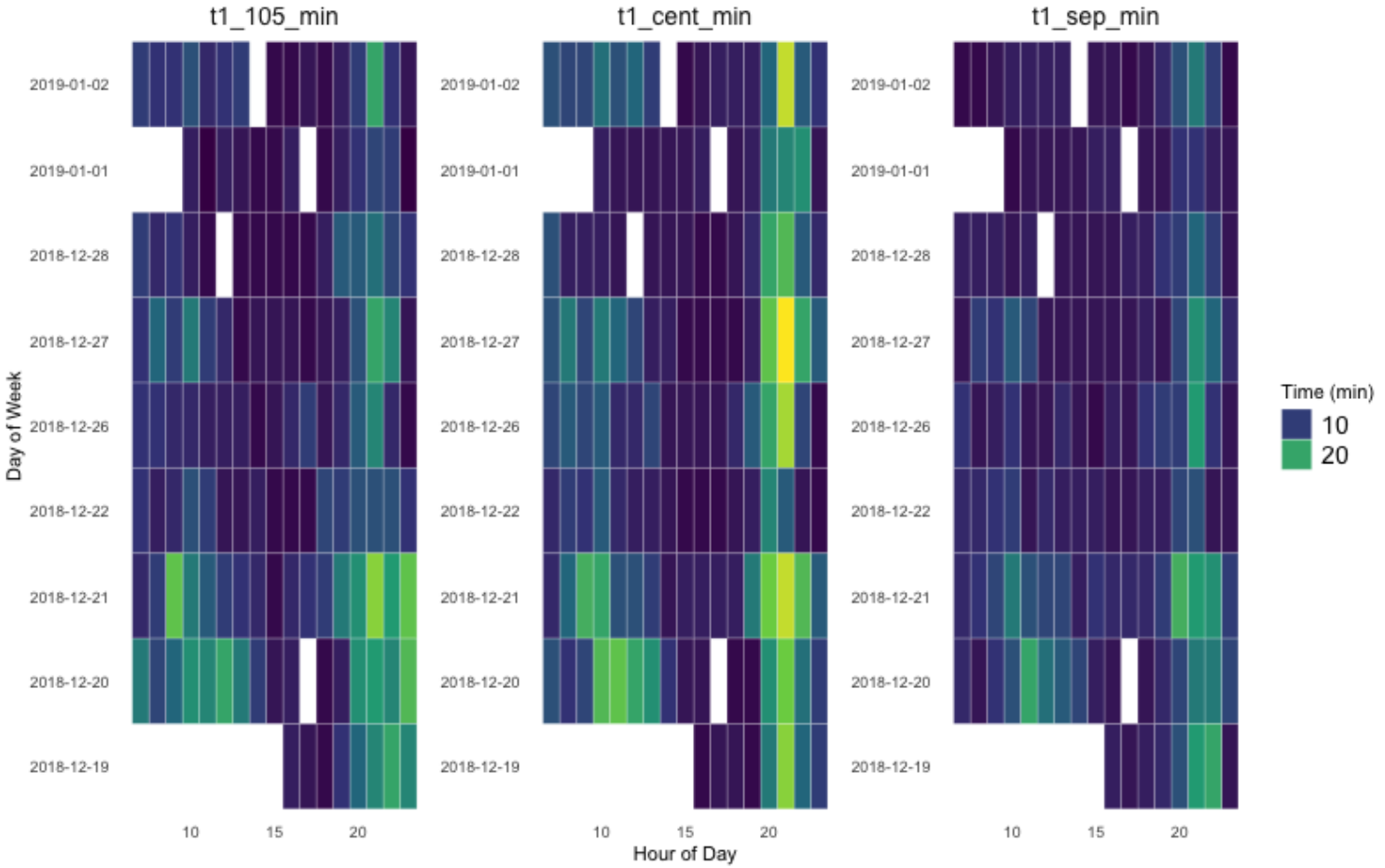


Time to Terminal 1 By Route, Day and Hour

```
#Assign color variables
col1 = "#d8e1cf"
col2 = "#438484"

df %>%
  filter(!route %in% c("upper_level_min", "lower_level_min")) %>%
  ggplot(aes(hour, as.factor(day))) +
  geom_tile(aes(fill = time),colour = "white", na.rm = FALSE) +
  scale_fill_viridis_c()+
  guides(fill=guide_legend(title="Time (min)")) +
  theme_bw() + theme_minimal() +
  labs(y = "Day of Week", x = "Hour of Day") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = elemer
        legend.text=element_text(size=14), strip.text.x = element_tex
  facet_wrap(~route, scales = "free")
```

Time to Terminal 1 By Route, Day and Hour



Conclusions and Thank You

- Tidy data principles can help by outlining a useful data structure
- Try not to drive to LAX during the holidays
- If you have to:

  -- Take the lower loop

  -- Avoid Century Blvd.
- Slides and full code available at https://github.com/greg-botwin
- **Thank you** for your attention!