Summary of Weeks 1-8

Amber Lee

4/27/2020

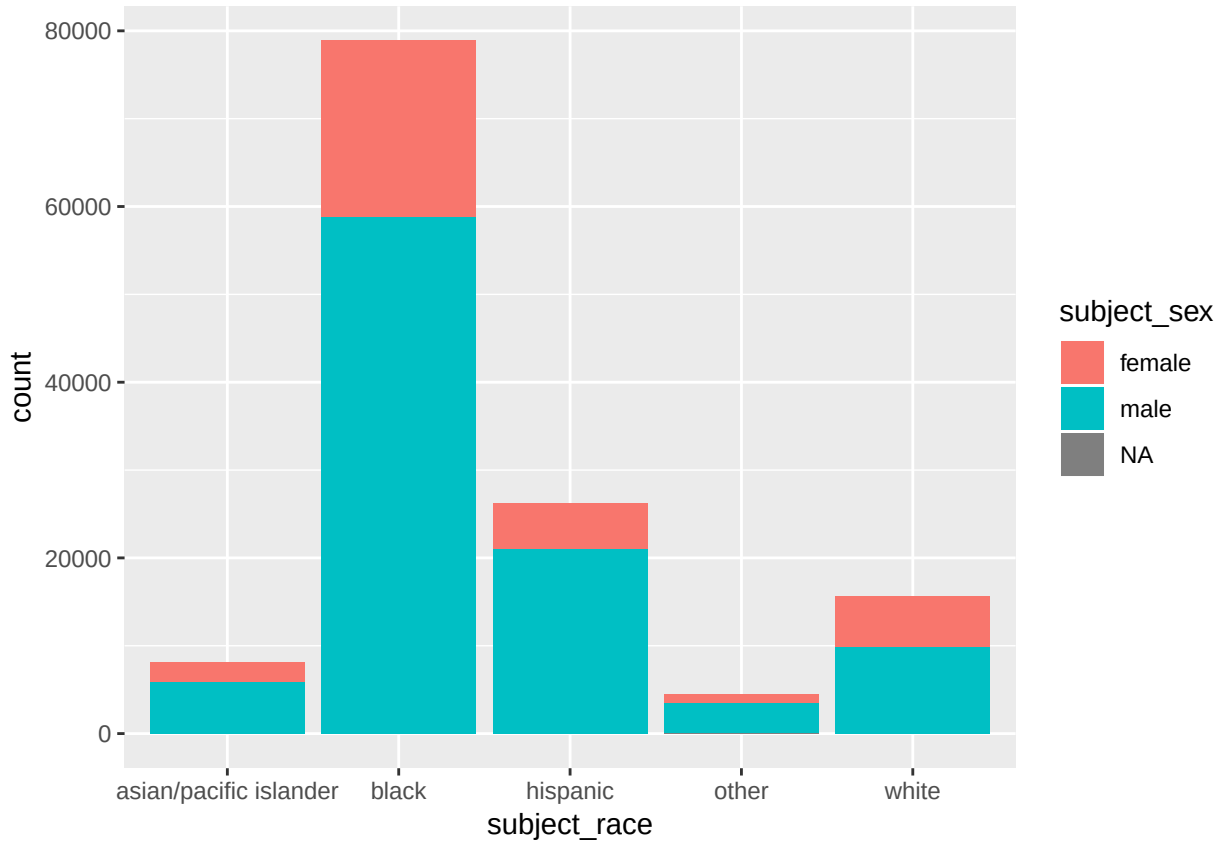
```
OaklandTZ <- lutz::tz_lookup_coords(37.75241810000001, -122.18087990000001, warn = F)
# SanAntonioTZ <- lutz::tz_lookup_coords(29.4241, -98.4936, warn = F)

temp_data <- data.frame(date = CAoak$nice_date, lat = CAoak$lat, lon = CAoak$lng)

getSunlightTimes(data = temp_data, keep = c("sunrise", "sunset"), tz = "America/Los_Angeles") %>% head()
```

```
##         date      lat       lon             sunrise              sunset
## 1 2013-04-01 37.82060 -122.2707 2013-04-01 06:54:53 2013-04-01 19:33:00
## 2 2013-04-01 37.82125 -122.2765 2013-04-01 06:54:54 2013-04-01 19:33:02
## 3 2013-04-01 37.80294 -122.2717 2013-04-01 06:54:54 2013-04-01 19:33:00
## 4 2013-04-01 37.81220 -122.2764 2013-04-01 06:54:54 2013-04-01 19:33:01
## 5 2013-04-01 37.81576 -122.2851 2013-04-01 06:54:56 2013-04-01 19:33:03
## 6 2013-04-01 37.73451 -122.1972 2013-04-01 06:54:38 2013-04-01 19:32:39
```

```
oursunriseset <- function(latitude, longitude, date, direction = c("sunrise", "sunset")) {
  date.lat.long <- data.frame(date = date, lat = latitude, lon = longitude)
  if(direction == "sunrise"){
    getSunlightTimes(data = date.lat.long, keep=direction, tz = OaklandTZ)$sunrise }else{
      getSunlightTimes(data = date.lat.long, keep=direction, tz = OaklandTZ)$sunset } }

# add light variable
CAoak <- CAoak %>%

  # use oursunriseset function to return posixct format sunrise and sunset times
  mutate(sunrise = oursunriseset(lat, lng, nice_date, direction = "sunrise"),
         sunset = oursunriseset(lat, lng, nice_date, direction = "sunset")) %>%

  mutate(light = ifelse(posix_date_time > sunrise & posix_date_time < sunset, "day", "night")) %>%

  # about 100 NA's to filter out

  filter(!is.na(light))
```

Race-related Visualizations

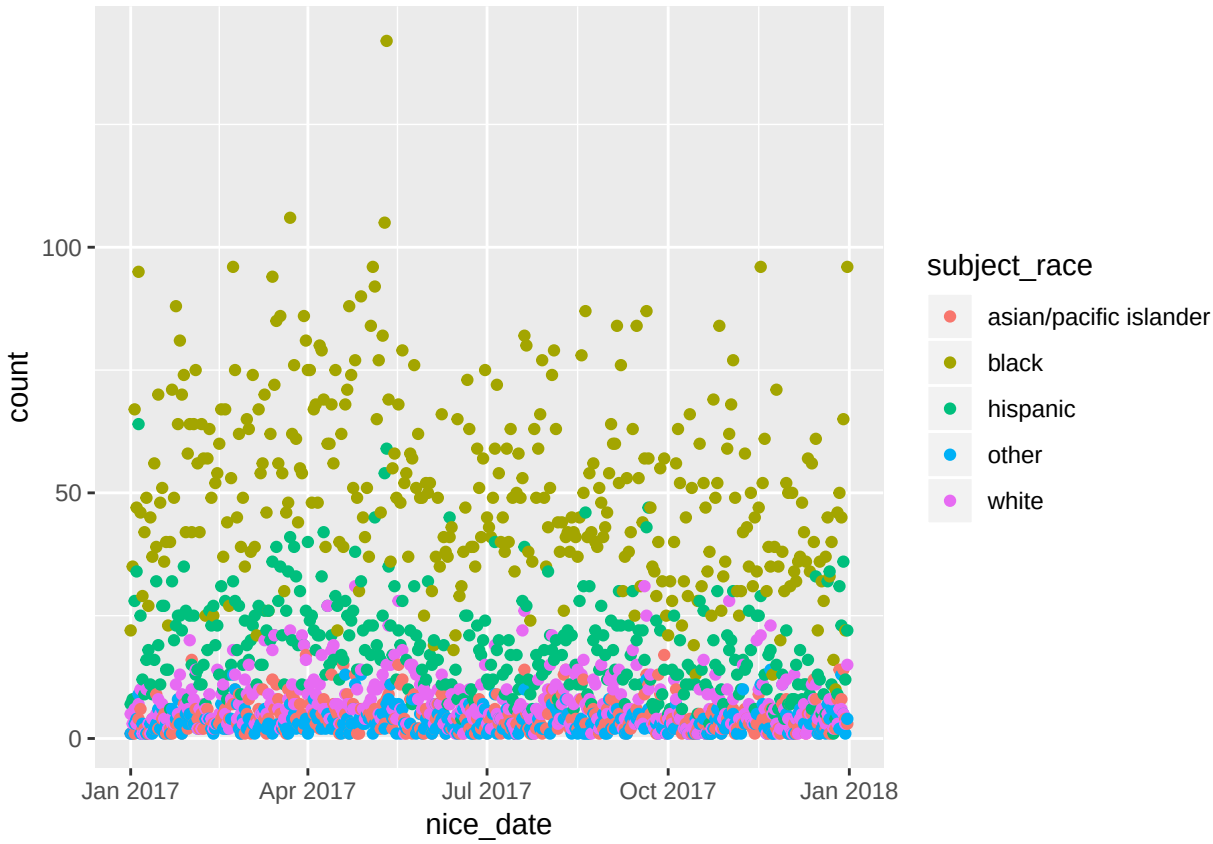
```
# Race and sex

ggplot(data = CAoak) +
  geom_bar(mapping = aes(x = subject_race, fill = subject_sex))
```




```
ggsave("race and sex.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# Race and age

# 102,724 (77%) of the observations *do not* have subject_age recorded
# CAoak %>%
#   group_by(subject_age) %>%
#   summarize(count = n())

## Age is only recorded for the year 2017

CAoak %>%
  filter(!is.na(subject_age)) %>%
  group_by(nice_date, subject_race) %>%
  summarize(count = n()) %>%
  ggplot(aes(x = nice_date, y = count, color = subject_race)) +
  geom_point()
```



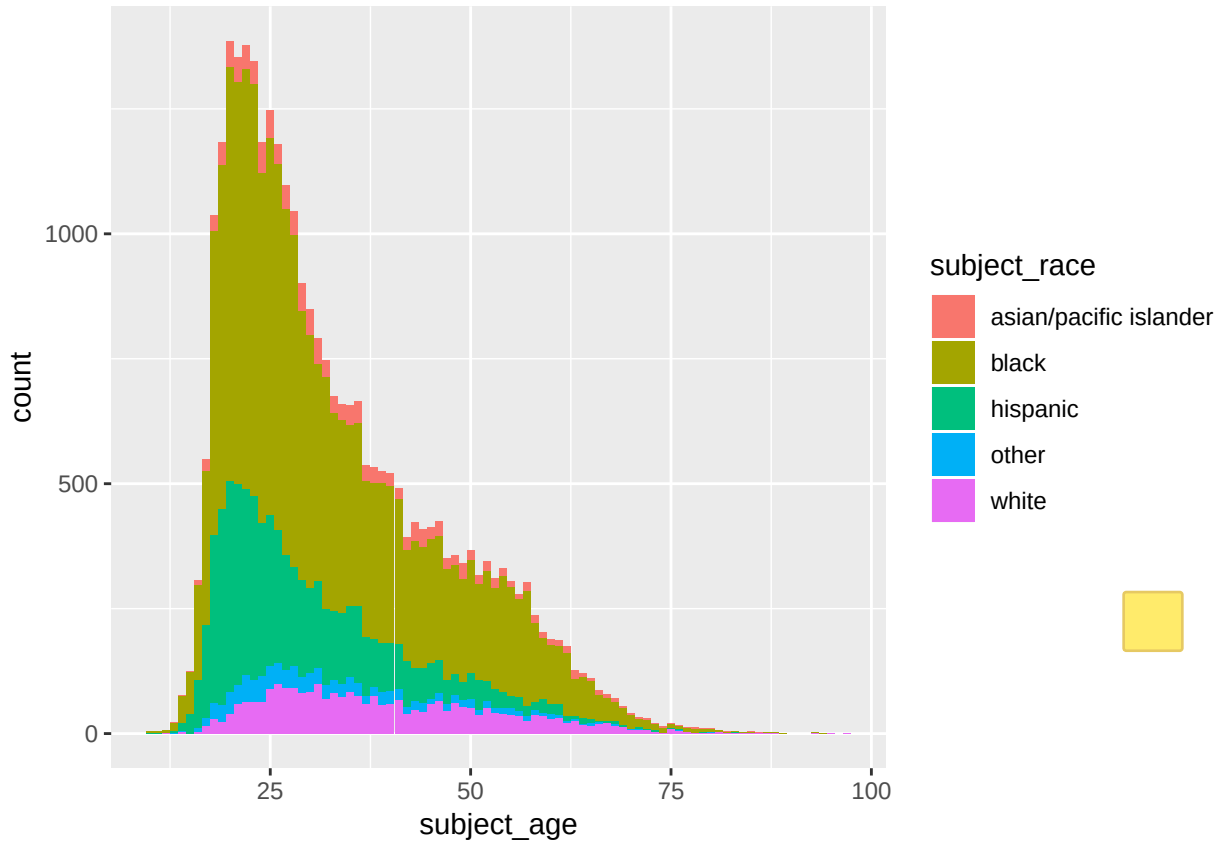

```
# average age of stop for Black and Latinx is lower than for white and API

dbGetQuery(con,
  "SELECT subject_race, AVG(subject_age) AS 'ave age'
  FROM CAoakland
  GROUP BY subject_race
  ORDER BY `ave age`")
```

```
## Warning in .local(conn, statement, ...): Decimal MySQL column 1 imported as
## numeric
```

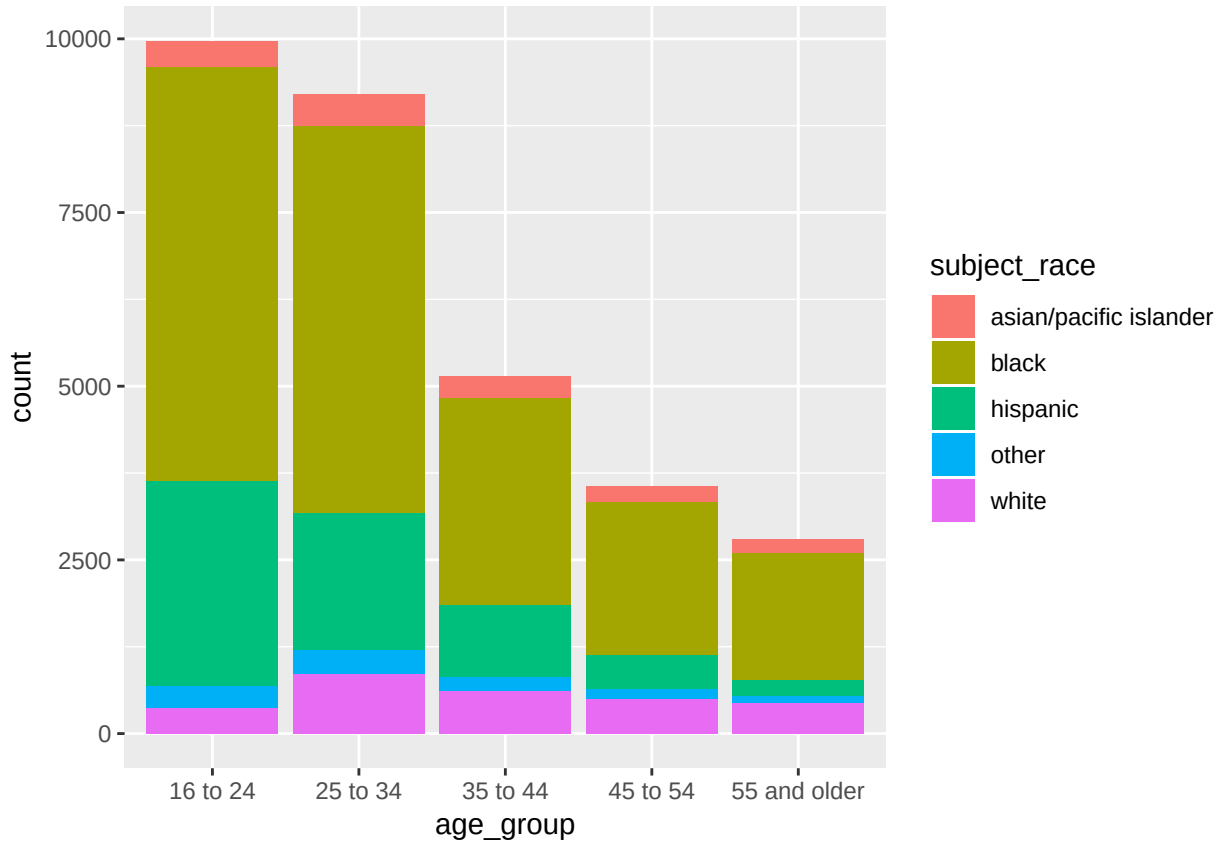
```
##             subject_race ave age
## 1               hispanic 29.1843
## 2                  black 33.4254
## 3                  other 33.8663
## 4 asian/pacific islander 36.2352
## 5                  white 39.3381
```

```
ggplot(data = CAoak) +
  geom_bar(mapping = aes(x = subject_age, fill = subject_race))
```

```
## Warning: Removed 102622 rows containing non-finite values (stat_count).
```



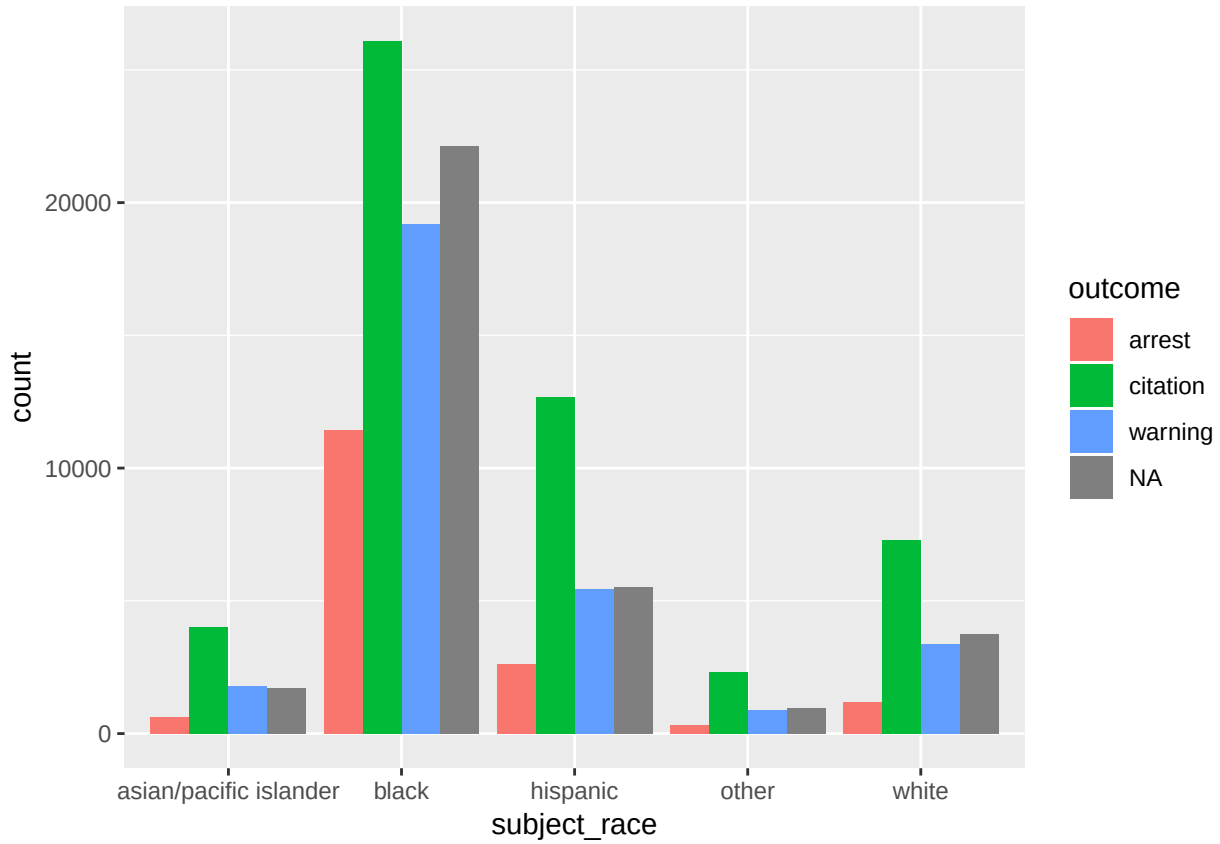

```
CAoak %>%
  filter(!is.na(subject_age)) %>%
  mutate(age_group = case_when(subject_age < 25 ~ "16 to 24",
                               subject_age >= 25 & subject_age < 35 ~ "25 to 34",
                               subject_age >= 35 & subject_age < 45 ~ "35 to 44",
                               subject_age >= 45 & subject_age < 55 ~ "45 to 54",
                               subject_age >= 55 ~ "55 and older")) %>%
  ggplot(aes(x = age_group, fill = subject_race)) +
  geom_bar()
```




```
ggsave("race and subject age groups.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# outcome and race
CAoak %>%
  ggplot()+
  geom_bar(mapping = aes(x = subject_race, fill = outcome), position = "dodge")
```

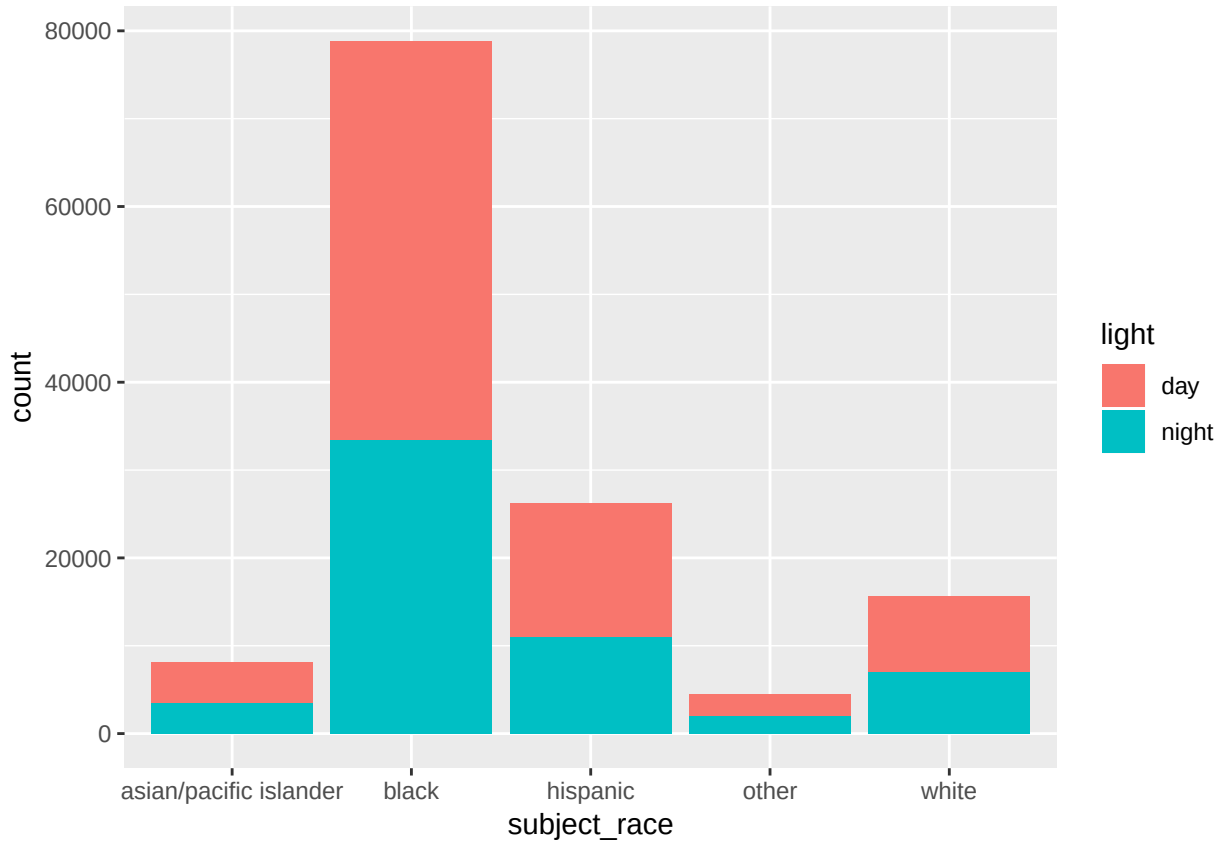



```
ggsave("outcome and race.png")
```

```
## Saving 6.5 x 4.5 in image
```

Incorporating Day/Night Variable in race analysis

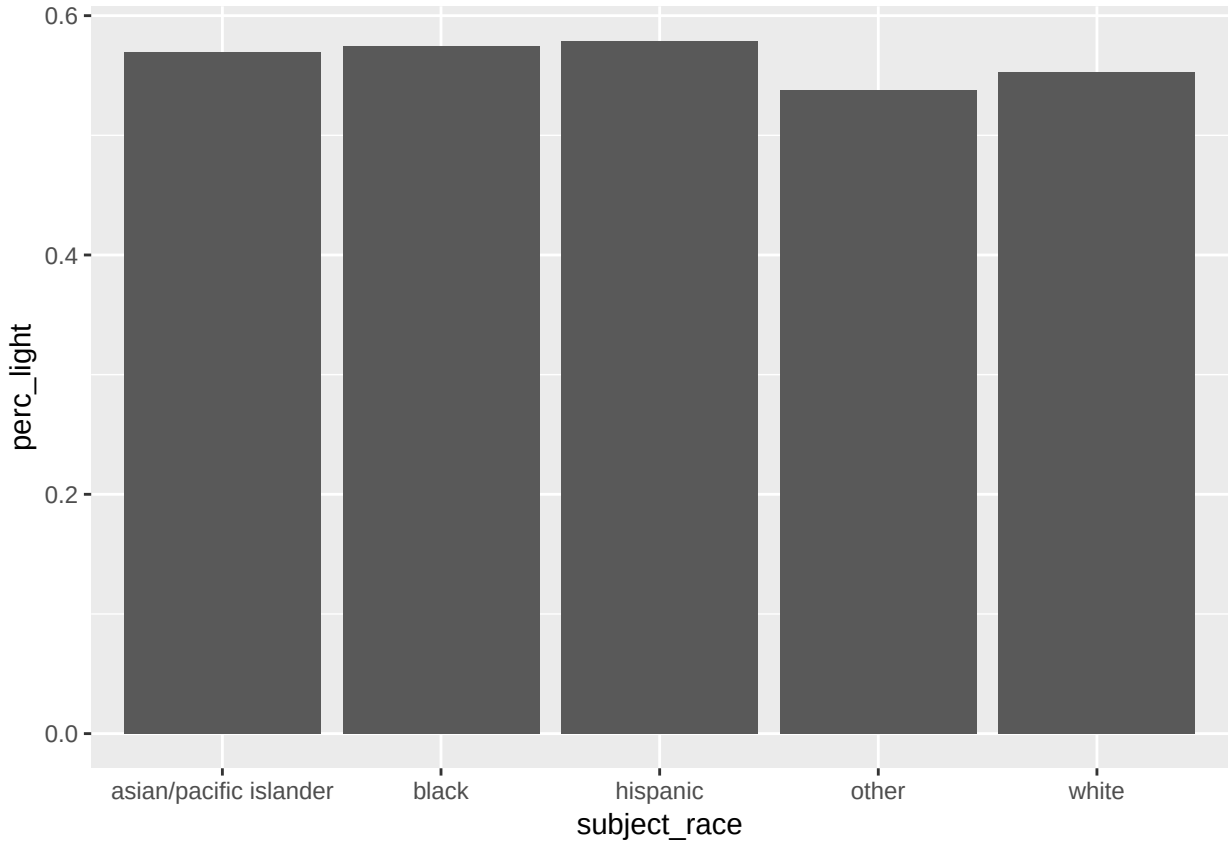
```
# Day/night

# see how drivers are stopped by race and light in absolute counts
ggplot(data = CAoak) +
  geom_bar(mapping = aes(x = subject_race, fill = light))
```




```
ggsave("night and day stop counts by race.png")
```

```
## Saving 6.5 x 4.5 in image
```

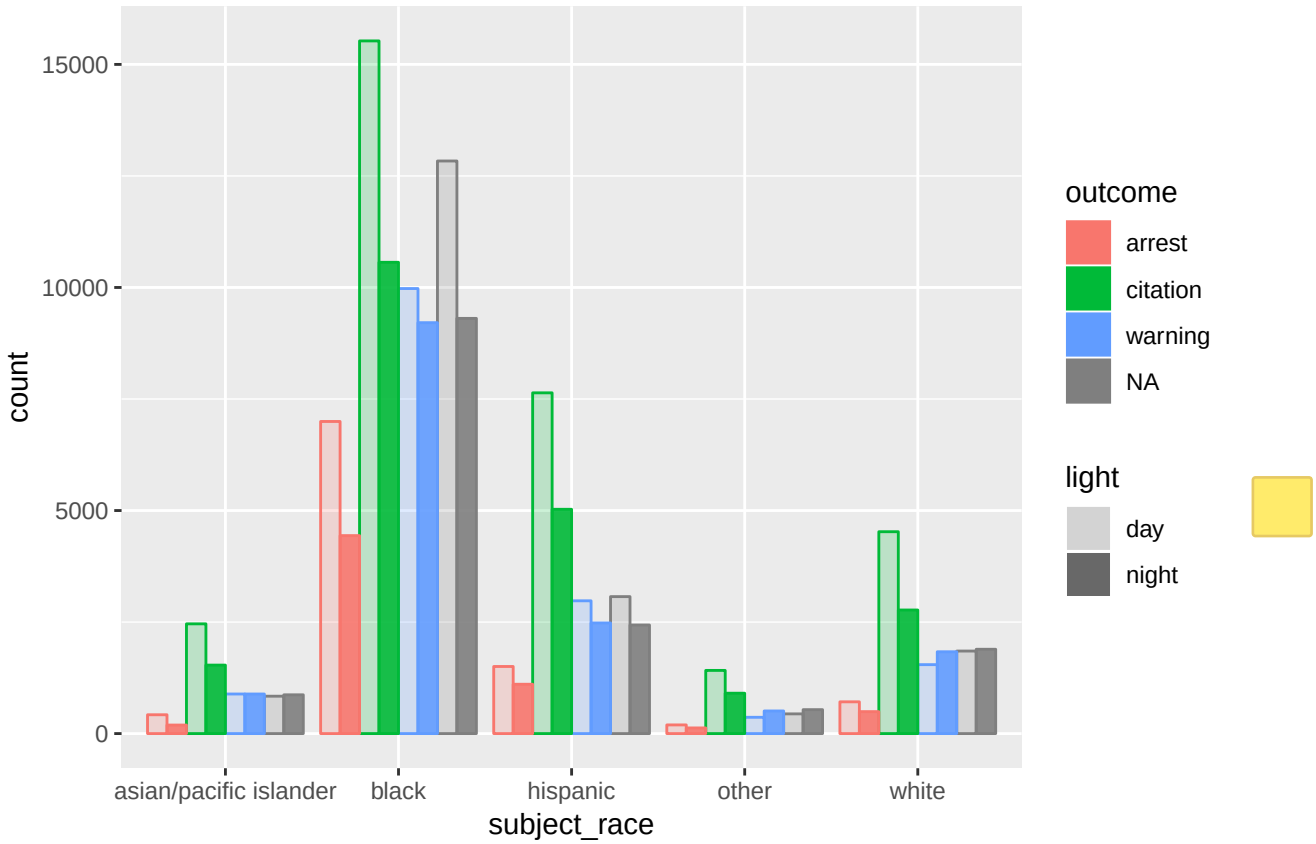
```
# percentage of each racial group stopped during the day
CAoak %>%
  group_by(subject_race, light) %>%
  summarize(count = n()) %>%
  spread(key = light, value = count) %>%
  mutate(total_stops = sum(day, night),
         perc_light = day / total_stops) %>%
  ggplot(aes(x = subject_race, y = perc_light)) +
  geom_bar(stat = "identity")
```




```
ggsave("night and day stop percents by race.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# how are outcomes of stops affected by race and time of day?

CAoak %>%
  ggplot(aes(x = subject_race, fill = outcome, color = outcome, alpha = light)) +
  geom_bar(position="dodge") +
  scale_alpha_manual(values=c(.2, .9))
```




```
ggsave("outcome and race and light.png")
```

```
## Saving 6.5 x 4.5 in image
```

Outcomes

find the difference between # of drivers with search conducted - # of drivers with search conducted and arrest made, for each age and race

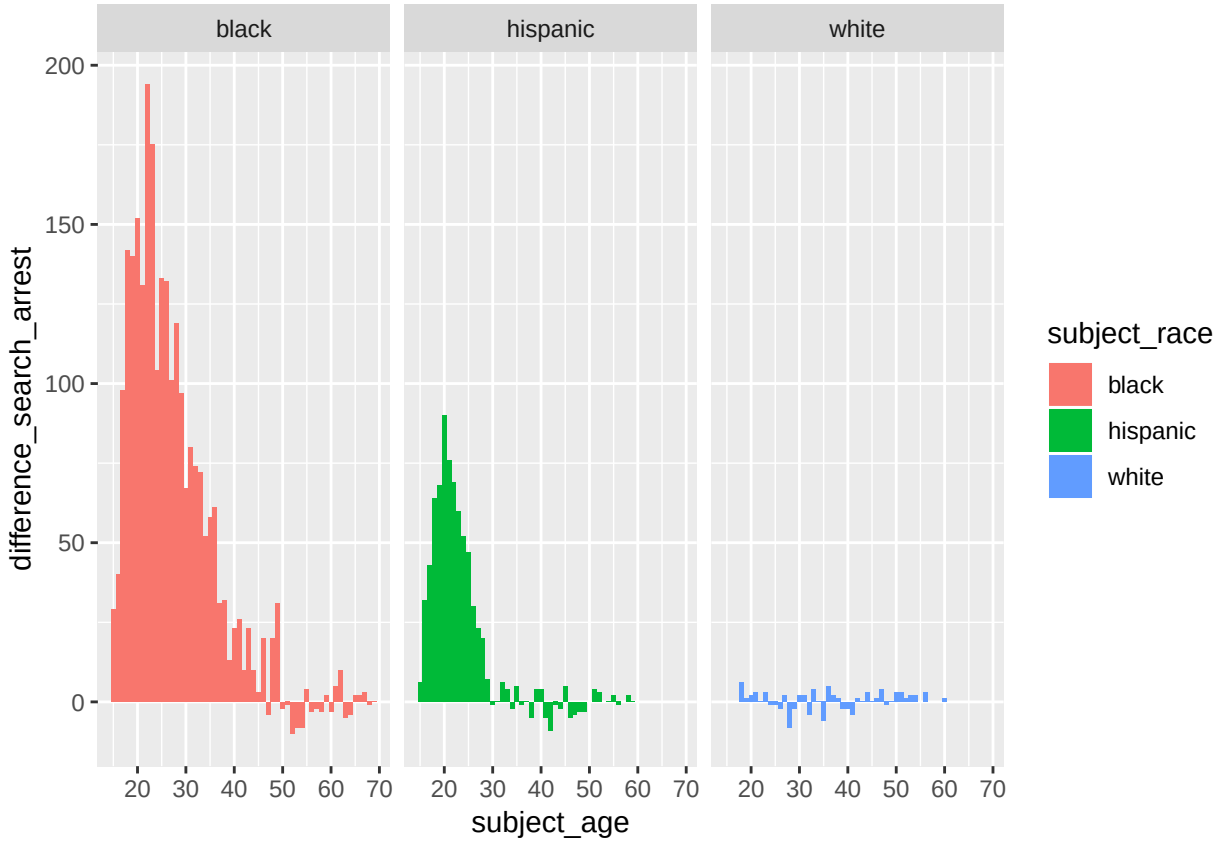
```
# looking at stops that led to at least a search
search_arrests <- CAoak %>%
  filter(search_conducted == 1, subject_race != "asian/pacific islander", subject_race != "other", subj
  group_by(subject_race, subject_age, search_conducted, arrest_made) %>%
  summarize(count = n()) %>%
  mutate(search_only = case_when(search_conducted == 1 & arrest_made == 0 ~ 1,
                                 search_conducted == 1 & arrest_made == 1 ~ 0))

searches_only <- search_arrests %>%
  filter(search_only == 1) %>%
  select(subject_race, subject_age, count)
```

```
## Adding missing grouping variables: `search_conducted`
```

```
arrests_only <- search_arrests %>%
  ungroup() %>%
  filter(search_only == 0) %>%
  select(subject_race, subject_age, count) %>%
  rename(arrest_count = count)
```

```
# absolute numbers
searches_only %>%
  full_join(arrests_only, by = c("subject_race", "subject_age")) %>%
  mutate(difference_search_arrest = count - arrest_count) %>%
  ggplot() +
  geom_bar(aes(x = subject_age, y = difference_search_arrest, fill = subject_race), stat = "identity") +
  facet_wrap(~ subject_race)
```

```
## Warning: Removed 18 rows containing missing values (position_stack).
```



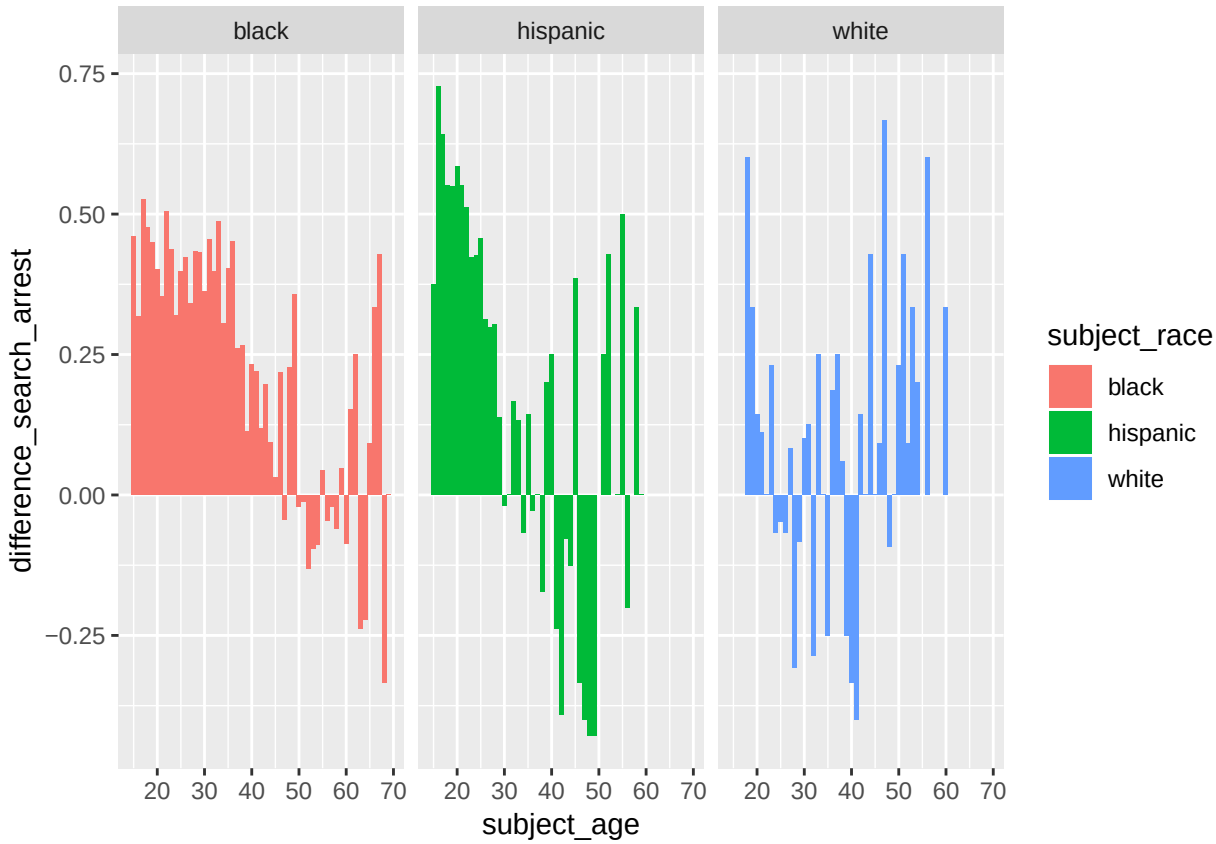

```
ggsave("search and arrest differences absolute.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 18 rows containing missing values (position_stack).
```

```
#percentages
searches_only %>%
  full_join(arrests_only, by = c("subject_race", "subject_age")) %>%
  mutate(difference_search_arrest = (count - arrest_count)/(count+arrest_count)) %>%
  ggplot() +
  geom_bar(aes(x = subject_age, y = difference_search_arrest, fill = subject_race), stat = "identity") +
  facet_wrap(~ subject_race)
```

```
## Warning: Removed 18 rows containing missing values (position_stack).
```




```
ggsave("search and arrest differences percent.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 18 rows containing missing values (position_stack).
```

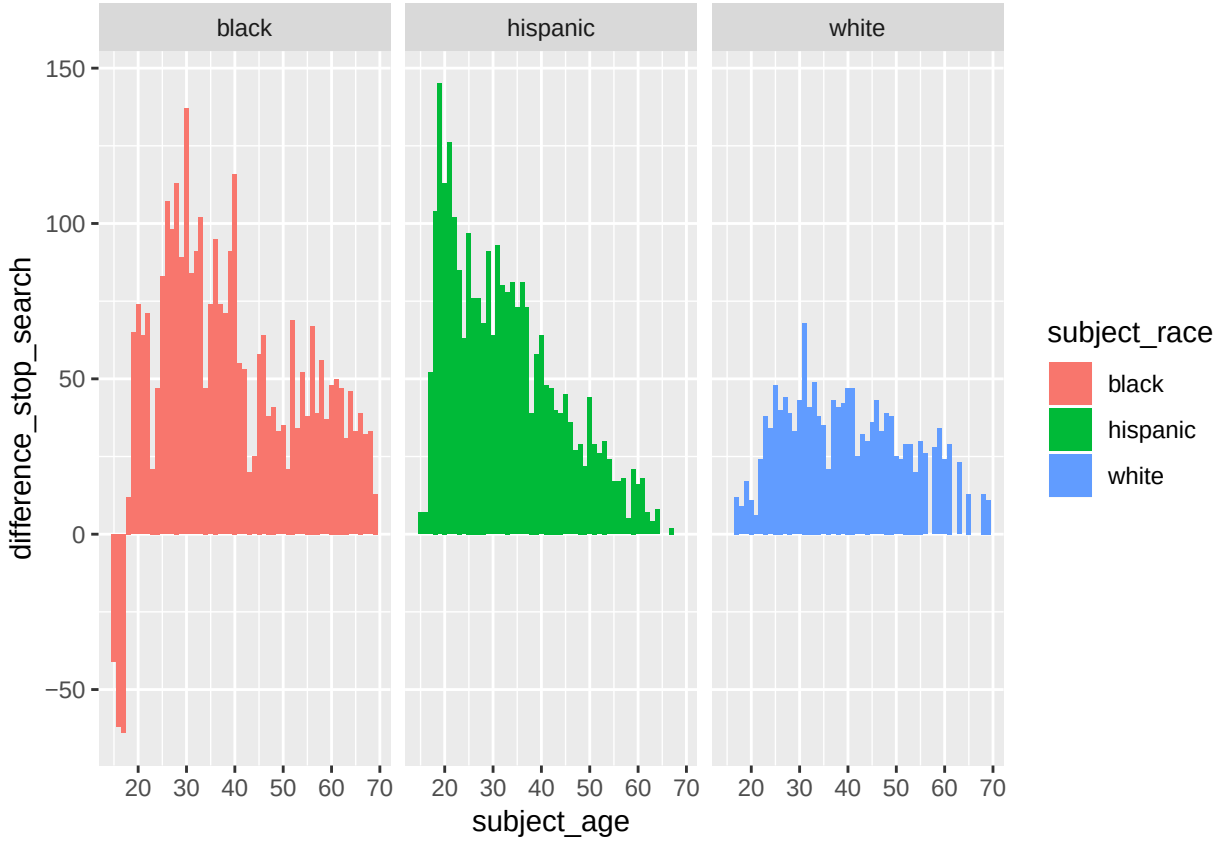
When difference_search_arrest is positive, then the number of high-discretionary searches (meaning, the number of searches that occured *without* resulting in an arrest, so police had high discretion in pulling those drivers over) is high. We see that the percent of high-discretionary stops over all discretionary stops is almost entirely positive for black drivers under 50 and hispanic drivers under 30.

repeat the above, but compare no search with search

```
# looking at stops that didn't result in an arrest
stop_searches <- CAoak %>%
  filter(subject_race != "asian/pacific islander", subject_race != "other", subject_age > 14, subject_ag
  group_by(subject_race, subject_age, search_conducted) %>%
  summarize(count = n())

stops_only <- stop_searches %>%
  ungroup() %>%
  filter(search_conducted == 0) %>%
  select(subject_race, subject_age, count) %>%
  rename(stop_count = count)

searches_only_2 <- stop_searches %>%
  filter(search_conducted == 1) %>%
```

```
  select(subject_race, subject_age, count)

# absolute numbers
stops_only %>%
  full_join(searches_only_2, by = c("subject_race", "subject_age")) %>%
  mutate(difference_stop_search = stop_count - count) %>%
  ggplot() +
  geom_bar(aes(x = subject_age, y = difference_stop_search, fill = subject_race), stat = "identity") +
  facet_wrap(~ subject_race)
```

```
## Warning: Removed 10 rows containing missing values (position_stack).
```




```
ggsave("stop and search differences absolute.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 10 rows containing missing values (position_stack).
```

```
#percentages
stops_only %>%
  full_join(searches_only_2, by = c("subject_race", "subject_age")) %>%
  mutate(difference_stop_search = (stop_count - count)/(count+stop_count)) %>%
  ggplot() +
  geom_bar(aes(x = subject_age, y = difference_stop_search, fill = subject_race), stat = "identity") +
  facet_wrap(~ subject_race)
```

```
## Warning: Removed 10 rows containing missing values (position_stack).
```


```
ggsave("stop and search differences percent.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 10 rows containing missing values (position_stack).
```

Reason for stop investigation

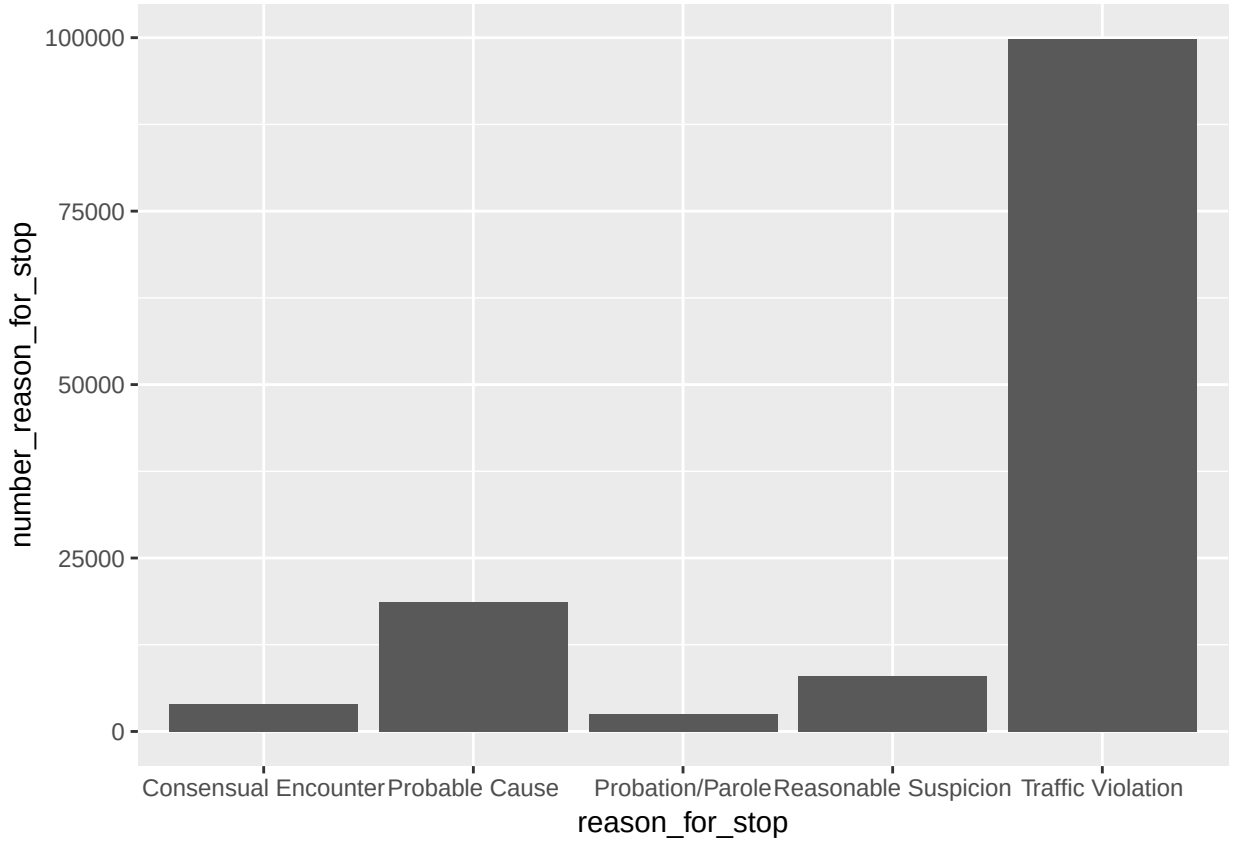
```
# Most frequent reason for stops
frequent_reasons_CAoak <- dbGetQuery(con,
  "SELECT reason_for_stop, COUNT(reason_for_stop) AS 'number_reason_for_stop'
  FROM CAoakland
  GROUP BY(reason_for_stop)
  ORDER BY `number_reason_for_stop` DESC
  LIMIT 5")

frequent_reasons_CAoak
```

```
##        reason_for_stop number_reason_for_stop
## 1    Traffic Violation                 99847
## 2       Probable Cause                 18643
## 3 Reasonable Suspicion                  8038
## 4 Consensual Encounter                  3998
## 5     Probation/Parole                  2451
```

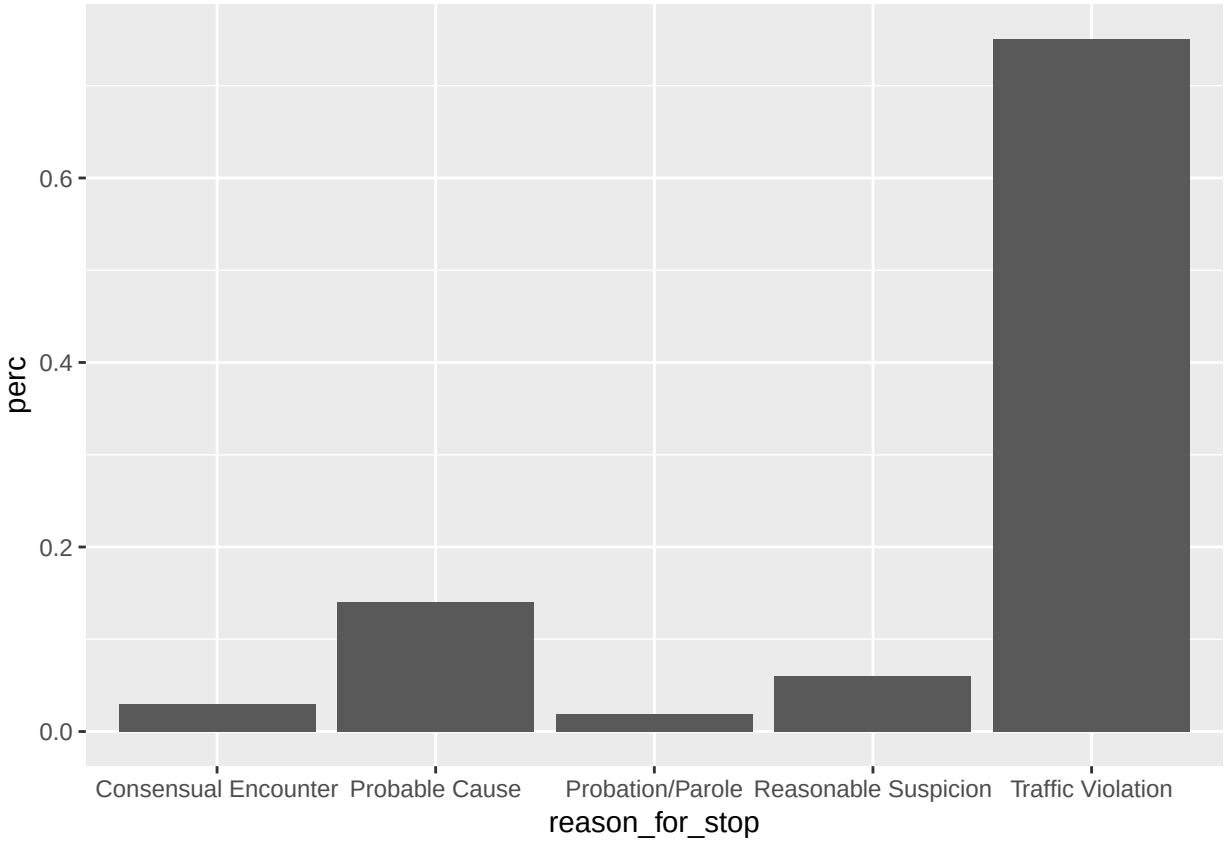
```
# Least frequent reason for stops (messy data)

dbGetQuery(con,
  "SELECT reason_for_stop, COUNT(reason_for_stop) AS 'number_reason_for_stop'
  FROM CAoakland
  GROUP BY(reason_for_stop)
  ORDER BY `number_reason_for_stop` ASC
  LIMIT 5")
```

```
##                                                                 reason_for_stop
## 1                                                                Other-Consensual,
## 2 Reasonable Suspicion|Reasonable Suspicion|Reasonable Suspicion|Probable Cause
## 3        Traffic Violation|Probation/Parole|Traffic Violation|Traffic Violation
## 4                Reasonable Suspicion|Reasonable Suspicion|Consensual Encounter
## 5                          Reasonable Suspicion|Probation/Parole|Probable Cause
##   number_reason_for_stop
## 1                      1
## 2                      1
## 3                      1
## 4                      1
## 5                      1
```

```
# Plot absolute and relative number of stops for frequent reasons

ggplot(data = frequent_reasons_CAoak) +
  geom_bar(mapping = aes(x = reason_for_stop, y = number_reason_for_stop), stat = "identity")
```



```
frequent_reasons_CAoak %>%
  mutate(perc = number_reason_for_stop/sum(frequent_reasons_CAoak$number_reason_for_stop)) %>%
  ggplot(aes(x = reason_for_stop, y = perc)) +
  geom_bar(stat = "identity")
```




```
# Relative number of cleanly coded reason for stop
sum(frequent_reasons_CAoak$number_reason_for_stop)/dim(CAoak)[1]
```

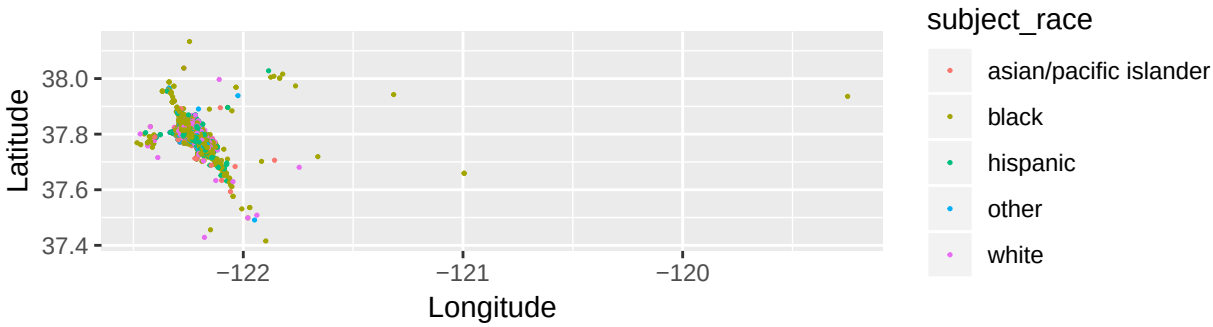
```
## [1] 0.9976592
```

Most of the 133,407 traffic stops in Oakland have a cleanly-coded reason for stop. The most frequently cited reason is traffic violation (70+%).

Maps

```
coor <- DBI::dbGetQuery(con, "SELECT lng, lat, subject_race FROM CAoakland")

# Note that lng and lat are of type double

ggplot(coor, aes(x = as.numeric(lng), y = as.numeric(lat), xaxt = 'n', yaxt = 'n')) +
  geom_point(aes(color = subject_race), position = "jitter", size = .25) +
  xlab("Longitude") +
  ylab("Latitude") +
  coord_quickmap()
```

```
## Warning: Removed 114 rows containing missing values (geom_point).
```




```
# Facet wrap race_plot
ggplot(coor, aes(x = as.numeric(lng), y = as.numeric(lat), xaxt = 'n', yaxt = 'n')) +
  geom_point(aes(color = subject_race), position = "jitter", size = .25) +
  xlab("Longitude") +
  ylab("Latitude") +
  coord_quickmap() +
  facet_wrap( ~ subject_race, nrow = 2)
```

```
## Warning: Removed 114 rows containing missing values (geom_point).
```