EDA

Exploratory Data Analysis

Useful links

- Data set

Imports

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(glue)
```

```
##
## Attaching package: 'glue'
```

```
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(UpSetR)
library(naniar)
devtools::load_all("chigcrim")
```

```
## Loading chigcrim
```

```
##
## Attaching package: 'testthat'
```

```
## The following object is masked from 'package:dplyr':
##
##     matches
```

```
## The following object is masked from 'package:purrr':
##
##     is_null
```

```
## The following object is masked from 'package:tidyr':
##
##     matches
```

Load data

```
df = read_csv("./data/crime-2019.csv", col_types = cols())
df
```

```
## # A tibble: 260,444 x 22
##        ID `Case Number` Date  Block IUCR  `Primary Type` Description
##     <dbl> <chr>         <chr> <chr> <chr> <chr>          <chr>
##  1 1.22e7 JD451055      01/0~ 020X~ 0890  THEFT          FROM BUILD~
##  2 1.22e7 JD450398      02/1~ 054X~ 1153  DECEPTIVE PRA~ FINANCIAL ~
##  3 1.22e7 JD450436      01/0~ 033X~ 0281  CRIMINAL SEXU~ NON-AGGRAV~
##  4 1.22e7 JD450960      01/0~ 056X~ 1753  OFFENSE INVOL~ SEXUAL ASS~
##  5 1.19e7 JC555894      12/2~ 004X~ 041A  BATTERY        AGGRAVATED~
##  6 1.19e7 JC540133      12/0~ 021X~ 1020  ARSON          BY FIRE
##  7 1.22e7 JD449684      12/1~ 020X~ 1585  SEX OFFENSE    OTHER
##  8 1.22e7 JD449113      06/1~ 019X~ 1153  DECEPTIVE PRA~ FINANCIAL ~
##  9 2.46e4 JC337772      07/0~ 002X~ 0110  HOMICIDE       FIRST DEGR~
## 10 1.20e7 JC543338      12/1~ 067X~ 2024  NARCOTICS      POSSESS - ~
## # ... with 260,434 more rows, and 15 more variables: `Location
## #   Description` <chr>, Arrest <lgl>, Domestic <lgl>, Beat <chr>,
## #   District <chr>, Ward <dbl>, `Community Area` <dbl>, `FBI Code` <chr>, `X
## #   Coordinate` <dbl>, `Y Coordinate` <dbl>, Year <dbl>, `Updated On` <chr>,
## #   Latitude <dbl>, Longitude <dbl>, Location <chr>
```

```
colnames(df) <- str_replace(tolower(colnames(df)), " ", "_")

# sort out dates
df = df %>%
  mutate(timestamp =  ymd_hms(mdy_hms(date)),
         updated_on = ymd_hms(mdy_hms(updated_on))) %>%
  select(-date)
```

Summary:

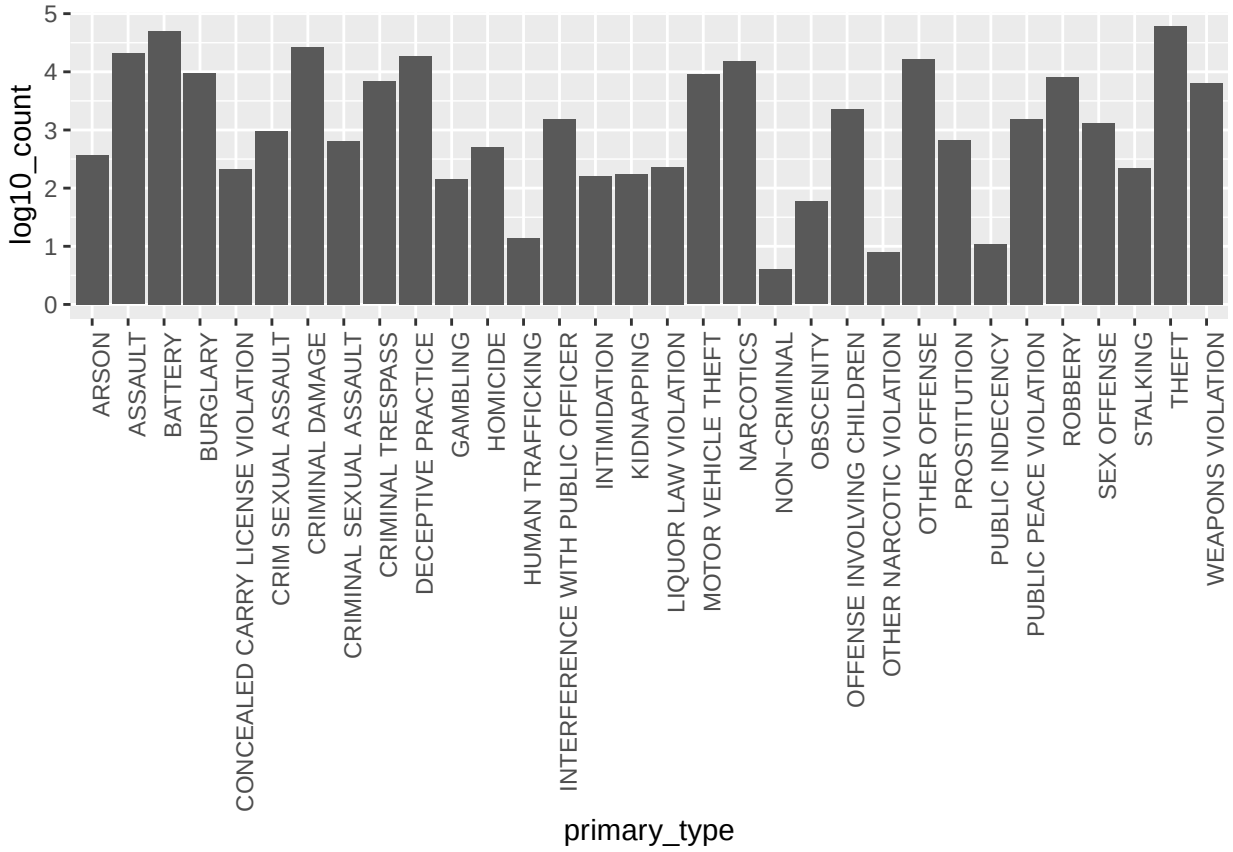
```
summary(df)
```

```
##        id           case_number           block               iucr
##  Min.   :   24368   Length:260444      Length:260444      Length:260444
##  1st Qu.:11654233   Class :character   Class :character   Class :character
##  Median :11751350   Mode  :character   Mode  :character   Mode  :character
##  Mean   :11728939
##  3rd Qu.:11846842
##  Max.   :12238231
##
##  primary_type       description        location_description   arrest
##  Length:260444      Length:260444      Length:260444        Mode :logical
##  Class :character   Class :character   Class :character     FALSE:204462
##  Mode  :character   Mode  :character   Mode  :character     TRUE :55982
```

```
##
##
##
##
##   domestic           beat             district             ward
##  Mode :logical   Length:260444      Length:260444      Min.   : 1.00
##  FALSE:217293    Class :character   Class :character   1st Qu.:10.00
##  TRUE :43151     Mode  :character   Mode  :character   Median :24.00
##                                                        Mean   :23.33
##                                                        3rd Qu.:34.00
##                                                        Max.   :50.00
##                                                        NA's   :15
##  community_area    fbi_code          x_coordinate      y_coordinate
##  Min.   : 1.00   Length:260444      Min.   :      0   Min.   :      0
##  1st Qu.:23.00   Class :character   1st Qu.:1153429   1st Qu.:1859114
##  Median :32.00   Mode  :character   Median :1167012   Median :1893829
##  Mean   :36.62                      Mean   :1165110   Mean   :1886304
##  3rd Qu.:53.00                      3rd Qu.:1176561   3rd Qu.:1908244
##  Max.   :77.00                      Max.   :1205116   Max.   :1951520
##                                     NA's   :1290      NA's   :1290
##       year        updated_on                    latitude       longitude
##  Min.   :2019   Min.   :2019-01-10 15:16:50   Min.   :36.62   Min.   :-91.69
##  1st Qu.:2019   1st Qu.:2019-04-23 16:22:49   1st Qu.:41.77   1st Qu.:-87.71
##  Median :2019   Median :2019-07-19 16:22:44   Median :41.86   Median :-87.66
##  Mean   :2019   Mean   :2019-07-23 10:23:20   Mean   :41.84   Mean   :-87.67
##  3rd Qu.:2019   3rd Qu.:2019-10-12 16:05:42   3rd Qu.:41.90   3rd Qu.:-87.63
##  Max.   :2019   Max.   :2020-12-06 15:49:24   Max.   :42.02   Max.   :-87.52
##                                               NA's   :1290    NA's   :1290
##    location           timestamp
##  Length:260444      Min.   :2019-01-01 00:00:00
##  Class :character   1st Qu.:2019-04-10 14:29:30
##  Mode  :character   Median :2019-07-05 10:15:00
##                     Mean   :2019-07-04 03:30:23
##                     3rd Qu.:2019-09-26 23:15:00
##                     Max.   :2019-12-31 23:55:00
##
```

Overall:

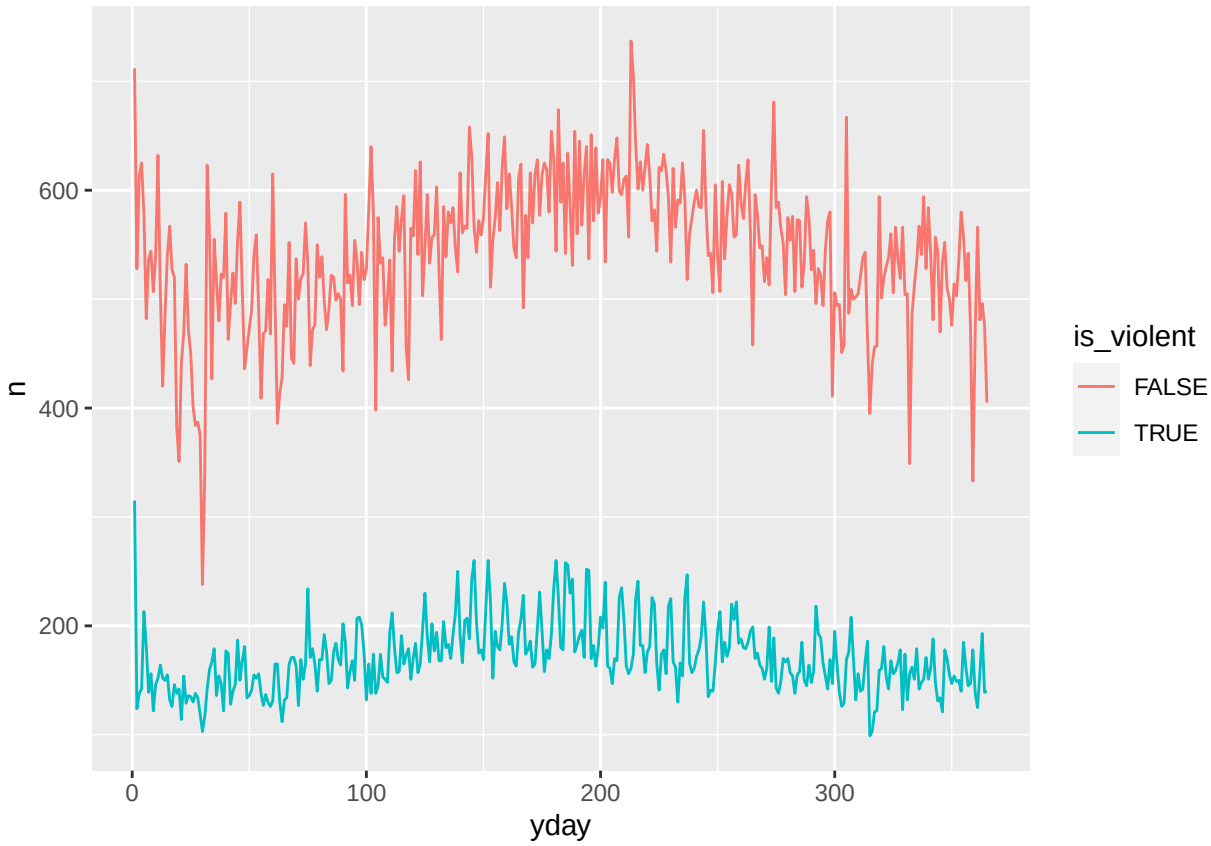
```
df %>%
  group_by(primary_type) %>%
  tally(name = "count") %>%
  mutate(log10_count = log10(count)) %>%
  ggplot() +
  geom_col(aes(x=primary_type, y=log10_count)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Date and time

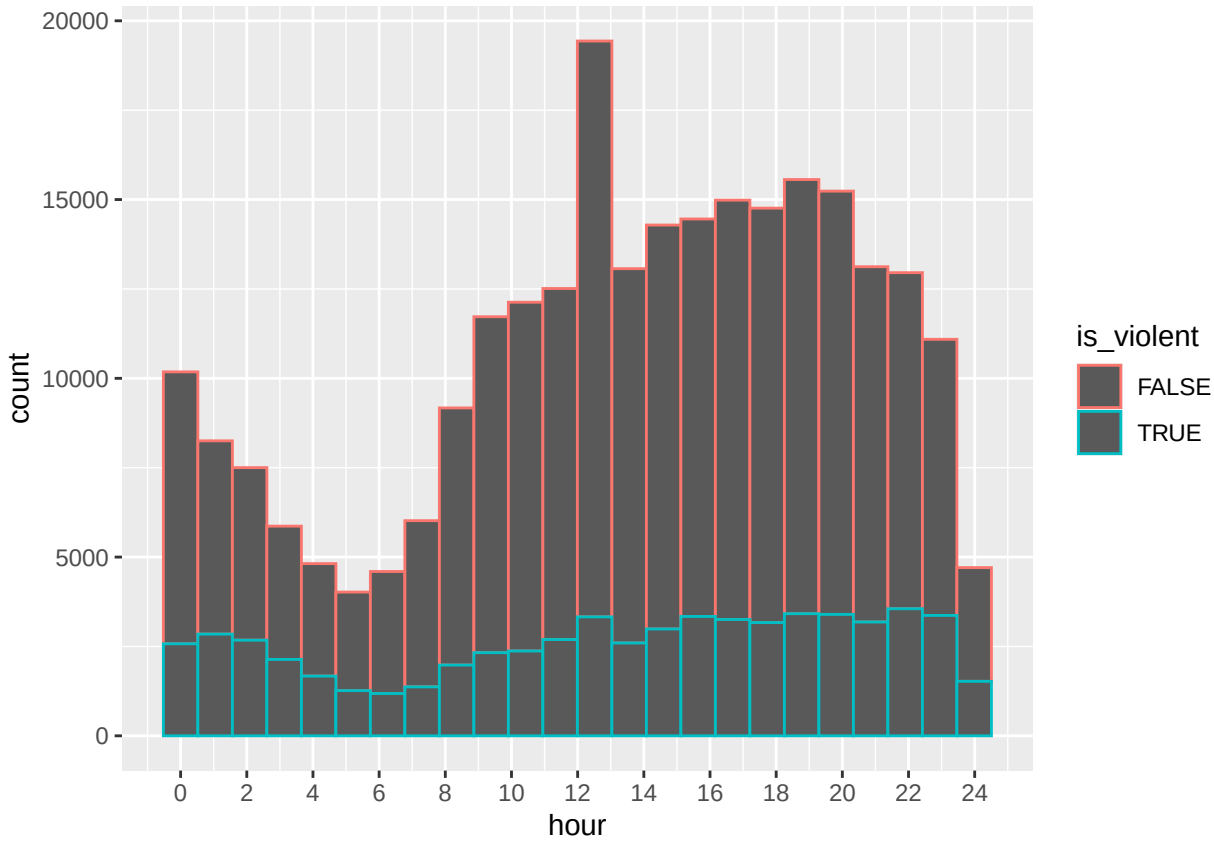
Over the year:

```
# This might not be perfect but looks pretty good.
violent_queries <- c("SEX", "ABUSE", "HOMICIDE", "VIOLENT",
                     "BATTERY", "AGG ", "AGGRAVATED")

df$description <- paste(df$description, df$primary_type)


df$is_violent <- str_detect(df$description,
                      pattern = paste(violent_queries, collapse = "|"))

df %>%
  mutate(yday = yday(timestamp)) %>%
  group_by(yday, is_violent) %>%
  tally() %>%
  ggplot(aes(x=yday, y=n, col=is_violent)) +
    geom_line()
```



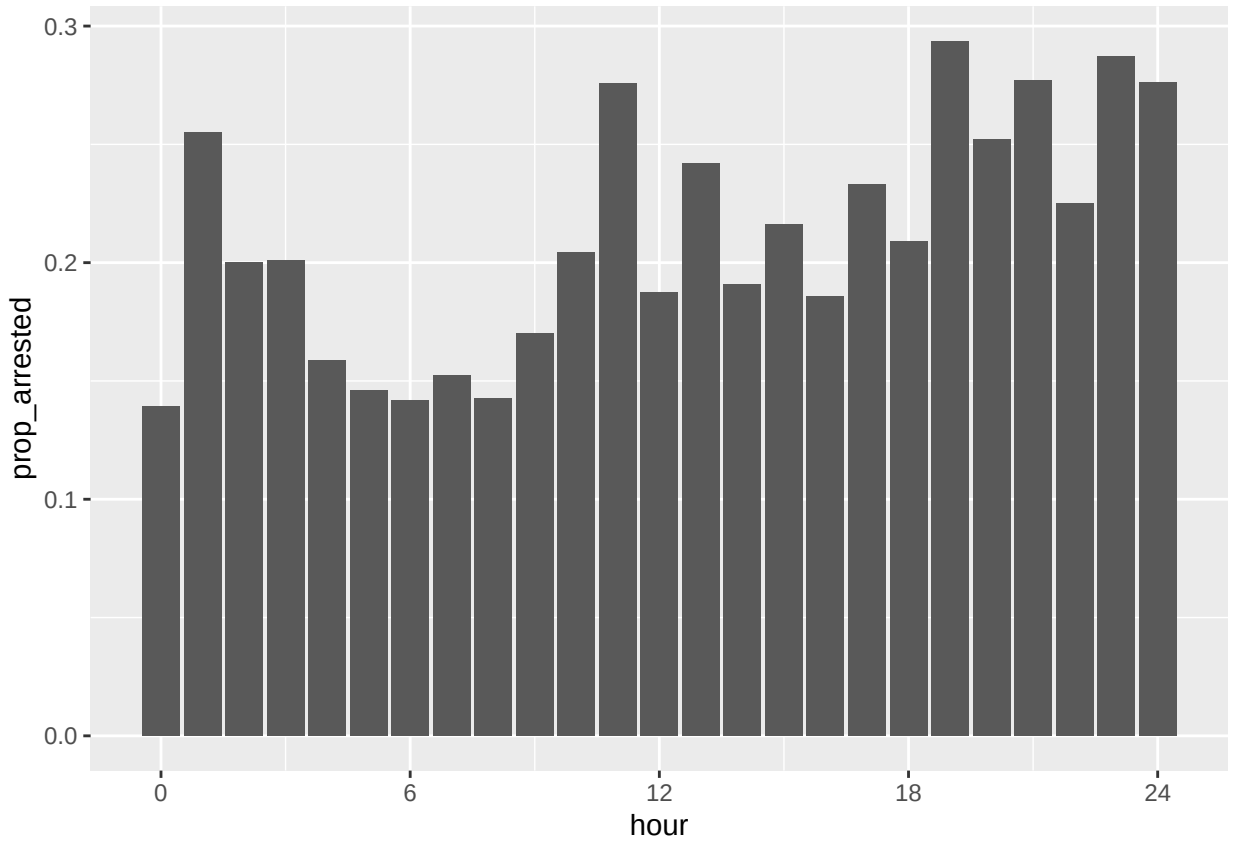
Over 24 hour period:

```
df$hour = hour(df$timestamp) + minute(df$timestamp)/60

df %>%
  ggplot(aes(x=hour, col=is_violent)) +
    geom_histogram(bins=24) +
    scale_x_continuous(breaks = seq(0,24, 2))
```



Proportion of incidents leading to arrests by time of day:

```
df %>%
  mutate(hour = round(hour)) %>%
  group_by(hour, arrest) %>%
  tally() %>%
  pivot_wider(names_from = arrest, values_from = n) %>%
  mutate(prop_arrested = `TRUE`/(`TRUE`+`FALSE`)) %>%
  ggplot(aes(x=hour, y=prop_arrested)) +
  geom_col() +
  scale_x_continuous(breaks = seq(0,24,6))
```



Are there any duplicate case numbers?

```
sum(duplicated(df$case_number))
```

```
## [1] 21
```

Duplicated IDs?

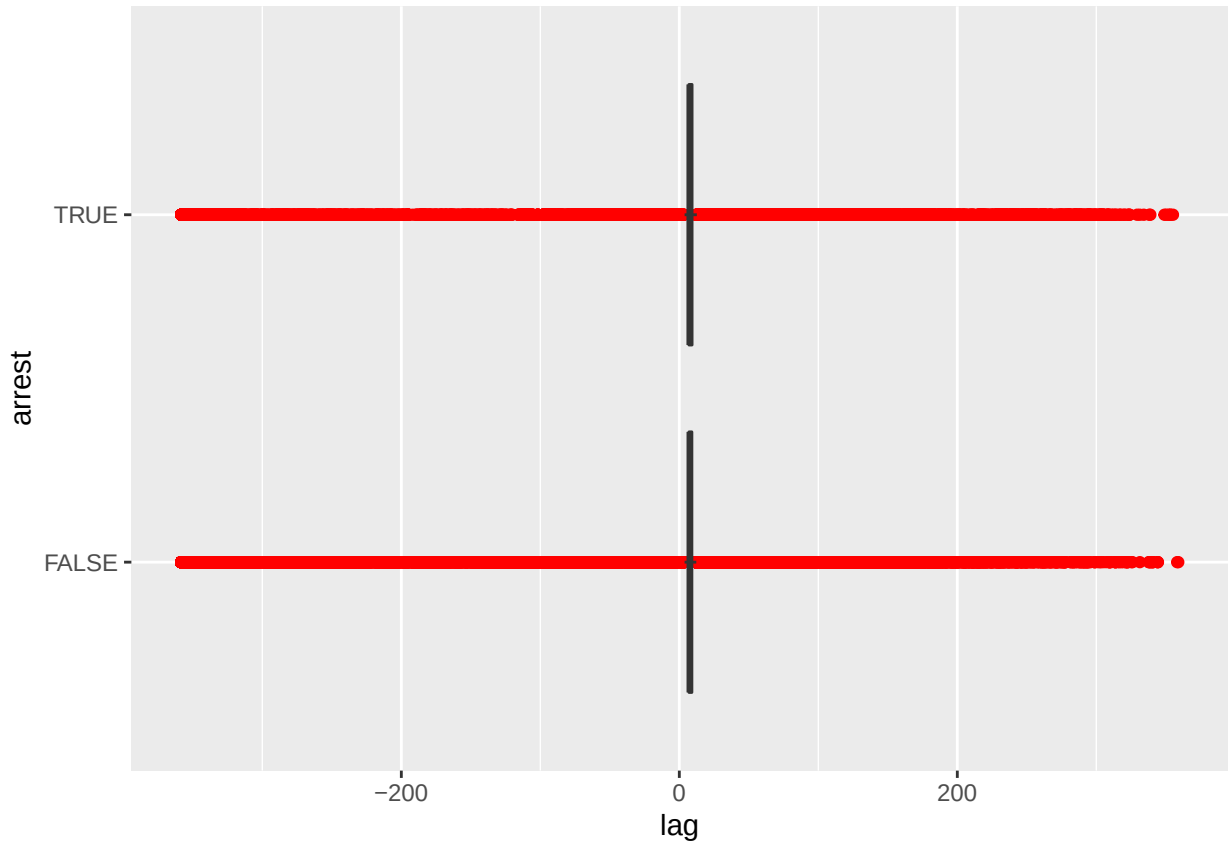
```
sum(duplicated(df$id))
```

```
## [1] 0
```

Updated on

Perhaps the updated_on feature is useful for predicting arrests. In particular, perhaps the lag between the crime and when it was updated is informative:

```
lag = yday_float(df$updated_on)  - yday_float(df$timestamp)
lag_df = tibble(lag, arrest=df$arrest)

ggplot(lag_df) +
  geom_boxplot(aes(x=lag, y=arrest), outlier.colour = "red", )
```

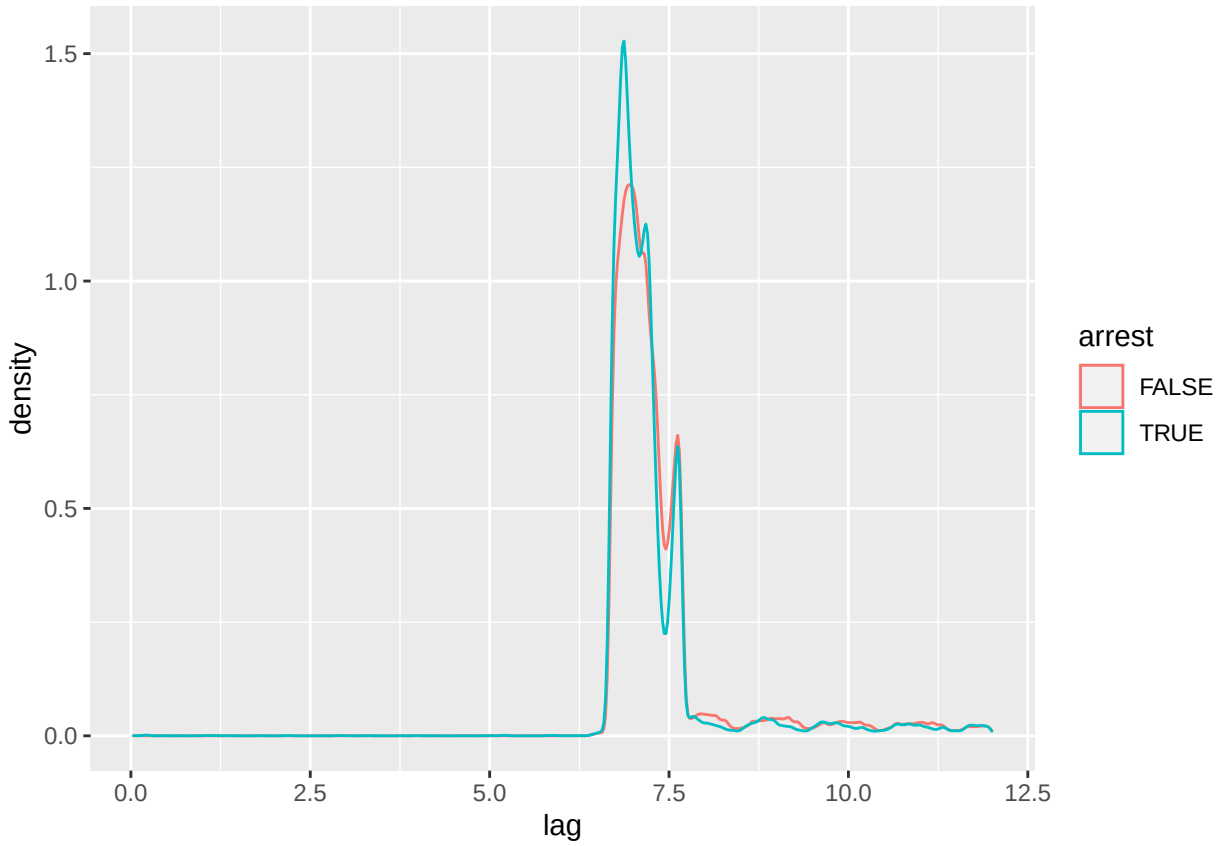


Some crimes seem to have been last updated before the crime took place? Lots of big outliers. Not sure I understand this... But most seem to be reasonable:

```
lag_df %>%
  group_by(arrest) %>%
  summarise(mean = mean(lag), .groups="drop",
            median = median(lag),
            iqr = IQR(lag))
```

```
## # A tibble: 2 x 4
##   arrest  mean median   iqr
##   <lgl>  <dbl>  <dbl> <dbl>
## 1 FALSE   2.50   7.23  2.11
## 2 TRUE    5.16   7.17  2.29
```

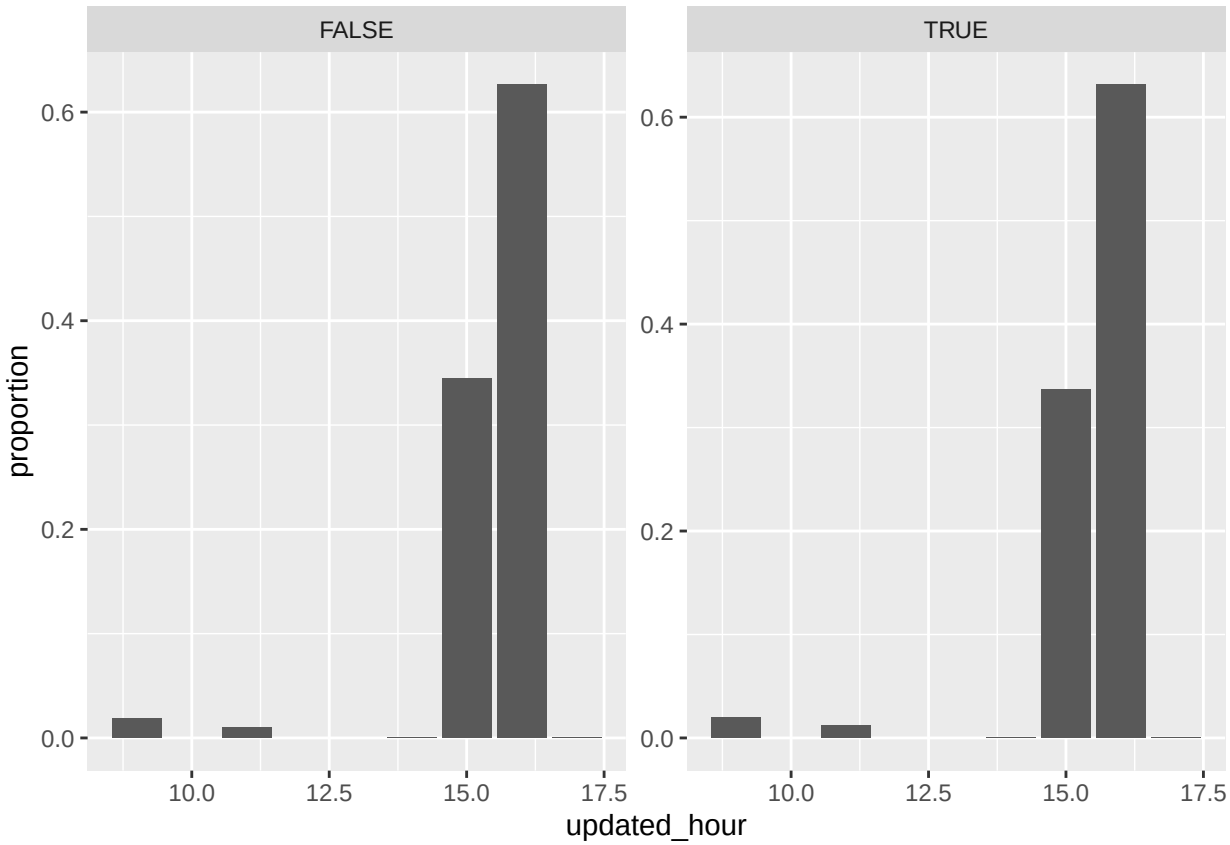
```
lag_df %>%
  filter(lag>0, lag < 12) %>%
  ggplot(aes(x=lag, col=arrest)) +
    geom_density()
```



Maybe some slight difference between densities, but doesn't seem like much. Most are uploaded a week later, mean is skewed by outliers.

What time are they updated:

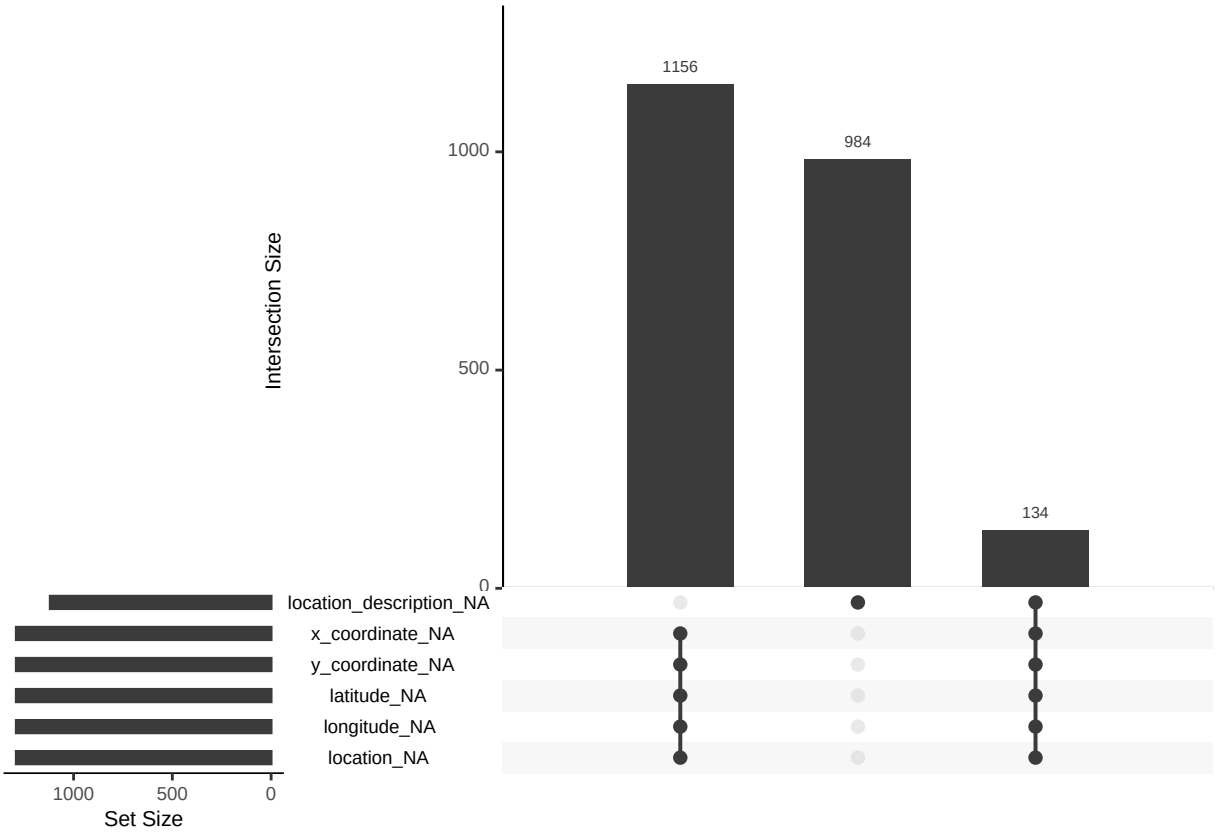
```
df %>%
  mutate(updated_hour = hour(df$updated_on)) %>%
  group_by(updated_hour, arrest) %>%
  tally() %>%
  group_by(arrest) %>%
  mutate(proportion = n/sum(n)) %>%
  ggplot() +
  geom_col(aes(x=updated_hour, y=proportion)) +
  facet_wrap(~arrest, scales = "free")
```



Looks like they both tend to be updated at similar times for arrested/not arrested.

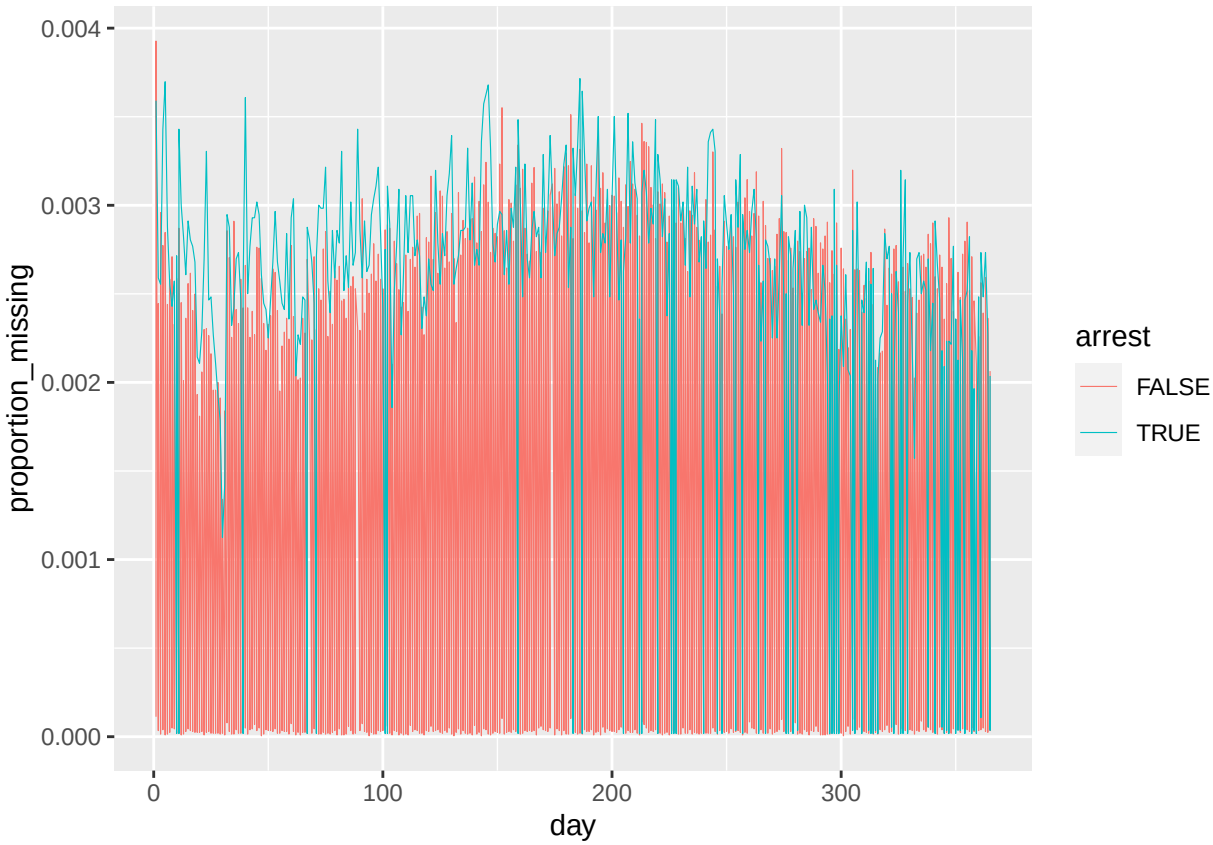
Missing values

```
p = gg_miss_upset(df, nsets=6)
p
```



This plot tells us that only the 6 variables have missing values, and they fall into 3 subsets. Note that other variables `district` and `block`, are not missing to these could be used to impute the missing values. Given they are such a small proportion of the dataset, we could also just drop them. To have a quick look to check they are missing completely at random:

```
# Plot index of missing values
df %>%
  mutate(complete = complete.cases(df)) %>%
  mutate(day = yday(timestamp)) %>%
  group_by(day, complete, arrest) %>%
  tally() %>%
  group_by(arrest) %>%
  mutate(proportion_missing = n/sum(n)) %>%
  ggplot() +
  geom_line(aes(y=proportion_missing, x=day, col=arrest), size=0.2)
```



Does not seem to be missing completely at random, but perhaps a small enough part of the data set that we can ignore it anyway.

Sp

```
library(sf)
```

```
## Linking to GEOS 3.8.1, GDAL 3.0.4, PROJ 6.3.1
## WARNING: different compile-time and runtime versions for GEOS found:
## Linked against: 3.8.1-CAPI-1.13.3 compiled against: 3.8.0-CAPI-1.13.1
## It is probably a good idea to reinstall sf, and maybe rgeos and rgdal too
```

```
library(raster)
```

```
## Loading required package: sp
##
## Attaching package: 'raster'
## The following objects are masked from 'package:chigcrim':
##
##     extract, select
## The following object is masked from 'package:glue':
##
##     trim
## The following object is masked from 'package:dplyr':
##
```

```
##     select
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
# Import neighbourhood boundaries
bounds <- st_read("data/nbd_bounds.shp")
```

```
## Reading layer `nbd_bounds' from data source `/home/dw16200/Documents/compass/group_project/chicago-c
## Simple feature collection with 98 features and 4 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -87.94011 ymin: 41.64454 xmax: -87.52414 ymax: 42.02304
## geographic CRS: WGS84(DD)
```