

Cincinnati fire

Saikat Banerjee

April 12, 2018

```
# Loading the required packages for data analysis  
library(lubridate)
```

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

```
## The following object is masked from 'package:base':  
##  
##     date
```

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.  
2.1 --
```

```
## v ggplot2 2.2.1      v purrr   0.2.4  
## v tibble  1.4.1      v dplyr    0.7.4  
## v tidyr   0.7.2      v stringr  1.2.0  
## v readr   1.1.1      vforcats  0.2.0
```

```
## -- Conflicts ----- tidyverse_conflicts  
() --  
## x lubridate::as.difftime() masks base::as.difftime()  
## x lubridate::date()       masks base::date()  
## x dplyr::filter()        masks stats::filter()  
## x lubridate::intersect() masks base::intersect()  
## x dplyr::lag()          masks stats::lag()  
## x lubridate::setdiff()   masks base::setdiff()  
## x lubridate::union()    masks base::union()
```

```
library(ggmap)
```

```
# Reading in the data  
cin_fire <- read_csv("~/Cincinnati fire/Cfire.csv")
```

```
## Parsed with column specification:
## cols(
##   ADDRESS_X = col_character(),
##   LATITUDE_X = col_double(),
##   LONGITUDE_X = col_double(),
##   AGENCY = col_character(),
##   CREATE_TIME INCIDENT = col_character(),
##   DISPOSITION_TEXT = col_character(),
##   EVENT_NUMBER = col_character(),
##   INCIDENT_TYPE_ID = col_character(),
##   INCIDENT_TYPE_DESC = col_character(),
##   NEIGHBORHOOD = col_character(),
##   ARRIVAL_TIME_PRIMARY_UNIT = col_character(),
##   BEAT = col_character(),
##   CLOSED_TIME INCIDENT = col_character(),
##   DISPATCH_TIME_PRIMARY_UNIT = col_character(),
##   CFD INCIDENT_TYPE = col_character(),
##   CFD INCIDENT_TYPE_GROUP = col_character(),
##   COMMUNITY_COUNCIL_NEIGHBORHOOD = col_character()
## )
```

#Let's see what the framework of the dataset looks like

```
glimpse(cin_fire)
```

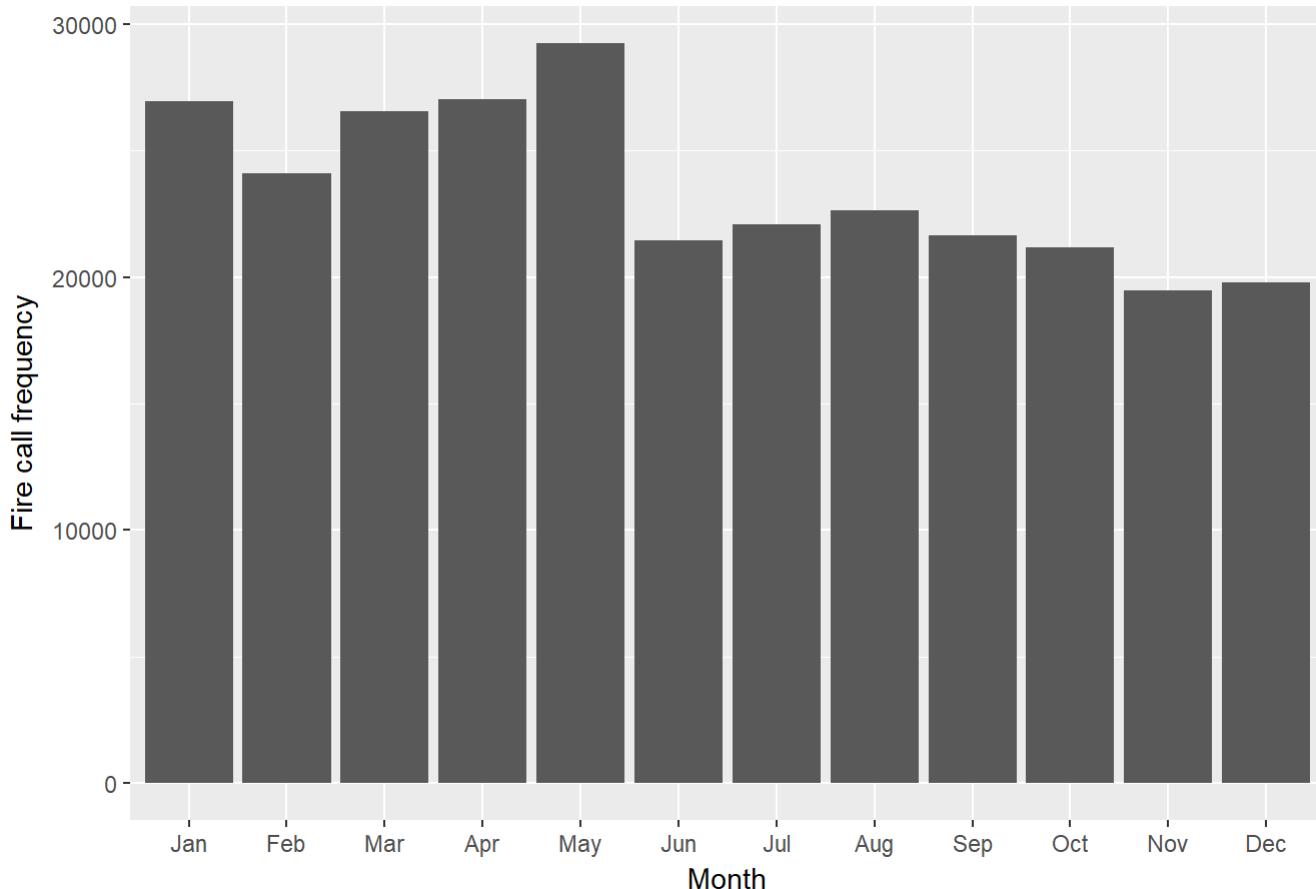
```
## Observations: 282,092
## Variables: 17
## $ ADDRESS_X <chr> "S I471 AT LIBERTY", "BROADWAY"...
## $ LATITUDE_X <dbl> NA, 39.10869, 39.12836, 39.1064...
## $ LONGITUDE_X <dbl> NA, -84.50847, -84.48643, -84.5...
## $ AGENCY <chr> "CF", "CFD", "CFD", "CFD", "CFD...
## $ CREATE_TIME INCIDENT <chr> "03/07/2015 08:10:04 PM", "03/0...
## $ DISPOSITION_TEXT <chr> "DUPLICATE INCIDENT", "AV: ADVI...
## $ EVENT_NUMBER <chr> "FCF150307000176", "CFD18030900...
## $ INCIDENT_TYPE_ID <chr> "ACCIF", "FALARM", "FALARM", "3...
## $ INCIDENT_TYPE_DESC <chr> "AUTO ACCIDENT INJURI", NA, NA,...
## $ NEIGHBORHOOD <chr> "N/A", "PENDLETON", "WALNUT HIL...
## $ ARRIVAL_TIME_PRIMARY_UNIT <chr> "03/07/2015 08:15:15 PM", "03/0...
## $ BEAT <chr> "1296", "ST03", "ST23", "ST03",...
## $ CLOSED_TIME INCIDENT <chr> "03/07/2015 08:20:50 PM", "03/0...
## $ DISPATCH_TIME_PRIMARY_UNIT <chr> "03/07/2015 08:11:08 PM", "03/0...
## $ CFD INCIDENT_TYPE <chr> "BLS", "FIRE", "FIRE", "ALS", "...
## $ CFD INCIDENT_TYPE_GROUP <chr> "ACCIDENT WITH INJURY - FIRE ON...
## $ COMMUNITY_COUNCIL_NEIGHBORHOOD <chr> "N/A", "PENDLETON", "WALNUT HIL..."
```

```
#renaming cols of the cin_fire
names(cin_fire) <- c("Address", "Latitude", "Longitude", "Agency",
                      "Create.datetime", "Disposition.text", "Event.no",
                      "Inc.ID", "Inc.des", "Neighborhood", "PU.arrival.datetime",
                      "Beat", "Close.datetime", "PU.dispatch.datetime",
                      "CFD.Inc.type", "CFD.inc.type.group", "Council.neighborhood")
# Parsing datetime using lubridate
cin_fire$Create.datetime <- parse_date_time(cin_fire$Create.datetime,
                                              "%m/%d/%y %I:%M:%S %p")
# Dividing datetime into useful features
cin_fire$Month <- month(cin_fire$Create.datetime, label = T)
cin_fire$DayoftheWeek <- wday(cin_fire$Create.datetime, label = T)
cin_fire$Hour <- hour(cin_fire$Create.datetime)
```

```
# Making dataviz for trends in the fire calls
cin_fire %>% group_by(Month) %>% summarise(n.calls = n()) %>%
  ggplot(mapping = aes(x = Month, y = n.calls)) +
  geom_histogram(stat = "identity") + labs(y = "Fire call frequency",
    x = "Month", title = "Monthly distribution of fire calls")
```

Warning: Ignoring unknown parameters: binwidth, bins, pad

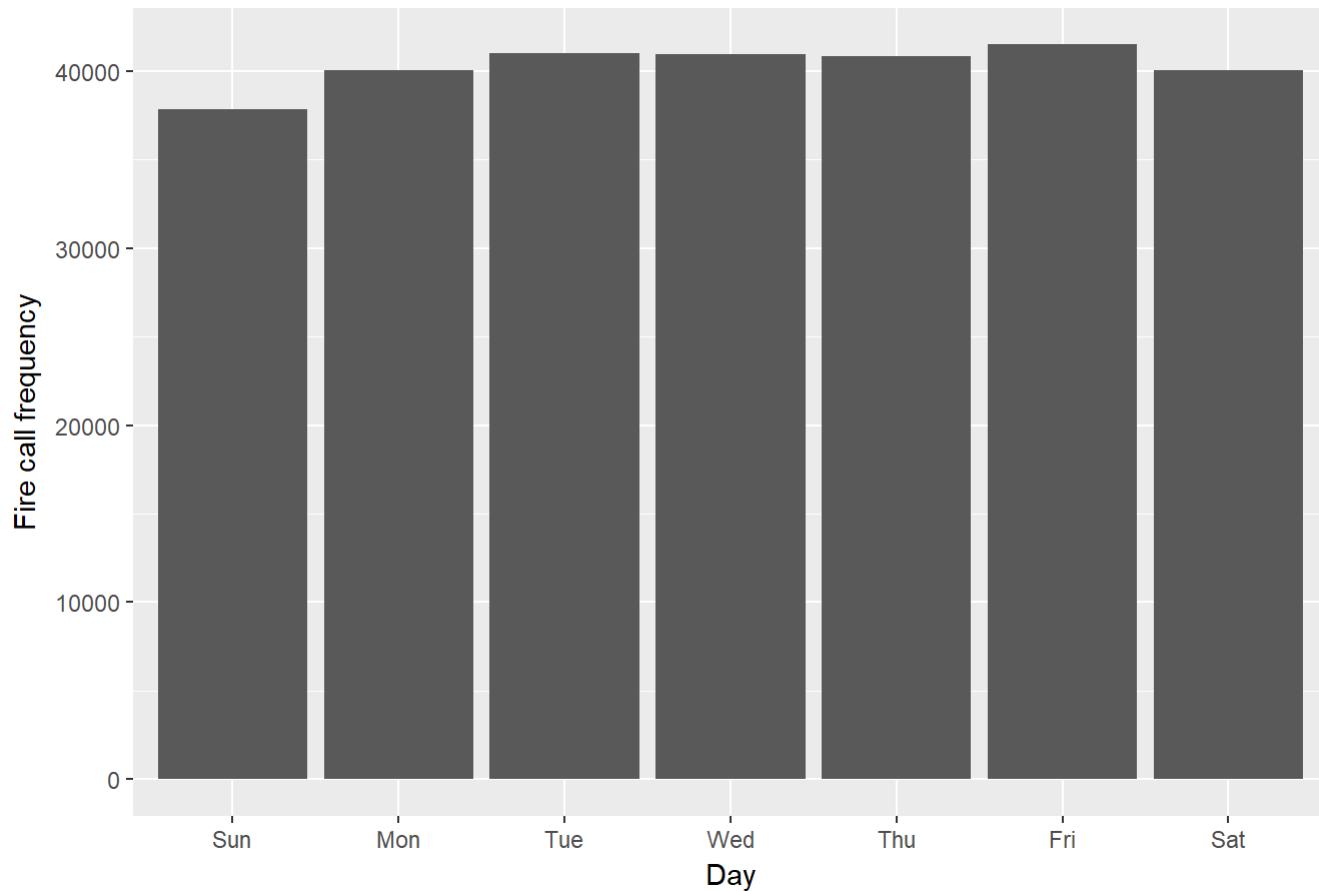
Monthly distribution of fire calls



```
cin_fire %>% group_by(DayoftheWeek) %>% summarise(n.calls = n()) %>%
  ggplot(mapping = aes(x = DayoftheWeek, y = n.calls)) +
  geom_histogram(stat = "identity") + labs(y = "Fire call frequency",
  x = "Day", title = "Daily distribution of fire calls")
```

Warning: Ignoring unknown parameters: binwidth, bins, pad

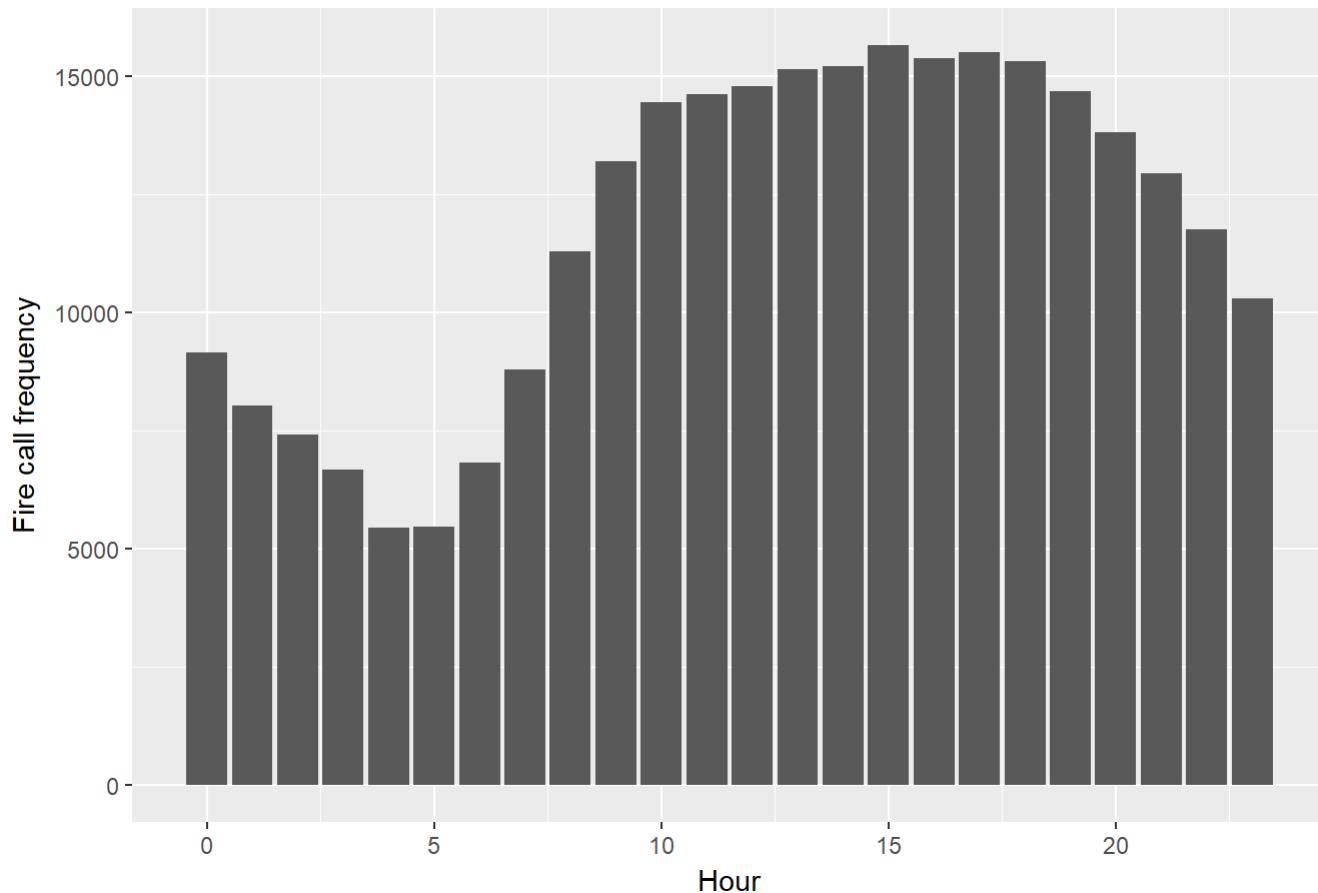
Daily distribution of fire calls



```
cin_fire %>% group_by(Hour) %>% summarise(n.calls = n()) %>%
  ggplot(mapping = aes(x = Hour, y = n.calls)) +
  geom_histogram(stat = "identity") + labs(y = "Fire call frequency",
  x = "Hour", title = "Hourly distribution of fire calls")
```

Warning: Ignoring unknown parameters: binwidth, bins, pad

Hourly distribution of fire calls



```
# Hourly trends are evident with Late afternoon being the busiest of times and early
# morning the lightest time; no significant patterns in daily data; monthly data show
# that there are greater number of calls between Jan-May compared with rest of the year.
```

```
# Selecting the coordinates and other variables for spatial visualizations
cin_fire_maps <- cin_fire %>% select(Longitude, Latitude)
```

```
cin_fire_maps <- cin_fire_maps %>% filter(!is.na(Longitude) | !is.na(Latitude))
```

```
cin_code <- as.numeric/geocode("Cincinnati")
```

```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Cincinnati&sensor=false
```

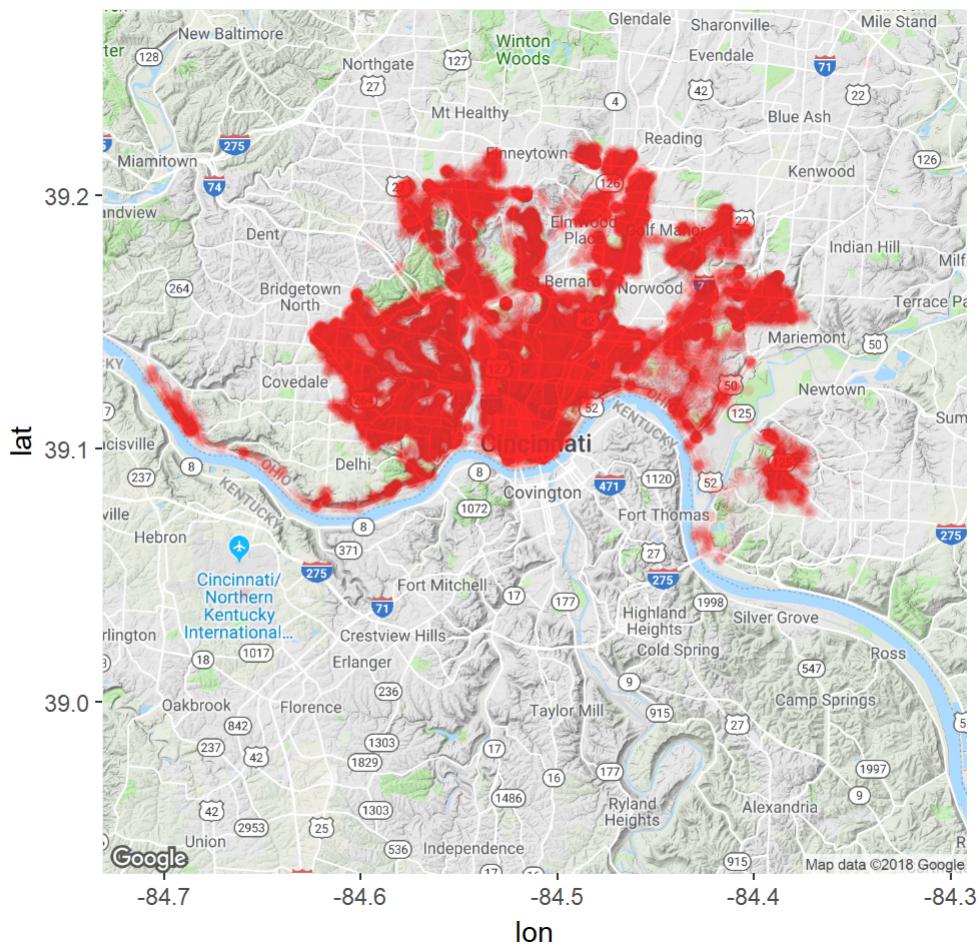
```
## The corresponding googlemap is imported and used here.
```

```
cin_map <- ggmap(get_googlemap(center = cin_code, scale = 2, zoom = 11,
                                 extent = "device"))
```

```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=39.103118,-84.51202&zoom=11&size=640x640&scale=2&maptype=terrain&sensor=false
```

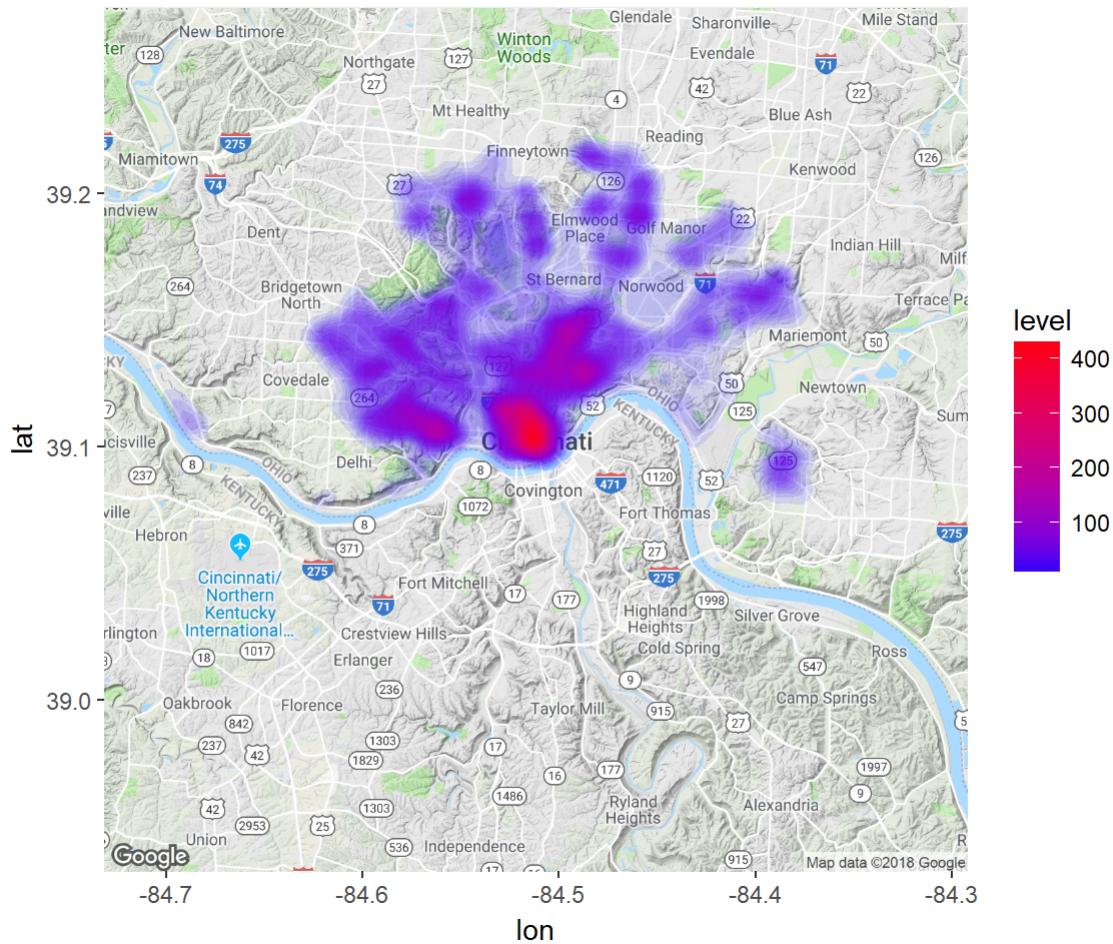
```
cin_map + geom_jitter(aes(x = Longitude, y = Latitude), color = "red",
alpha = 0.01, data = cin_fire_maps)
```

```
## Warning: Removed 16 rows containing missing values (geom_point).
```



```
##The heat density maps are used to qualify the areas with most fire calls.
cin_map + stat_density2d(data = cin_fire_maps,
aes(x = Longitude, y = Latitude, fill = ..level.., alpha = ..level..),
bins = 64, geom = "polygon") + scale_fill_gradient(low = "blue", high = "red")
+
scale_alpha(guide = 'none')
```

```
## Warning: Removed 16 rows containing non-finite values (stat_density2d).
```



```
## Parsing the date and time for dispatch and closing of the event
cin_fire$PU.arrival.datetime <- parse_date_time(cin_fire$PU.arrival.datetime,
                                              "%m/%d/%y %I:%M:%S %p")
cin_fire$PU.dispatch.datetime <- parse_date_time(cin_fire$PU.dispatch.datetime,
                                              "%m/%d/%y %I:%M:%S %p")
cin_fire$Close.datetime <- parse_date_time(cin_fire$Close.datetime,
                                             "%m/%d/%y %I:%M:%S %p")
# Creating a new variable PU.time, which is time taken from dispatch upto arrival
# of help to gauge the efficiency of the department
cin_fire$PU.time <- cin_fire$PU.arrival.datetime - cin_fire$PU.dispatch.datetime

cin_fire$PU.time %>% as.integer() %>% summary()
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	-86918306	198	268	-61893	354	44192	45616

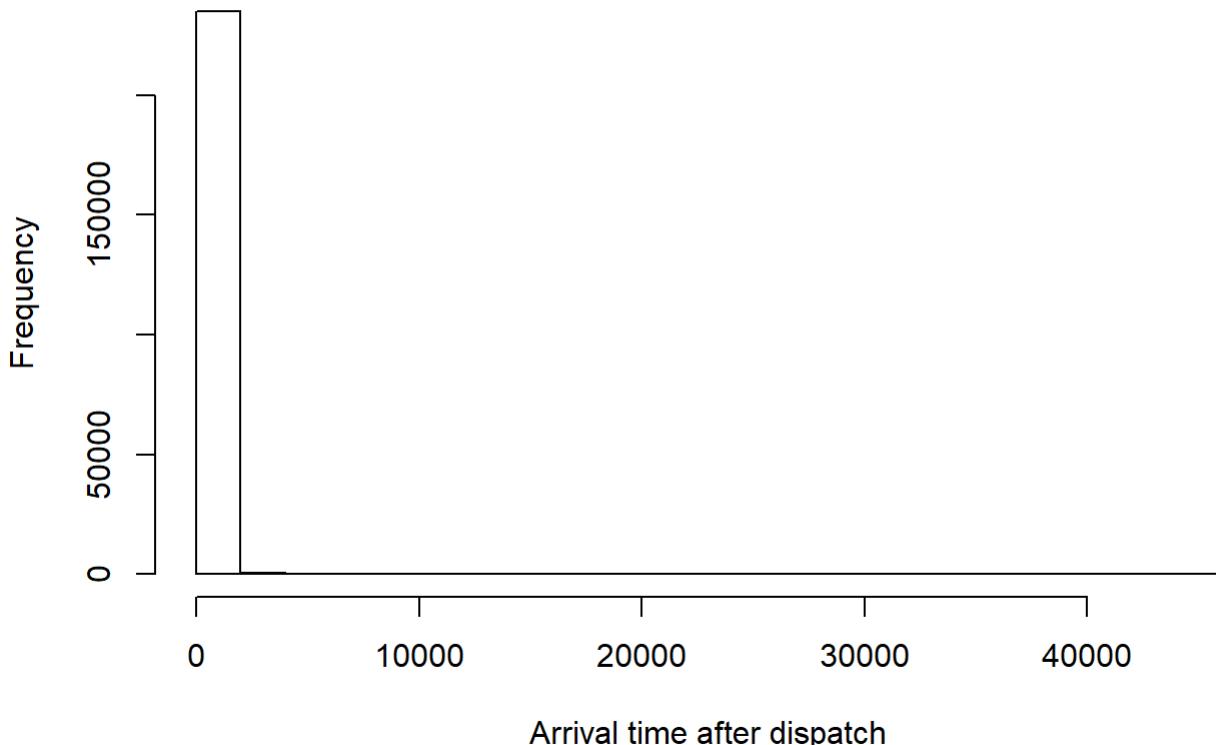
```
## Interesting! PU.time has Large negative values, which is unrealistic
## because dispatch time cannot be larger than arrival time. Upon careful
## review of the rows with very large negative PU.time, it was found that
## some of the dates have been wrongly entered. This has to be taken care of.
```

```
# Number of possible wrong entry in the dataset
num_wrong_entry <- sum(cin_fire$PU.dispatch.datetime > cin_fire$PU.arrival.datetime,
                      na.rm = T)
id_wrong_entry <- which(cin_fire$PU.dispatch.datetime > cin_fire$PU.arrival.datetime)
# The number is Less than 4% of the data. Therefore, the misentered data was decided
# to be removed.
cin_fire_new <- cin_fire [-(id_wrong_entry),]
# Similar steps carried out with the new dataset to get a glimpse
cin_fire_new$PU.time <- cin_fire_new$PU.arrival.datetime - cin_fire_new$PU.dispatch.datetime
# What does the new summary Look Like?
cin_fire_new$PU.time %>% as.integer() %>% summary()
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	0	199	268	333	355	44192	45616

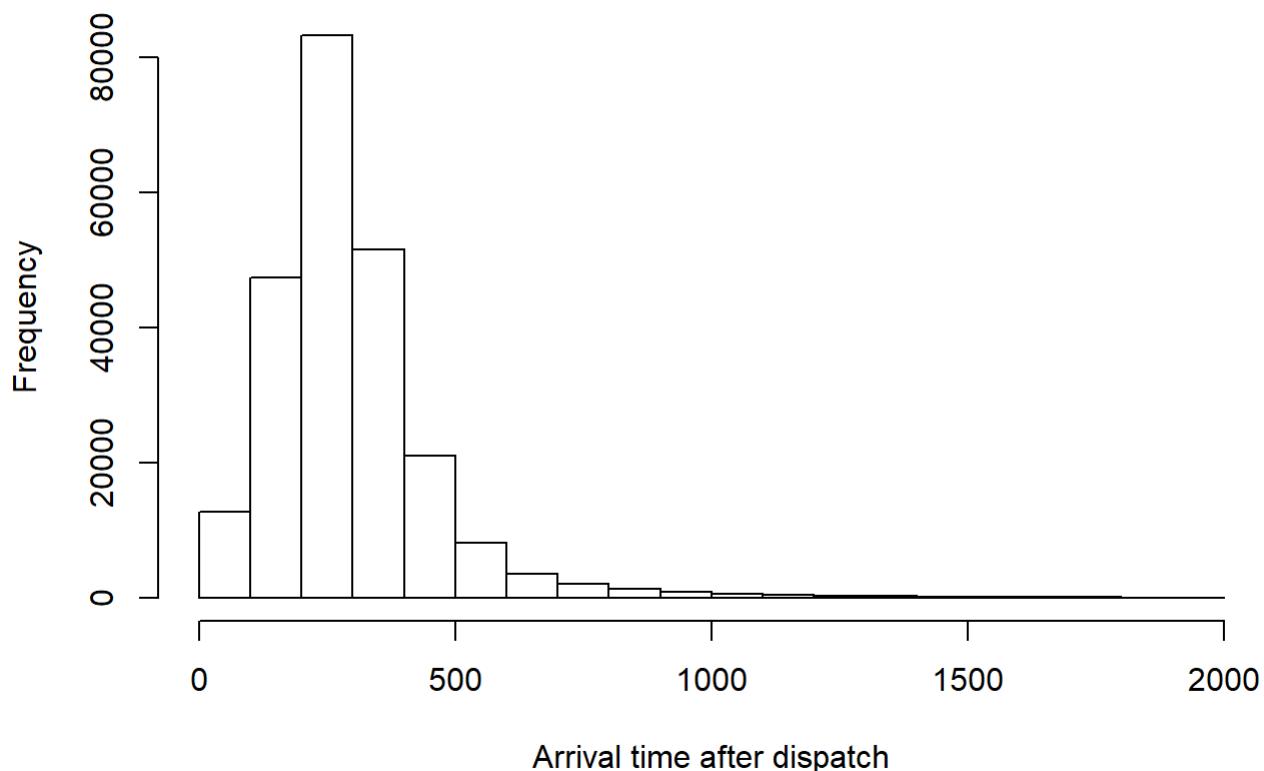
```
# Let's quickly see the distribution of PU.time
hist(cin_fire_new$PU.time %>% as.integer(), xlab = "Arrival time after dispatch",
     main = "Distribution of arrival times")
```

Distribution of arrival times



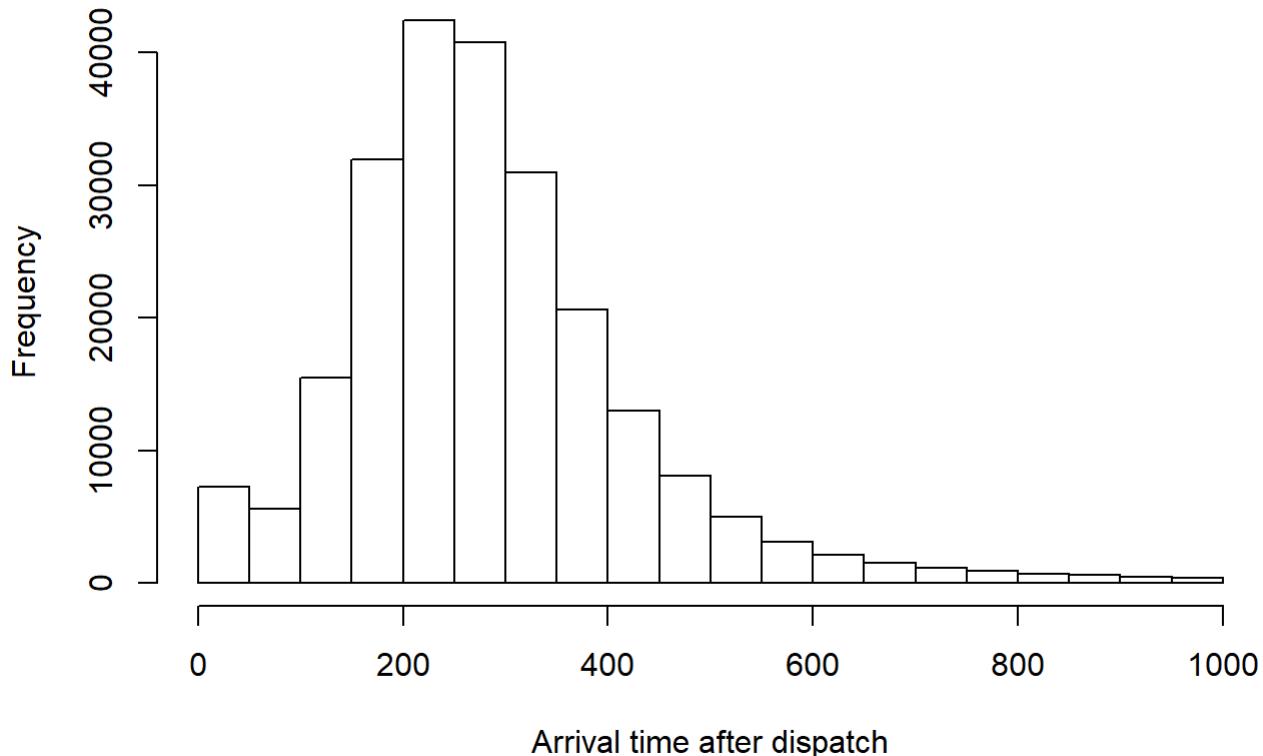
```
# Extraordinary outliers, subsetting required to get a clearer picture
hist(cin_fire_new$PU.time %>% as.integer() %>% .[cin_fire_new$PU.time < 2000], xlab = "Arrival t
ime after dispatch", main = "Distribution of arrival times")
```

Distribution of arrival times



```
# Still not great, zooming in further
hist(cin_fire_new$PU.time %>% as.integer() %>% .[cin_fire_new$PU.time < 1000], xlab = "Arrival time after dispatch", main = "Distribution of arrival times")
```

Distribution of arrival times



```
# Okay, this is much better! We can see that the time taken by fire officials vary
# mostly between 150 and 600 seconds.
```

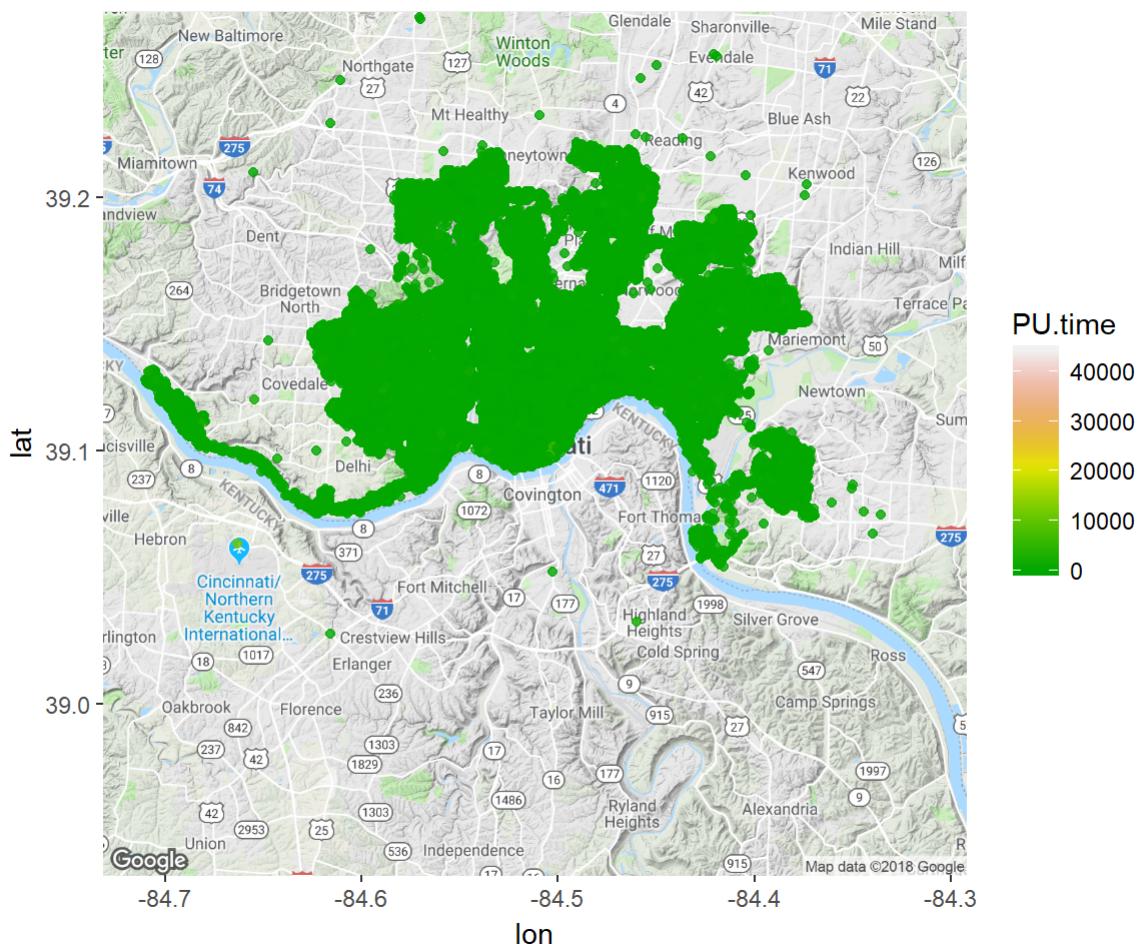
```
# Selecting the coordinates for spatial visualizations of PU.time variable

cin_fire_new$PU.time <- as.numeric(cin_fire_new$PU.time)

cin_fire_time <- cin_fire_new %>% select(Longitude, Latitude, PU.time, Agency, Month, DayoftheWeek, Hour) %>% filter(!is.na(Longitude) | !is.na(Latitude)) %>% filter(!is.na(PU.time))

##The heat density maps are used to qualify the areas with most fire calls.
cin_map + geom_jitter(aes(x = Longitude, y = Latitude, color = PU.time),
                      data = cin_fire_time, alpha = .8) + scale_color_gradientn(colors = terrain.colors(20))
```

```
## Warning: Removed 7 rows containing missing values (geom_point).
```



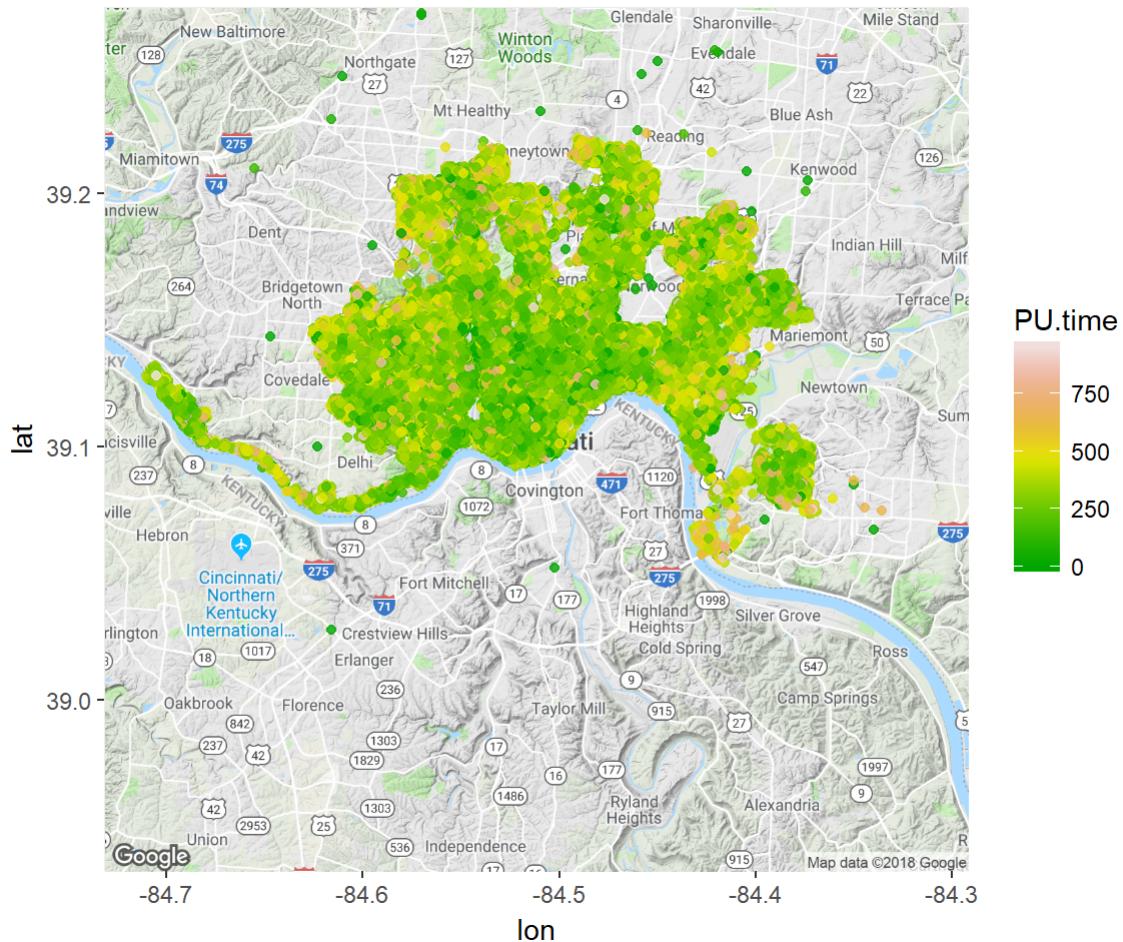
```
# The above plot is not very informative as majority of the points are below 1000
# if we recall from the histogram of the distribution of PU.time.
```

```
# Selecting the coordinates for spatial visualizations of PU.time variable, keeping
# the value of the variable below 1000.
```

```
cin_fire_time <- cin_fire_time %>% filter(PU.time < 1000)

##The heat density maps are used to qualify the areas with most fire calls.
cin_map + geom_jitter(aes(x = Longitude, y = Latitude, color = PU.time),
                      data = cin_fire_time, alpha = .8) + scale_color_gradientn(colors = terrain.colors(20))
```

```
## Warning: Removed 5 rows containing missing values (geom_point).
```



```
## Okay! The above plot is a much better visualization of the arrival times on the city map.
```

```
# Let's take this one step further by binning PU.time and create a more
# focused data visualization based on different binned classes of arrival times.
# Redefining cin_fire_new_maps to include all the data
cin_fire_time_bin <- cin_fire_new %>% select(Longitude, Latitude, PU.time, Agency, Month,
DayoftheWeek, Hour) %>% filter(!is.na(Longitude) | !is.na(Latitude)) %>%
filter(!is.na(PU.time))

cin_fire_time_bin$PU.time <- as.numeric(cin_fire_time_bin$PU.time)

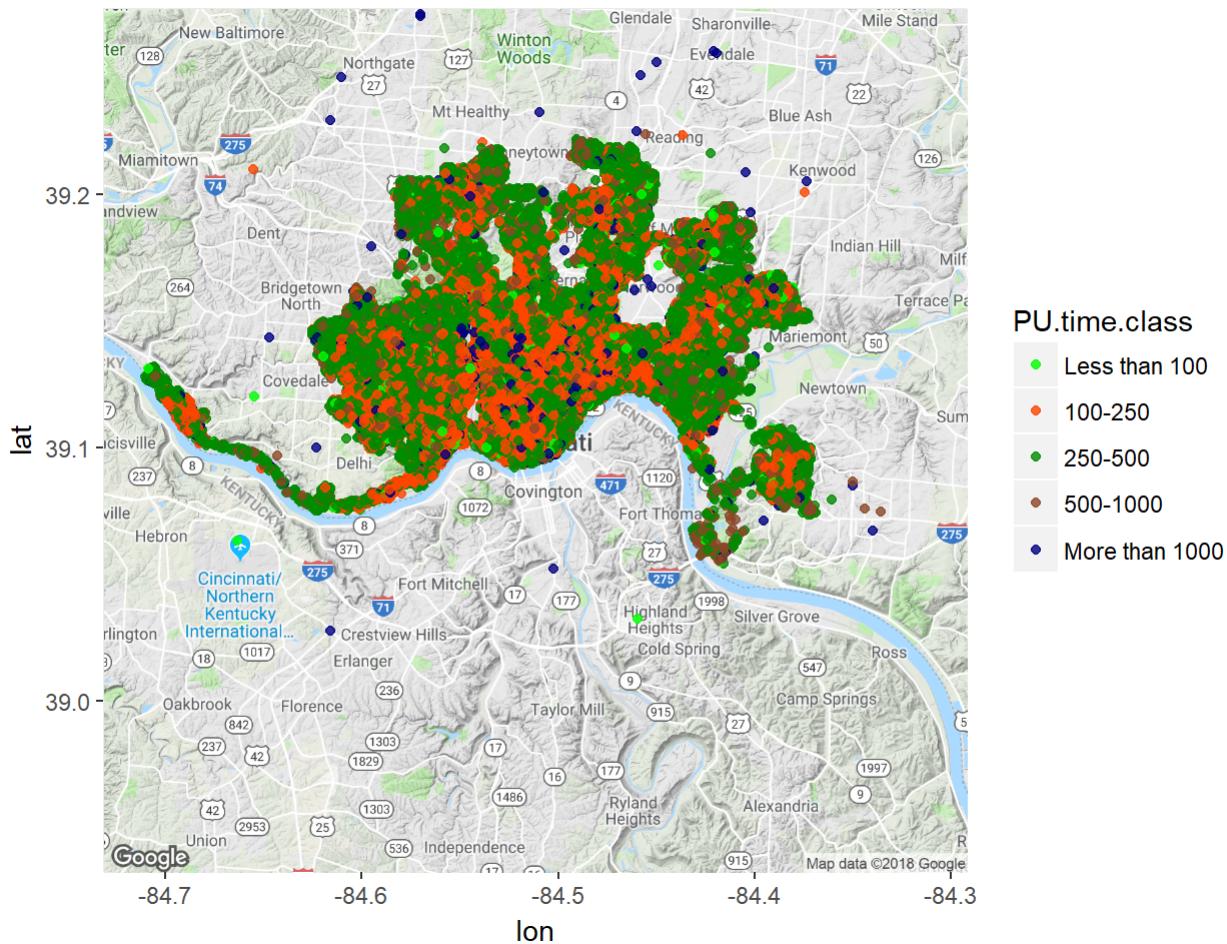
cin_fire_time_bin <- mutate(cin_fire_time_bin, PU.time.class = ifelse(PU.time %in% 0:99, "Less than 100",
ifelse(PU.time %in% 100:249, "100-250", ifelse(PU.time %in% 250:499, "250-500",
ifelse(PU.time %in% 500:999, "500-1000", ifelse(PU.time %in% 1000:45000, "More than 1000", "000")))))

cin_fire_time_bin$PU.time.class <- as.factor(cin_fire_time_bin$PU.time.class)

levels(cin_fire_time_bin$PU.time.class) <- c("Less than 100", "100-250", "250-500",
"500-1000", "More than 1000")

cin_map + geom_jitter(aes(x = Longitude, y = Latitude, color = PU.time.class),
data = cin_fire_time_bin, alpha = .8) + scale_alpha(guide = FALSE) +
scale_color_manual(values = c("green", "orangered", "green4", "sienna4", "navyblue"))
```

```
## Warning: Removed 7 rows containing missing values (geom_point).
```



```
## okay, we have a good spatial visualization of the fire calls with different classes of
## time taken to arrive at the place. Interestingly, most of the calls with most amount of
## time lapse are scattered on the outskirts!
```