A Mapping of Denver Marijuana Businesses and Arrests

Author: David Martinez Institution: Regis University Class: MSDS692 Data Science Practicum #1 Professor: Dr. Michael Busch

Introduction

On November 7th, 2000, voters in Colorado amended the state Constitution to allow the sale and use of Marijuana upon written consent by medical professional (Amendment 20, Colorado Constitution, 2000). Twelve years later, Colorado approved the sale and use of marijuana recreational use for adults over the age of twenty-one (Amendment 64, COlorado Constitution, 2012). Denver, the state capital and the largest population center in Colorado, has published data records since 2010 for medical marijuana and 2013 for recreational marijuana, including sales, government revenue, licensing information, and crime statistics.

This project will focus on identifying the types and locations of Marijuana businesses as well as the types and locations of arrests made.

The following datasets are used in this project:

https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-marijuana-active-business-licenses

https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-crime-marijuana

Other references: https://ballotpedia.org/Marijuana_on_the_ballot

https://developers.google.com/maps/documentation/geocoding/usage-and-billing

citations:

ggmap - D. Kahle and H. Wickham. ggmap: Spatial Visualization with ggplot2. The R Journal, 5(1), 144-161. URL http://journal.r-project.org/archive/2013-1/kahle-wickham.pdf

stringr - Hadley Wickham (2017). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.2.0. https://CRAN.R-project.org/package=stringr

ggplot2 - H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2009.

dplyr - Hadley Wickham, Romain Francois, Lionel Henry and Kirill Müller (2017). dplyr: A Grammar of Data Manipulation. R package version 0.7.4. https://CRAN.R-project.org/package=dplyr

RgoogleMaps - Markus Loecher and Karl Ropkins (2015). RgoogleMaps and loa: Unleashing R Graphics Power on Map Tiles. Journal of Statistical Software 63(4), 1-18. URL http://www.jstatsoft.org/v63/i04/.

A google developer key is required to perfrom this operation. This operation is masked to protect my key. The next section contains the code (commented out).

```
#dependencies
library(stringr)  #string operations
library(ggplot2)  #graphics
library(ggmap)    #used for geocoding
```

```
## Google Maps API Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it: see citation("ggmap") for details.
```

```
library(RgoogleMaps)    #used for mapping lat/long to static map
library(dplyr)    #dataframe manipulation
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
#API Key is required from google for the next section.
#register_google(key = 'YOUR KEY HERE')
```

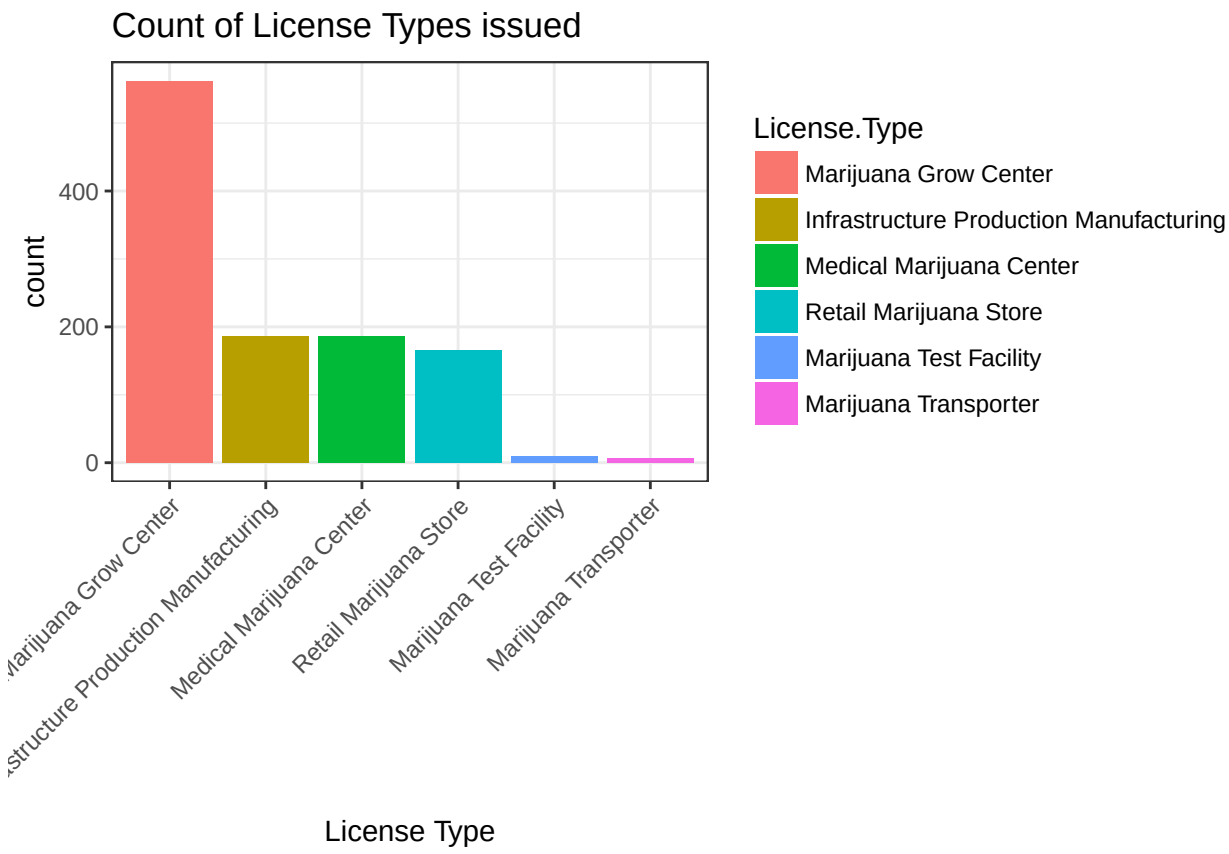
Retrieve Denver Marijuana Licenses

```
library(stringr)  #string operations
library(ggplot2)  #graphics
library(RgoogleMaps)    #used for geocoding
library(dplyr)    #dataframe manipulation

#retrieve dataset from denvergov.org
denver_mj_licenses <- read.csv("https://www.denvergov.org/media/gis/DataCatalog/marijuana_active_busines

#create factor for license type
denver_mj_licenses$License.Type <- factor(denver_mj_licenses$License.Type)

#combine levels and rename for easier reading
denver_mj_licenses$License.Type <- plyr::revalue(denver_mj_licenses$License.Type, c("Med Marijuana Inf

#create factor for zip code
denver_mj_licenses$Facility.Zip.Code <- factor(denver_mj_licenses$Facility.Zip.Code)

#current license status and expiration date don't appear to be very interesting so they can go
denver_mj_licenses <- denver_mj_licenses[-c(5:6)]

#convert street address information to one field for geocoding
num <- paste(word(denver_mj_licenses$Facility.Street.Number))
dir <- paste(word(denver_mj_licenses$Facility.Pre.Direction))
street <- paste(word(denver_mj_licenses$Facility.Street.Name))
type <- paste(word(denver_mj_licenses$Facility.Street.Type))
denver_mj_licenses$ADDRESS <- paste(num, dir, street, type, ", DENVER, CO", sep=" ")

#remove old address information (except zip code)
denver_mj_licenses <- denver_mj_licenses[-c(5:10)]

#geocode for lat/long - Data Science Toolkit (dsk) is used here instead of to retrieve the lat/long
for(i in 1:nrow(denver_mj_licenses)) {
  result <- geocode(denver_mj_licenses$ADDRESS[i], output="latlona", source="google")
  denver_mj_licenses$LONGITUDE[i] <- as.numeric(result[1])
  denver_mj_licenses$LATITUDE[i] <- as.numeric(result[2])
}

#save for posterity
#write.csv(denver_mj_licenses, "mj_licenses_geocoded.csv", row.names=FALSE)
```

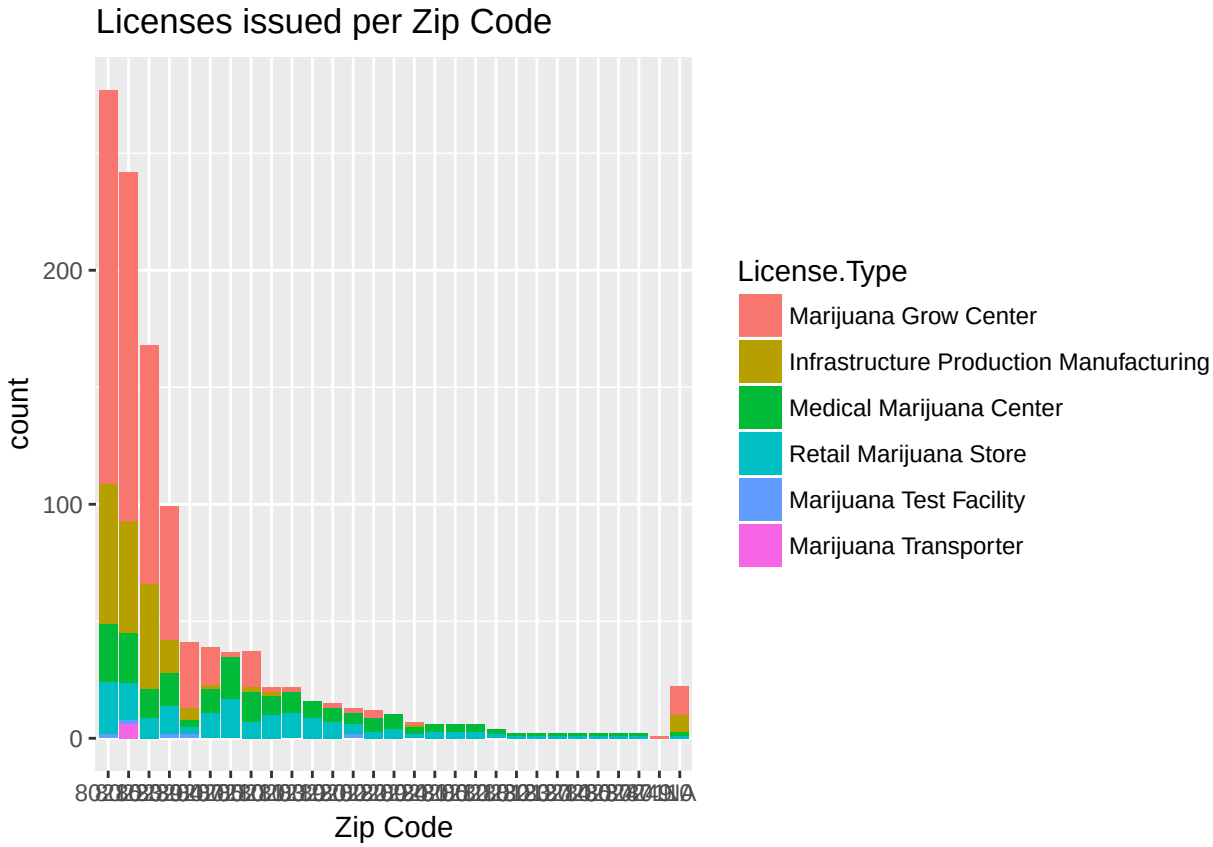
```r
summary(denver_mj_licenses)
```

```
##  Business.File.Number                                  License.Type
##  Length:1116          Marijuana Transporter                  :  6
##  Class :character     Infrastructure Production Manufacturing:186
##  Mode  :character     Marijuana Grow Center                  :562
##                       Marijuana Test Facility                : 10
##                       Medical Marijuana Center               :186
##                       Retail Marijuana Store                 :166
##
##  Entity.Name         Trade.Name        Facility.Zip.Code
##  Length:1116        Length:1116        80216  :277
##  Class :character   Class :character   80223  :242
##  Mode  :character   Mode  :character   80239  :168
##                                        80204  : 99
##                                        80207  : 41
##                                        (Other):267
##                                        NA's   : 22
##    ADDRESS             LONGITUDE         LATITUDE
##  Length:1116        Min.   :-105.1   Min.   :39.63
##  Class :character   1st Qu.:-105.0   1st Qu.:39.71
##  Mode  :character   Median :-105.0   Median :39.75
##                     Mean   :-105.0   Mean   :39.74
##                     3rd Qu.:-104.9   3rd Qu.:39.78
##                     Max.   :-104.8   Max.   :39.84
##
```

```r
#cleanup
rm(i, num, dir, street, type, result)

#sort by License.Type
denver_mj_licenses <- within(denver_mj_licenses, License.Type <- factor(License.Type,
                                                                        levels=names(sort(table(License

#generate plot to show graphically the number of license types issued
ggplot(denver_mj_licenses,aes(x=License.Type, fill=License.Type))+
  geom_bar()+
  labs(title="Count of License Types issued", x= "License Type")+
  theme_bw()+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
#sort by Facility Zip Code
denver_mj_licenses <- within(denver_mj_licenses, Facility.Zip.Code <- factor(Facility.Zip.Code,
                                                                              levels=names(sort(table(Facility.Z

#generate plot to show where licenses are being issued
ggplot(denver_mj_licenses,aes(x=Facility.Zip.Code, fill=License.Type))+
  geom_bar()+
  labs(title="Licenses issued per Zip Code", x="Zip Code")
```




```
  theme_bw()+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

```
## List of 57
##  $ line                :List of 6
##   ..$ colour       : chr "black"
##   ..$ size         : num 0.5
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : chr "black"
##   ..$ size         : num 0.5
##   ..$ linetype     : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                :List of 11
##   ..$ family       : chr ""
##   ..$ face         : chr "plain"
##   ..$ colour       : chr "black"
##   ..$ size         : num 11
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
```

```
##   ..$ angle        : num 0
##   ..$ lineheight   : num 0.9
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 5.5 0 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top     :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 5.5 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 5.5 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.right   :List of 11
```

```
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 0 5.5
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text            :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "grey30"
##   ..$ size         :Class 'rel'  num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 2.2 0 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top      :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 2.2 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
```

```
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 2.2 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.right    :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 0 2.2
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks           :List of 6
##   ..$ colour       : chr "grey20"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ axis.ticks.length    :Class 'unit'  atomic [1:1] 2.75
##   .. ..- attr(*, "valid.unit")= int 8
##   .. ..- attr(*, "unit")= chr "pt"
##  $ axis.line            : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.line.x          : NULL
##  $ axis.line.y          : NULL
##  $ legend.background    :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
```

```
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.margin        :Classes 'margin', 'unit'  atomic [1:4] 0.2 0.2 0.2 0.2
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  $ legend.spacing       :Class 'unit'  atomic [1:1] 0.4
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  $ legend.spacing.x     : NULL
##  $ legend.spacing.y     : NULL
##  $ legend.key           :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.key.size      :Class 'unit'  atomic [1:1] 1.2
##   .. ..- attr(*, "valid.unit")= int 3
##   .. ..- attr(*, "unit")= chr "lines"
##  $ legend.key.height    : NULL
##  $ legend.key.width     : NULL
##  $ legend.text          :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         :Class 'rel'  num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.align    : NULL
##  $ legend.title         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.align   : NULL
##  $ legend.position      : chr "right"
##  $ legend.direction     : NULL
##  $ legend.justification : chr "center"
##  $ legend.box           : NULL
##  $ legend.box.margin    :Classes 'margin', 'unit'  atomic [1:4] 0 0 0 0
```

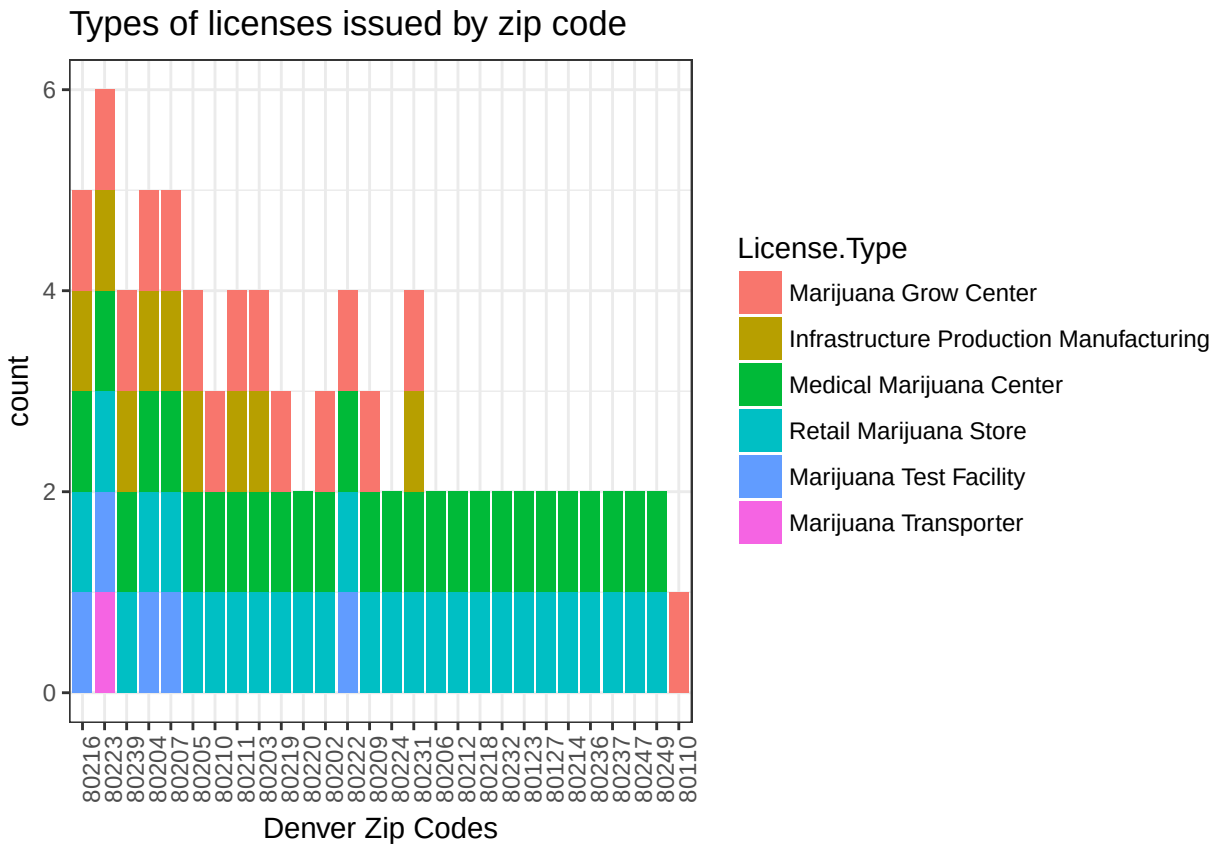
```
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  $ legend.box.background: list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing   :Class 'unit'  atomic [1:1] 0.4
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  $ panel.background     :List of 5
##   ..$ fill         : chr "white"
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.border         :List of 5
##   ..$ fill         : logi NA
##   ..$ colour       : chr "grey20"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.spacing        :Class 'unit'  atomic [1:1] 5.5
##   .. ..- attr(*, "valid.unit")= int 8
##   .. ..- attr(*, "unit")= chr "pt"
##  $ panel.spacing.x      : NULL
##  $ panel.spacing.y      : NULL
##  $ panel.grid.major     :List of 6
##   ..$ colour       : chr "grey92"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.minor     :List of 6
##   ..$ colour       : chr "grey92"
##   ..$ size         : num 0.25
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.ontop          : logi FALSE
##  $ plot.background      :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : chr "white"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ plot.title           :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
```

```
##   ..$ size         :Class 'rel'  num 1.2
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 6.6 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.subtitle        :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         :Class 'rel'  num 0.9
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 0 4.95 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         :Class 'rel'  num 0.9
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 4.95 0 0 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.margin          :Classes 'margin', 'unit'  atomic [1:4] 5.5 5.5 5.5 5.5
##   .. ..- attr(*, "valid.unit")= int 8
##   .. ..- attr(*, "unit")= chr "pt"
##  $ strip.background     :List of 5
##   ..$ fill         : chr "grey85"
##   ..$ colour       : chr "grey20"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ strip.placement      : chr "inside"
##  $ strip.text           :List of 11
##   ..$ family       : NULL
```

```
##   ..$ face         : NULL
##   ..$ colour       : chr "grey10"
##   ..$ size         :Class 'rel'  num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.x         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 5.5 0 5.5 0
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.y         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       :Classes 'margin', 'unit'  atomic [1:4] 0 5.5 0 5.5
##   .. .. ..- attr(*, "valid.unit")= int 8
##   .. .. ..- attr(*, "unit")= chr "pt"
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.switch.pad.grid:Class 'unit'  atomic [1:1] 0.1
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  $ strip.switch.pad.wrap:Class 'unit'  atomic [1:1] 0.1
##   .. ..- attr(*, "valid.unit")= int 1
##   .. ..- attr(*, "unit")= chr "cm"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

```
#create a dataframe of the types of licenses issued to each zip code
zip_tab <- as.data.frame(table(denver_mj_licenses$Facility.Zip.Code, denver_mj_licenses$License.Type))
```

```
#rename for more descriptive tags
zip_tab <- rename(zip_tab, ZipCode = Var1, License.Type = Var2)

#filter out the 0 frequency occurences
zip_tab <- filter(zip_tab, zip_tab$Freq > 0)

#generate a plot to show the types of licenses issued to each zip code
ggplot(zip_tab,aes(x=ZipCode, fill=License.Type))+
  geom_bar()+
  labs(title="Types of licenses issued by zip code", x= "Denver Zip Codes")+
  theme_bw()+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Clean and save marijuana crime dataset

```
#retrieve the DPD marijuana crime file from Denver open data portal
denver_mj_crime <- read.csv("https://www.denvergov.org/media/gis/DataCatalog/crime_marijuana/csv/crime_

#write original file to disk for posterity
write.csv(denver_mj_crime, "crime_marijuana.csv", row.names=FALSE)

#incident ID to character
denver_mj_crime$INCIDENT_ID <- as.character(denver_mj_crime$INCIDENT_ID)

#Only one date field is needed
```

```
denver_mj_crime <- data.frame(denver_mj_crime[-c(2:3, 6:7, 10:11)])

#The Report date needs to be in an R compliant format
day <- paste(word(denver_mj_crime$REPORTDATE, 1, sep="-"))
month <- paste(word(denver_mj_crime$REPORTDATE, 2, sep="-"))
month <- plyr::revalue(month, c("JAN"="01", "FEB"="02", "MAR"="03", "APR"="04", "MAY"="05", "JUN"="06",
year <- as.integer(paste(word(denver_mj_crime$REPORTDATE, -1, sep="-")))
year <- paste(year+2000)

#denver_mj_crime$REPORTDATE <- paste(year, month, sep="-")
denver_mj_crime$REPORTDATE <- paste(year, month, day, sep="-")
denver_mj_crime$REPORTDATE <- as.Date(denver_mj_crime$REPORTDATE)

#append city and state information to address - necessary for geocoding
denver_mj_crime$INCIDENT_ADDRESS <- sapply(denver_mj_crime$INCIDENT_ADDRESS , paste, ", Denver, CO", sep

#geo_x, geo_y are not needed for this activity
#denver_mj_crime <- data.frame(denver_mj_crime[-c(4:5)])

#convert fields to factors
denver_mj_crime$DISTRICT_ID <- factor(denver_mj_crime$DISTRICT_ID)
denver_mj_crime$PRECINCT_ID <- factor(denver_mj_crime$PRECINCT_ID)
denver_mj_crime$OFFENSE_CATEGORY_ID <- factor(denver_mj_crime$OFFENSE_CATEGORY_ID)
denver_mj_crime$MJ_RELATION_TYPE <- factor(denver_mj_crime$MJ_RELATION_TYPE)
denver_mj_crime$NEIGHBORHOOD_ID <- factor(denver_mj_crime$NEIGHBORHOOD_ID)

#geocode to add Longitude/Latitude data
for(i in 1:nrow(denver_mj_crime)) {
  result <- geocode(denver_mj_crime$INCIDENT_ADDRESS[i], output="latlona", source="google")
  denver_mj_crime$LONGITUDE[i] <- as.numeric(result[1])
  denver_mj_crime$LATITUDE[i] <- as.numeric(result[2])
}
```

```
## Warning: geocode failed with status ZERO_RESULTS, location = "1499 N
## BROADWAY ST, Denver, CO"

## Warning: geocode failed with status ZERO_RESULTS, location = "1499 N
## BROADWAY ST, Denver, CO"

## Warning: geocode failed with status ZERO_RESULTS, location = "1499 N
## BROADWAY ST, Denver, CO"

## Warning: geocode failed with status ZERO_RESULTS, location = "1499 N
## BROADWAY ST, Denver, CO"

## Warning: geocode failed with status ZERO_RESULTS, location = "1499 N
## BROADWAY ST, Denver, CO"
```

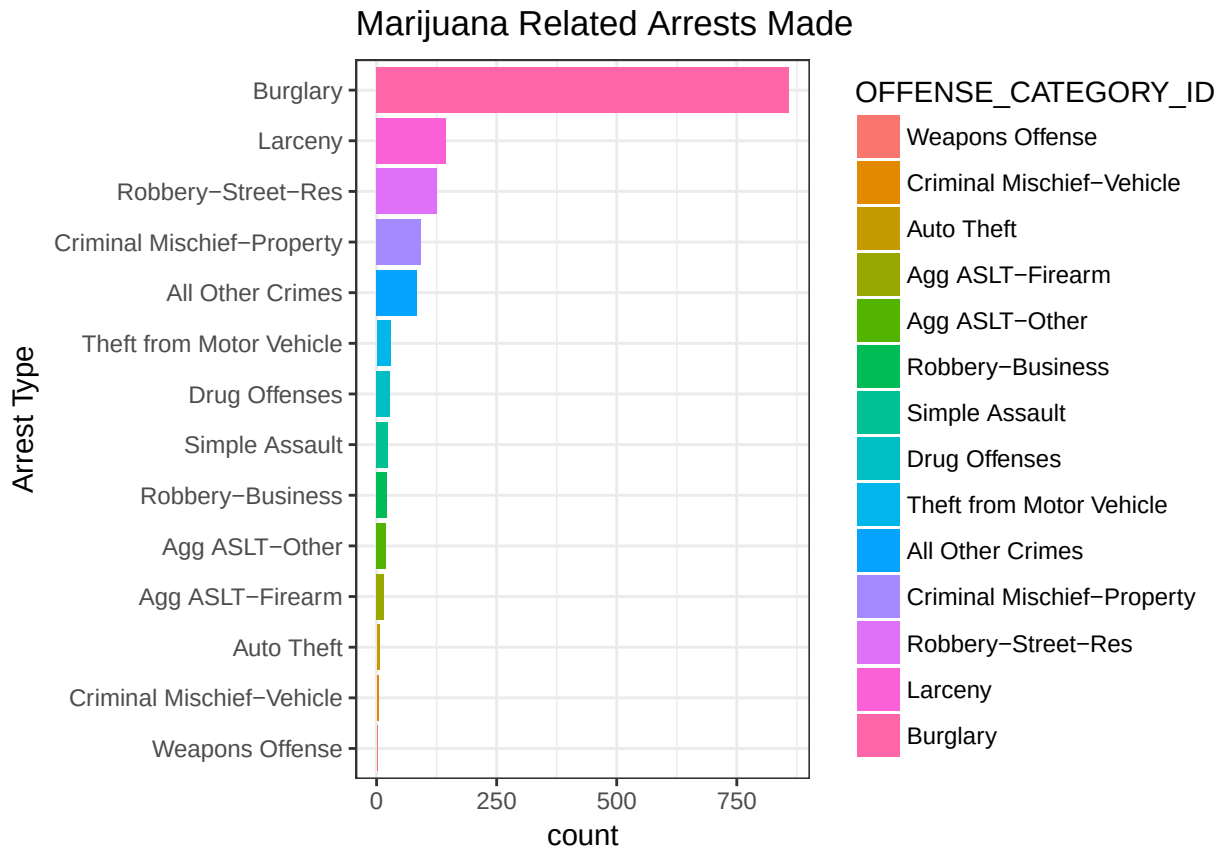
```
summary(denver_mj_crime)
```

```
##  INCIDENT_ID          REPORTDATE         INCIDENT_ADDRESS    DISTRICT_ID
##  Length:1454        Min.   :2012-01-03   Length:1454        3      :375
##  Class :character   1st Qu.:2013-08-08   Class :character   2      :329
##  Mode  :character   Median :2014-12-18   Mode  :character   1      :272
```

```
##                    Mean   :2014-12-16                    4      :265
##                    3rd Qu.:2016-05-21                    6      :137
##                    Max.   :2017-12-17                    5      : 74
##                                                          (Other):  2
##   PRECINCT_ID                  OFFENSE_CATEGORY_ID      MJ_RELATION_TYPE
##  313    :145   Burglary                  :859      INDUSTRY\n    :1039
##  212    :130   Larceny                   :144      NON-INDUSTRY\n: 415
##  112    : 83   Robbery-Street-Res        :125
##  422    : 81   Criminal Mischief-Property: 92
##  412    : 76   All Other Crimes          : 84
##  411    : 73   Theft from Motor Vehicle  : 29
##  (Other):866   (Other)                   :121
##        NEIGHBORHOOD_ID   LONGITUDE         LATITUDE
##  elyria-swansea:129    Min.   :-105.1   Min.   :39.63
##  overland      : 87    1st Qu.:-105.0   1st Qu.:39.70
##  globeville    : 67    Median :-105.0   Median :39.73
##  five-points   : 65    Mean   :-105.0   Mean   :39.73
##  valverde      : 54    3rd Qu.:-104.9   3rd Qu.:39.77
##  montbello     : 53    Max.   :-104.7   Max.   :39.84
##  (Other)       :999    NA's   :5        NA's   :5
```

```
#remove intermediate variables and capture the clean file for posterity
rm(result, i, day, month, year)

#sort by neighborhood
denver_mj_crime <- within(denver_mj_crime, NEIGHBORHOOD_ID <- factor(NEIGHBORHOOD_ID,
                                                                     levels=names(sort(table(NEIGHBORH

#generate plot to show neighborhoods and arrests
ggplot(denver_mj_crime,aes(x=NEIGHBORHOOD_ID, fill=OFFENSE_CATEGORY_ID))+
  geom_bar()+
  coord_flip()+
  labs(title="License Types issued by zip code", x= "Denver Neighborhoods")+
  theme_bw()
```

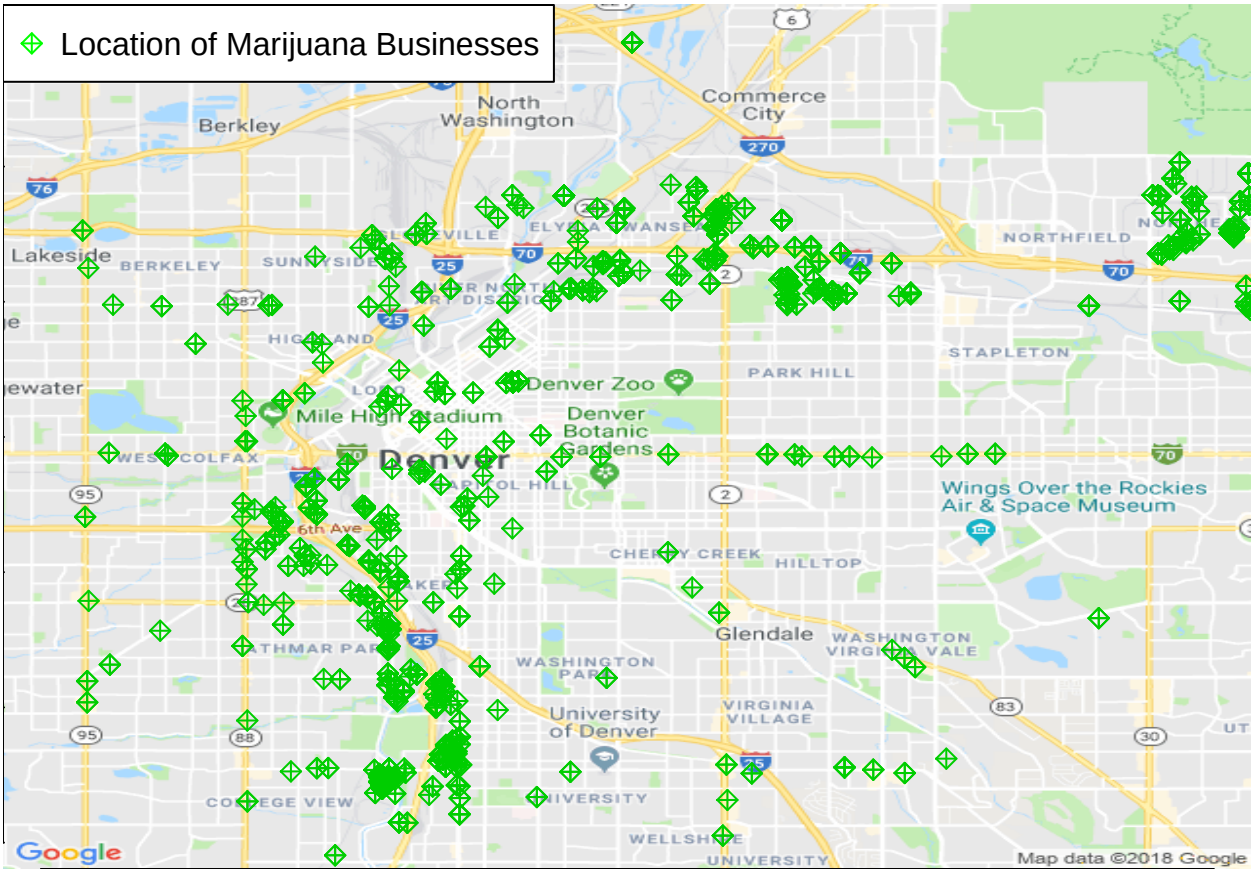

```
#sort by crime type
denver_mj_crime <- within(denver_mj_crime, OFFENSE_CATEGORY_ID <- factor(OFFENSE_CATEGORY_ID,
                                                                         levels=names(sort(table(OFFENSE_CA

#generate plot to show where licenses are being issued
ggplot(denver_mj_crime,aes(x=OFFENSE_CATEGORY_ID, fill=OFFENSE_CATEGORY_ID))+
  geom_bar()+
  coord_flip()+
  labels(title="Marijuana Related Arrests Made", x= "Arrest Type")+
  theme_bw()
```



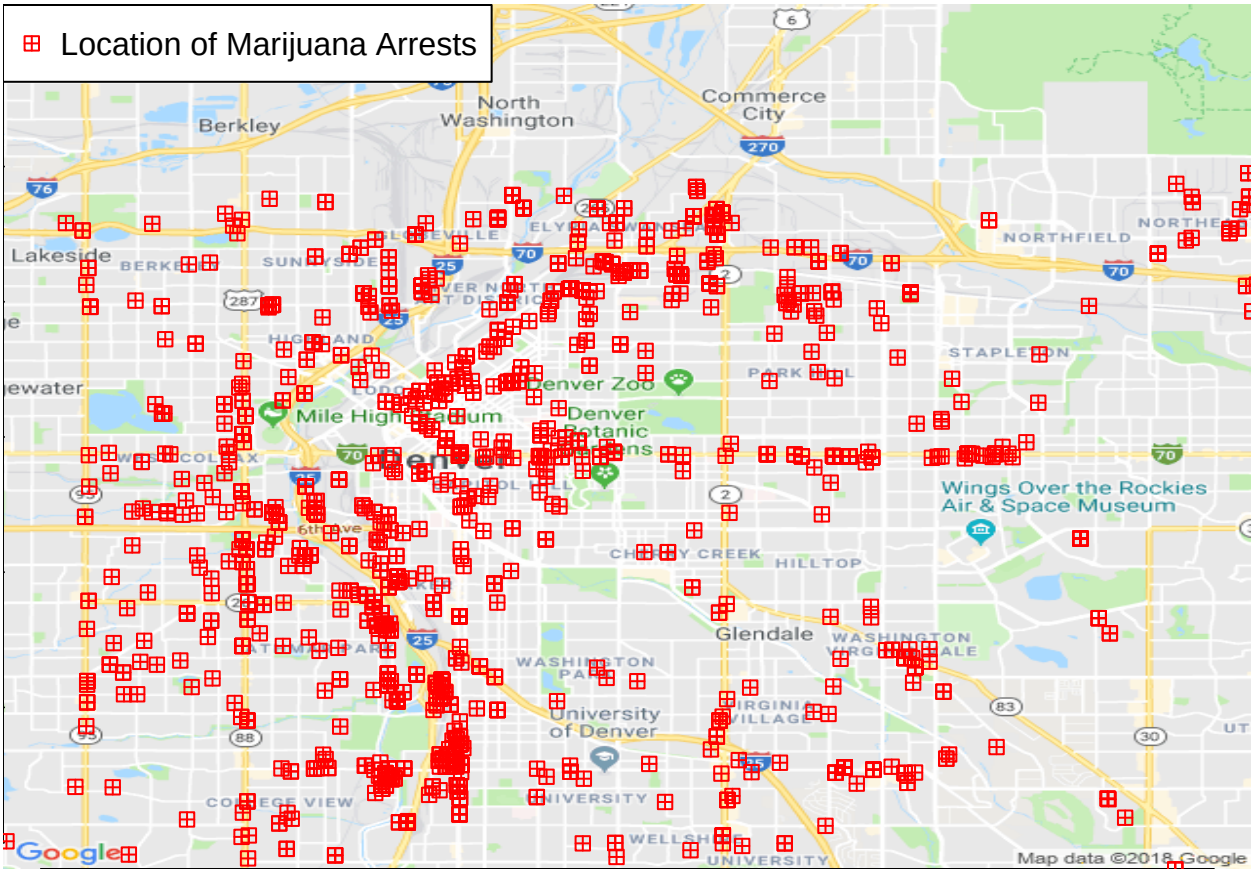
This section requires a Google developers API Key: (https://developers.google.com/maps/documentation/geocoding/usage-and-billing).

```
#apikey <- "YOUR KEY HERE"
```

```
library(RgoogleMaps) #interface to google maps

#generate denver map.
denver_map <- GetMap(center = c(lat = mean(denver_mj_licenses$LATITUDE), lon = mean(denver_mj_licenses$

p1 <- PlotOnStaticMap(denver_map, lat = denver_mj_licenses$LATITUDE, lon = denver_mj_licenses$LONGITUDE
legend("topleft", legend = "Location of Marijuana Businesses", col = "green", bg = "white", pch=9)
```




```
p2 <- PlotOnStaticMap(denver_map, lat = denver_mj_crime$LATITUDE, lon = denver_mj_crime$LONGITUDE, dest
legend("topleft", legend = "Location of Marijuana Arrests", col = "red", bg = "white", pch=12)
```


Location of Marijuana Arrests
Berkley
North Washington
Commerce City
Lakeside
Denver Zoo
Mile High Stadium
Denver Botanic Gardens
Wings Over the Rockies Air & Space Museum
Glendale
University of Denver
Google
Map data ©2018 Google