

# Assignment 4 Analysis

*Josh Burroughs*

*February 28, 2016*

```
library(data.table)
library(bit64)

## Loading required package: bit

## Attaching package bit

## package:bit (c) 2008-2012 Jens Oehlschlaegel (GPL-2)

## creators: bit bitwhich

## coercion: as.logical as.integer as.bit as.bitwhich which

## operator: ! & | xor != ==

## querying: print length any all min max range sum summary

## bit access: length<- [ [<- [[ [[<-

## for more help type ?bit

##
## Attaching package: 'bit'

## The following object is masked from 'package:data.table':
##       setattr

## The following object is masked from 'package:base':
##       xor

## Attaching package bit64

## package:bit64 (c) 2011-2012 Jens Oehlschlaegel (GPL-2 with commercial restrictions)

## creators: integer64 seq :

## coercion: as.integer64 as.vector as.logical as.integer as.double as.character as.bin

## logical operator: ! & | xor != == < <= > = >
```

```

## arithmetic operator: + - * / %/% %% ^
## math: sign abs sqrt log log2 log10
## math: floor ceiling trunc round
## querying: is.integer64 is.vector [is.atomic] [length] is.na format print
## aggregation: any all min max range sum prod
## cumulation: diff cummin cummax cumsum cumprod
## access: length<- [ [<- [[[[<-
## combine: c rep cbind rbind as.data.frame
## for more help type ?bit64
##
## Attaching package: 'bit64'
## The following object is masked from 'package:bit':
##   still.identical
## The following objects are masked from 'package:base':
##   :, %in%, is.double, match, order, rank

```

## San Francisco

Downloaded from <https://data.sfgov.org/view/vsk2-um2x>

```

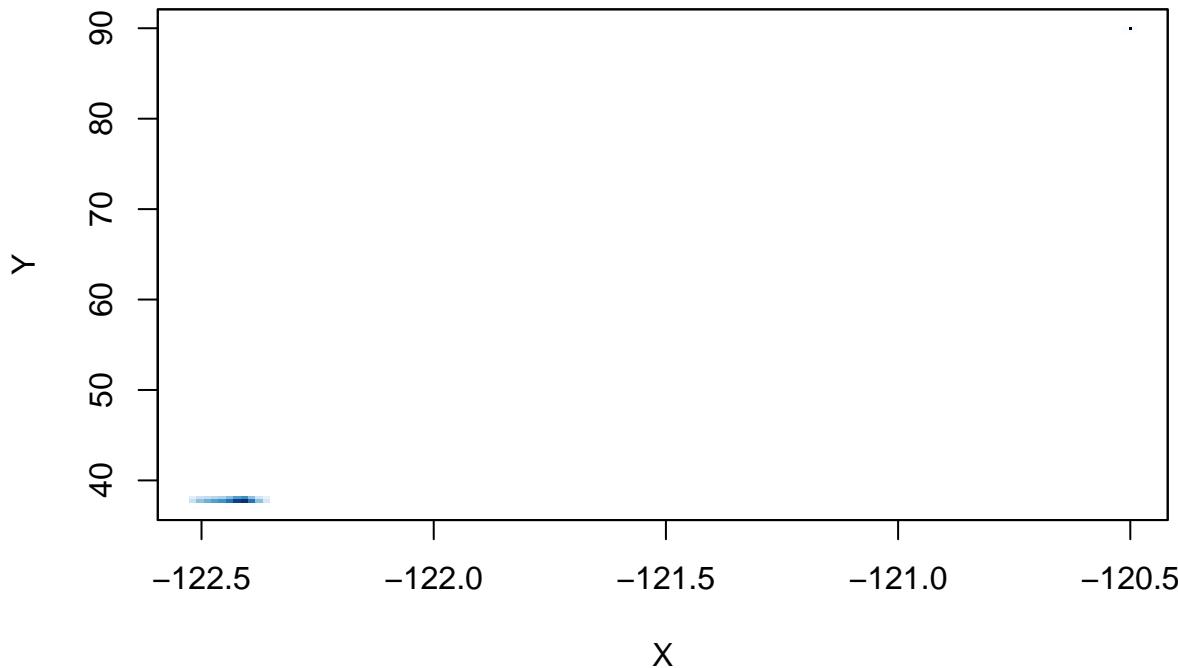
sf <- fread("data/sf/SFPD_Incidents_-_from_1_January_2003.csv")
## 
Read 31.9% of 1878050 rows
Read 64.4% of 1878050 rows
Read 96.4% of 1878050 rows
Read 1878050 rows and 17 (of 17) columns from 0.348 GB file in 00:00:05

setnames(sf,make.names(names(sf)))
sf.factor.cols <- c('Category','Descript','DayOfWeek','Resolution')
sf[,sf.factor.cols := lapply(.SD,factor),.SDcols = sf.factor.cols, with=F]
sf[,Date := as.POSIXct(Date,format='%m/%d/%Y',tz='PST')]
t_to_secs <- function(time) {
  lt <- as.POSIXlt(time,format='%M:%S')
  lt$min * 60 + lt$sec
}
sf[,DateTime := Date + t_to_secs(Time)]

```

```
library(ggplot2)
with(sf, smoothScatter(x=X, y=Y))
```

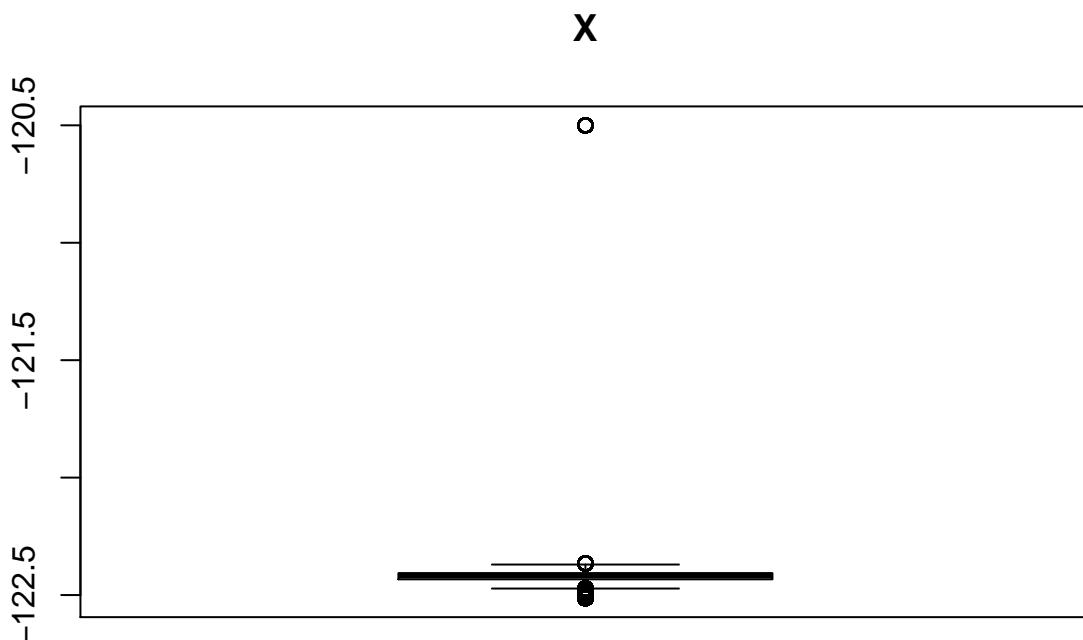
```
## Warning in KernSmooth::bkde2D(x, bandwidth = bandwidth, gridsize = nbin, :
## Binning grid too coarse for current (small) bandwidth: consider increasing
## 'gridsize'
```



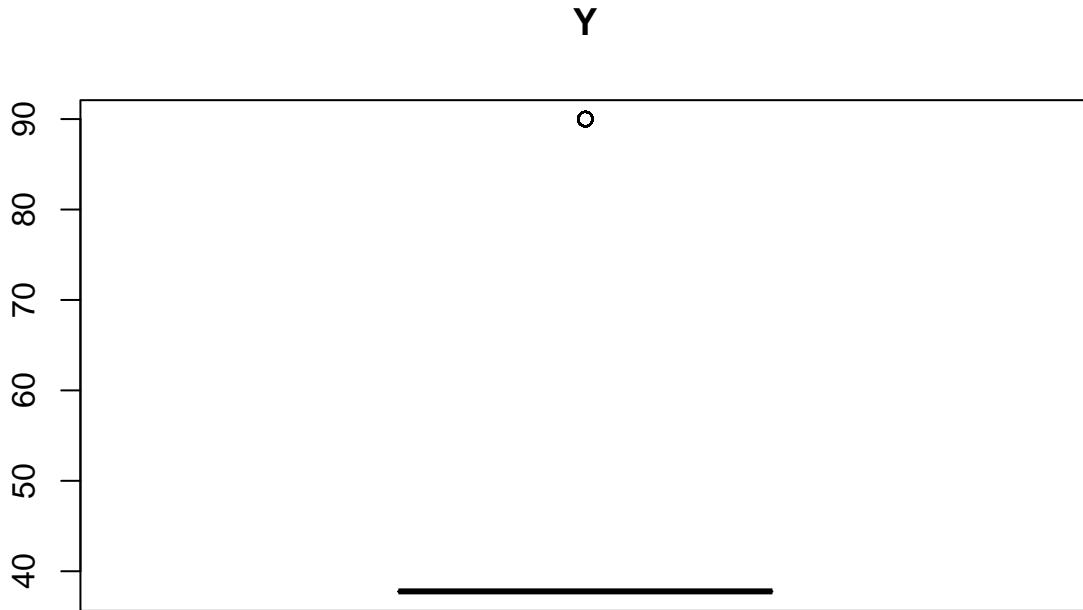
Data

outside the main geographical area:

```
boxplot(sf$X)
title("X")
```



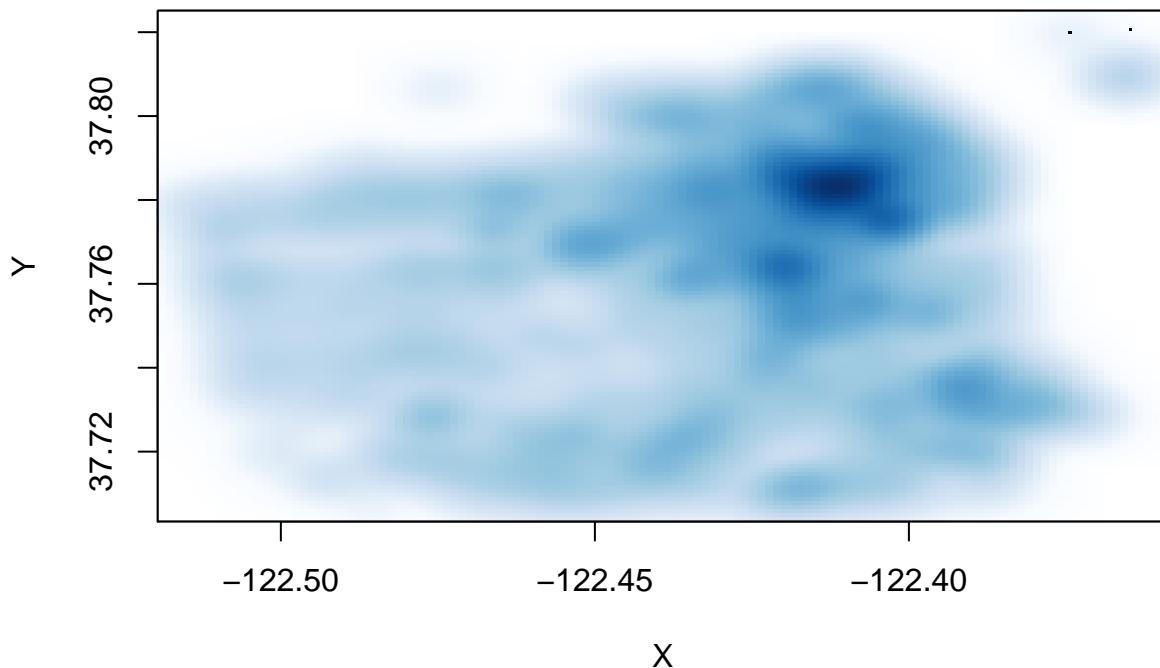
```
boxplot(sf$Y)
title("Y")
```



```
max.loc = sf[, sapply(.SD, max), .SDcols=c('X', 'Y')]
```

Looks like points at -120.5, 90 are bogus

```
sf[X == max.loc[['X']] & Y == max.loc[['Y']], `:=` (X=NA, Y=NA)]
with(sf, smoothScatter(x=X, y=Y))
```



Looking for murders:

```
sf[grep('HOMICIDE|MURDER|SLAUGHTER',Descript),unique(Descript)]
```

```
## [1] ATTEMPTED HOMICIDE WITH A KNIFE  
## [2] ATTEMPTED HOMICIDE WITH A GUN  
## [3] ATTEMPTED HOMICIDE WITH A DANGEROUS WEAPON  
## [4] ATTEMPTED HOMICIDE WITH BODILY FORCE  
## [5] DEATH, NON-MANSLAUGHTER AUTO ACCIDENT  
## [6] ASSAULT OR ATTEMPTED MURDER UPON GOV'T OFFICERS  
## [7] ATTEMPTED HOMICIDE WITH EXPLOSIVES  
## [8] ATTEMPTED HOMICIDE BY SNIPING  
## 913 Levels: ABANDONMENT OF CHILD ... YOUTH COURT
```

```
sf[grep('ASSAULT',Descript),unique(Descript)]
```

```
## [1] AGGRAVATED ASSAULT WITH BODILY FORCE  
## [2] AGGRAVATED ASSAULT WITH A DEADLY WEAPON  
## [3] AGGRAVATED ASSAULT WITH A KNIFE  
## [4] ASSAULT, AGGRAVATED, W/ GUN  
## [5] ASSAULT WITH CAUSTIC CHEMICALS  
## [6] ASSAULT  
## [7] ASSAULT TO RAPE WITH BODILY FORCE  
## [8] AGGRAVATED ASSAULT OF POLICE OFFICER,BODILY FORCE  
## [9] ASSAULT ON A POLICE OFFICER WITH A DEADLY WEAPON  
## [10] SEXUAL ASSAULT, AGGRAVATED, OF CHILD  
## [11] POSS OF DEADLY WEAPON WITH INTENT TO ASSAULT  
## [12] ASSAULT, AGGRAVATED, ON POLICE OFFICER, W/ GUN  
## [13] SEXUAL ASSAULT, ADMINISTERING DRUG TO COMMIT  
## [14] ATTEMPTED SIMPLE ASSAULT  
## [15] ASSAULT TO COMMIT MAYHEM OR SPECIFIC SEX OFFENSES  
## [16] AGGRAVATED ASSAULT WITH A GUN  
## [17] ASSAULT TO RAPE WITH A SHARP INSTRUMENT  
## [18] WEAPON, ASSAULT, POSSESSION, MANUFACTURE, OR SALE  
## [19] WEAPON, DEADLY, CARRYING WITH INTENT TO COMMIT ASSAULT  
## [20] ASSAULT BY POISONING  
## [21] ASSAULT TO RAPE WITH A DANGEROUS WEAPON  
## [22] WEAPON, ASSAULT, REGISTRATION OR TRANSFER VIOLATION  
## [23] ASSAULT TO RAPE WITH A GUN  
## [24] AGGRAVATED ASSAULT ON POLICE OFFICER WITH A KNIFE  
## [25] ASSAULT, AGGRAVATED, W/ MACHINE GUN  
## [26] ASSAULT BY POLICE OFFICER  
## [27] ASSAULT, AGGRAVATED, W/ SEMI AUTO  
## [28] AGGRAVATED ASSAULT ON POLICE OFFICER WITH A GUN  
## [29] ASSAULT, AGGRAVATED, ON POLICE OFFICER, W/ FULL AUTO  
## [30] ASSAULT TO ROB WITH BODILY FORCE  
## [31] ASSAULT OR ATTEMPTED MURDER UPON GOV'T OFFICERS  
## [32] ASSAULT, AGGRAVATED, ON POLICE OFFICER, W/ SEMI AUTO  
## [33] ASSAULT BY JUVENILE SUSPECT  
## [34] ASSAULT TO ROB ON THE STREET W/DEADLY WEAPON  
## [35] ASSAULT TO ROB BANK WITH A GUN  
## [36] AGGRAVATED ASSAULT OF POLICE OFFICER, SNIPING  
## 913 Levels: ABANDONMENT OF CHILD ... YOUTH COURT
```

Only attempted homicide...

```

sf[grep('GRAND THEFT',Descript),unique(Descript)]


## [1] GRAND THEFT FROM LOCKED AUTO
## [2] GRAND THEFT BICYCLE
## [3] GRAND THEFT FROM PERSON
## [4] GRAND THEFT FROM A BUILDING
## [5] GRAND THEFT PICKPOCKET
## [6] GRAND THEFT SHOPLIFTING
## [7] GRAND THEFT OF PROPERTY
## [8] TRICK AND DEVICE, GRAND THEFT
## [9] EMBEZZLEMENT, GRAND THEFT BY EMPLOYEE
## [10] GRAND THEFT FROM UNLOCKED AUTO
## [11] GRAND THEFT AUTO STRIP
## [12] GRAND THEFT PURSESNATCH
## [13] AUTO, GRAND THEFT OF
## [14] FALSE PRETENSES, GRAND THEFT
## [15] LOST PROPERTY, GRAND THEFT
## [16] ATTEMPTED GRAND THEFT PURSESNATCH
## [17] EMBEZZLEMENT, GRAND THEFT
## [18] GRAND THEFT BY PROSTITUTE
## [19] ATTEMPTED GRAND THEFT FROM PERSON
## [20] EMBEZZLEMENT, GRAND THEFT BY BROOKERS/AGENTS
## [21] GRAND THEFT COIN OPERATED MACHINE
## [22] ATTEMPTED GRAND THEFT PICKPOCKET
## [23] EMBEZZLEMENT, GRAND THEFT LEASED PROPERTY
## [24] EMBEZZLEMENT, GRAND THEFT PRIVATE PROPERTY
## [25] GRAND THEFT MOTORCYCLE STRIP
## [26] SHORT CHANGE, GRAND THEFT
## [27] EMBEZZLEMENT, GRAND THEFT PUBLIC/PRIVATE OFFICIAL
## [28] EMBEZZLEMENT, GRAND THEFT BY PROPERTY CARRIER
## [29] GRAND THEFT PHONE BOOTH
## [30] EMBEZZLEMENT, GRAND THEFT BY COLLECTOR
## 913 Levels: ABANDONMENT OF CHILD ... YOUTH COURT

```

## Raleigh

Downloaded from <https://data.raleighnc.gov/Police/Police-Incident-Data-from-Jan-1-2005-Master-File/csw9-dd5k>

```

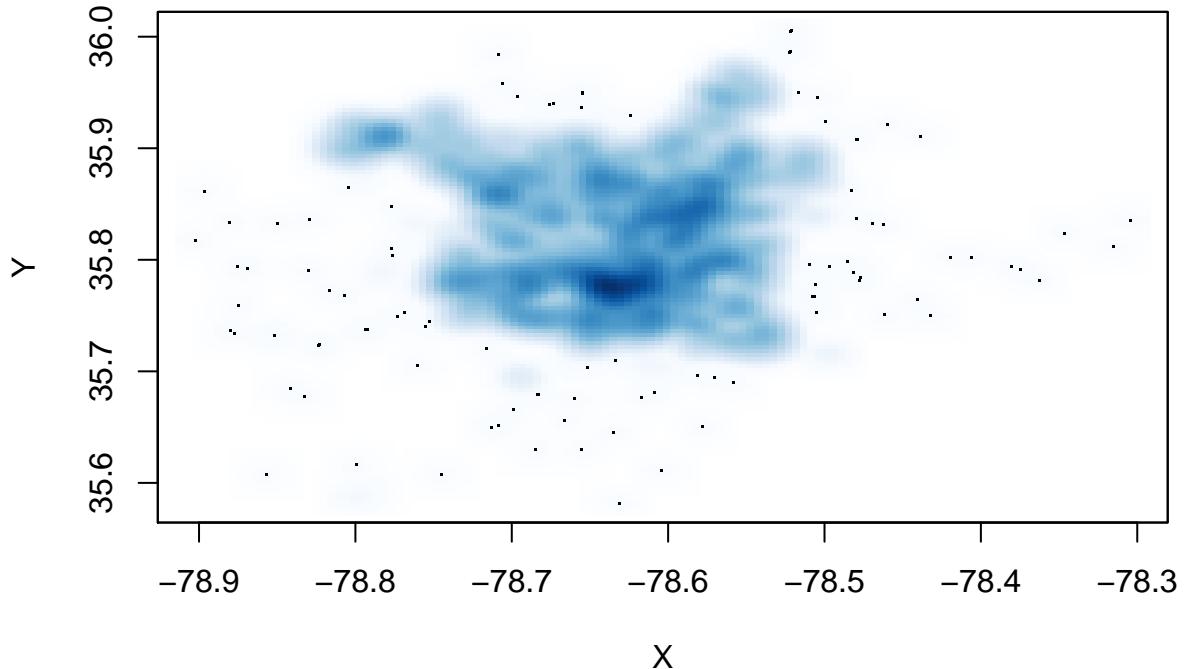
ral <- fread("data/raleigh/Police_Incident_Data_from_Jan_1_2005_-_Master_File.csv")

## 
## Read 87.1% of 493429 rows
## Read 493429 rows and 6 (of 6) columns from 0.052 GB file in 00:00:03

setnames(ral,make.names(names(ral)))
ral.factor.cols <- c('LCR','DISTRICT')
ral[,ral.factor.cols := lapply(.SD,factor),.SDcols = ral.factor.cols, with=F]
#split the Location column apart:
ral[,c('Y','X') := lapply(tstrsplit(gsub('[()]',' ',LOCATION),','),as.numeric)]
ral[,INC.DATETIME := as.POSIXct(INC.DATETIME,format="%m/%d/%Y %I:%M:%S %p",tz='EST')]

```

```
with(ral,smoothScatter(X,Y))
```



Incident Code:

```
ralgrep <- function(term) unique(ral[grep(term,LCR.DESC,ignore.case = T),.(LCR,LCR.DESC)])
ralgrep('murder')
```

```
##      LCR                  LCR.DESC
## 1: 11 Homicide/Murder-Nonnegligent Manslaughter
## 2: 11      MURDER AND NONNEGIGENT MANSLAUGHTER
## 3: 12      MURDER/MANSLAUGHTER BY NEGLIGENCE
```

```
ralgrep('attempt')
```

```
##      LCR                  LCR.DESC
## 1: 725 MISC/SUICIDE/ATTEMPTED
## 2: 22      RAPE/ATTEMPTED
```

```
ralgrep('theft')
```

```
##      LCR                  LCR.DESC
## 1: 40A          Motor Vehicle Theft
## 2: 35F      Larceny/Theft from Motor Vehicle
## 3: 35G      Larceny/Theft of MV Parts-Accessories
## 4: 56C          Fraud/ID Theft
## 5: 35D      Larceny/Theft from Building
## 6: 35E      Larceny/Theft from Coin-Operated Machine
## 7: 56C      Fraud/ID Theft or Impersonation
## 8: 56H      Fraud/ID Theft (other jurisdiction)
## 9: 73          MV THEFT/OTHER VEHICLES
```

```

## 10:    71          MV THEFT/AUTOMOBILE
## 11: 112I        FRAUD/FALSE PRETENSE / IDENTITY THEFT
## 12: 119I        FRAUD/ALL OTHER / IDENTITY THEFT
## 13: 117I        FRAUD/CREDIT CARD FRAUD / IDENTITY THEFT
## 14:    72          MV THEFT/TRUCK OR BUS
## 15:   79I        MISC/ID THEFT (OTHER JURISDICTION)
## 16: 101I        FORGERY / IDENTITY THEFT
## 17: 111I        FRAUD/WORTHLESS CHECK / IDENTITY THEFT
## 18: 103I        FORGERY/FORGED PRESCRIPTION / IDENTITY THEFT
## 19: 116I        FRAUD/UNLAWFUL CONVERSION / IDENTITY THEFT
## 20: 102I        FORGERY/COUNTERFEITING / IDENTITY THEFT
## 21: 113I        FRAUD/FLIM-FLAM / IDENTITY THEFT
## 22: 118I        FRAUD/DEFRAUDING INNKEEPER / IDENTITY THEFT
## 23: 115I        FRAUD/ALTERING SERIAL NUMBERS/ID THEFT
## 24: 114I        FRAUD/PRICE TAG SWITCHING / IDENTITY THEFT
##           LCR          LCR.DESC

```

```
ralgrep('larc')
```

```

##       LCR          LCR.DESC
## 1: 35H          Larceny/All Other
## 2: 35C          Larceny/Shoplifting
## 3: 35F          Larceny/Theft from Motor Vehicle
## 4: 35G          Larceny/Theft of MV Parts-Accessories
## 5: 35D          Larceny/Theft from Building
## 6: 35A          Larceny/Pocket-Picking
## 7: 35B          Larceny/Purse-Snatching
## 8: 35E          Larceny/Theft from Coin-Operated Machine
## 9: 64D          LARCENY/FROM MOTOR VEHICLES/FELONY (OVER $1,000)
## 10: 64A         LARCENY/FROM MOTOR VEHICLES ($200-$1,000)
## 11: 69B         LARCENY/ALL OTHERS ($50-$199)
## 12: 62A         LARCENY/PURSE-SNATCHING ($200-$1,000)
## 13: 69C         LARCENY/ALL OTHERS (-$50)
## 14: 69A         LARCENY/ALL OTHERS ($200-$1000)
## 15: 66B         LARCENY/BICYCLES ($50-$199)
## 16: 69D         LARCENY/ALL OTHERS ($1000+)
## 17: 67D         LARCENY/FROM BUILDING/FELONY (OVER $1,000)
## 18: 66A         LARCENY/BICYCLES ($200-$1,000)
## 19: 64B         LARCENY/FROM MOTOR VEHICLES ($50-$199)
## 20: 64C         LARCENY/FROM MOTOR VEHICLES (-$50)
## 21: 65B         LARCENY/MOTOR VEHICLE PARTS/ACC ($50-$199)
## 22: 67A         LARCENY/FROM BUILDING ($200-$1,000)
## 23: 63C         LARCENY/SHOPLIFTING (-$50)
## 24: 63A         LARCENY/SHOPLIFTING ($200-$1000)
## 25: 63B         LARCENY/SHOPLIFTING ($50-$199)
## 26: 67B         LARCENY/FROM BUILDING ($50-$199)
## 27: 64AF        LARCENY/FROM MOTOR VEHICLES/FELONY ($200-$1,000)
## 28: 63AF        LARCENY/SHOPLIFTING/FELONY ($200-$1,000)
## 29: 67C         LARCENY/FROM BUILDING (-$50)
## 30: 122         EMBEZZLEMENT/ALL OTHER(LARCENY FROM EMPLOYER)
## 31: 66BF        LARCENY/BICYCLE/FELONY ($50-$199)
## 32: 64CF        LARCENY/FROM MOTOR VEHICLE/FELONY (-$50)
## 33: 69AF        LARCENY/ALL OTHERS/FELONY($200-$1000)
## 34: 69BF        LARCENY/ALL OTHERS/FELONY($50-$199)

```

```

## 35: 65AF    LARCENY/MOTOR VEHICLE PARTS/ACC/FELONY($200-1000)
## 36: 65A      LARCENY/MOTOR VEHICLE PARTS/ACC ($200-$1,000)
## 37: 63CF     LARCENY/SHOPLIFTING/FELONY (-$50)
## 38: 61B      LARCENY/POCKET-PICKING ($50-$199)
## 39: 65C      LARCENY/MOTOR VEHICLE PARTS/ACC (-$50)
## 40: 64BF     LARCENY/FROM MOTOR VEHICLES/FELONY ($50-199)
## 41: 69CF     LARCENY/ALL OTHERS/FELONY(-$50)
## 42: 65D    LARCENY/MOTOR VEHICLE PARTS/ACC/FELONY (OVER $1,000)
## 43: 63D      LARCENY/SHOPLIFTING/FELONY (OVER $1,000)
## 44: 65CF     LARCENY/MOTOR VEHICLE PARTS/ACC/FELONY (-$50)
## 45: 62B      LARCENY/PURSE-SNATCHING ($50-$199)
## 46: 66AF     LARCENY/BICYCLE/FELONY ($200-$1,000)
## 47: 61CF     LARCENY/POCKET-PICKETING/FELONY (-$50)
## 48: 66C      LARCENY/BICYCLES (-$50)
## 49: 63BF     LARCENY/SHOPLIFTING/FELONY ($50-$199)
## 50: 65BF     LARCENY/MOTOR VEHICLE PARTS/ACC/FELONY ($50-$199)
## 51: 66D      LARCENY/BICYCLES/FELONY (OVER $1,000)
## 52: 61A      LARCENY/POCKET-PICKING ($200-$1,000)
## 53: 67AF     LARCENY/FROM BUILDING/FELONY ($200-$1,000)
## 54: 62D      LARCENY/PURSE-SNATCHING/FELONY (OVER $1,000)
## 55: 62C      LARCENY/PURSE-SNATCHING (-$50)
## 56: 68C      LARCENY/FROM COIN OP MACHINES (-$50)
## 57: 61D      LARCENY/POCKET-PICKING/FELONY (OVER $1,000)
## 58: 68B      LARCENY/FROM COIN OP MACHINES ($50-$199)
## 59: 63       LARCENY/SHOPLIFTING
## 60: 61AF     LARCENY/POCKET-PICKING/FELONY ($200-$1,000)
## 61: 69       LARCENY (NO LONGER USED)
## 62: 68A      LARCENY/FROM COIN OP MACHINES ($200-$1000)
## 63: 61C      LARCENY/POCKET-PICKING (-$50)
## 64: 67BF     LARCENY/FROM BUILDING/FELONY ($50-$199)
## 65: 66CF     LARCENY/BICYCLES/FELONY (-$50)
## 66: 68D      LARCENY/FROM COIN OPERATED DEV/FEL(OVER $1,000)
## 67: 68AF     LARCENY/FROM COIN OPERATED DEVICE/FEL($200-$1000)
## 68: 62AF     LARCENY/PURSE-SNATCHING/FELONY ($200-1,000)
## 69: 68CF     LARCENY/FROM COIN OPERATED DEVICE/FELONY(-$50)
## 70: 62CF     LARCENY/PURSE-SNATCHING/FELONY (-$50)
## 71: 60       LARCENY (CIVILIAN USE ONLY)
## 72: 67CF     LARCENY/FROM BUILDING/FELONY (-$50)
## 73: 62BF     LARCENY/PURSE-SNATCHING/FELONY ($50-$199)
## 74: 61BF     LARCENY/POCKET-PICKING/FELONY ($50-$199)
## 75: 64       LARCENY FROM MOTOR VEH (NO LONGER USED)
## 76: 68BF     LARCENY/FROM COIN OPERATED DEVICE/FEL($50-$199)
##          LCR          LCR.DESC

```

```
ralgrep('assault')
```

```

##      LCR          LCR.DESC
## 1: 25E          Assault/Simple
## 2: 25G          Assault/Aggravated
## 3: 25F          Assault/Intimidation-Communicating Threats
## 4: 17C          Sex Offense/Assault with an Object
## 5: 25A          Assault/Firearm
## 6: 25D          Assault/Other Weapon (Aggravated)
## 7: 25C          Assault/Hands (Aggravated)

```

```

## 8: 25B          Assault/Knife
## 9: 25F          Assault/Intimidation-Threats
## 10: 45          ASSAULT SIMPLE (NOT AGGRAVATED)
## 11: 42 ASSAULT WITH KNIFE OR OTHER CUTTING INSTRUMENT
## 12: A43          ASSAULT/LEO - OTHER DANGEROUS WEAPON
## 13: A45          ASSAULT/LEO - SIMPLE, NOT AGGRAVATED
## 14: 44 ASSAULT WITH HANDS,FISTS,FEET,ETC.(AGGRAVATED)
## 15: 41          ASSAULT WITH FIREARM
## 16: 43          ASSAULT WITH OTHER DANGEROUS WEAPON
## 17: A41          ASSAULT/LEO - FIREARM
## 18: A44          ASSAULT/LEO - HANDS,FIST,FEET,ETC-AGGRAVATED
## 19: A42          ASSAULT/LEO - KNIFE OR CUTTING INSTRUMENT

```

From here on, we'll analyze GTA and combined assault incidents. Assign category names to the crimes we're interested in:

```

ral[grep('assault',LCR.DESC,ignore.case = T), Cat := 'ASSAULT']
ral[LCR %in% c('40A','71','72','73'), Cat := 'GTA']
sf[grep('AUTO, GRAND THEFT|GRAND THEFT AUTO',Descript), Cat := 'GTA']
sf[grep('ASSAULT',Descript), Cat := 'ASSAULT']

```

Join the two data sets:

```

crimes <- rbind(
  ral[,(City='Raleigh',DateTime = INC.DATETIME,X,Y,Category=Cat)],
  sf[,(City='San Francisco',DateTime,X,Y,Category=Cat)][!is.na(Category),]
factor.cols <- c('City','Category')
crimes[,factor.cols := lapply(.SD,as.factor),.SDcols = factor.cols,with=F]

```

Only use dates where both cities have data:

```

max.t <- min(crimes[,(d = max(DateTime)),by=City][,d])
min.t <- max(crimes[,(d = min(DateTime)),by=City][,d])
crimes <- crimes[DateTime > min.t & DateTime < max.t]

```

We need to normalize by population to compare the 2 cities. Census population estimate for 2000-2009, downloaded from <https://www.census.gov/popest/data/cities/totals/2009/> on 2/28/2016:

```

pop.2009 <- fread("grep -E '^[0-9]' data/SUB-EST2009-01.csv",skip=3)
setnames(pop.2009,c('Rank','City','State',2009:2000,'Base','Census'))
pop.2014 <- fread("data/PEP_2014_PEPANRNSIP.US12A_with_ann.csv")
setnames(pop.2014,c(make.names(names(pop.2014)[1:8]),'Census','Base',2010:2014))
p.2009 <- melt(pop.2009,1:3,4:13,variable.name='Year',value.name="Population")[grep('San Francisco|Ralei
p.2014 <- melt(pop.2014[grep('San Francisco|Raleigh',GC_RANK.display.label)],c('GC_RANK.display.label')

predpop <- function(p,year) {
  m <- lm(Population ~ Year,p)
  predict(m,2015)
}
population <- rbind(p.2009,p.2014)
population[,Year := as.numeric(as.character(Year))]

```

```

population[,Population := as.numeric(gsub(',',' ',Population))]

#use lm to project 2015 populations
p.2015 <- population[,(Year=2015,Population=predict(lm(Population~Year),newdata=data.frame(Year=2015)))
population <- rbind(population,p.2015)
#population <- population[order(City,Year)]

```

Now to join the crime data with population data

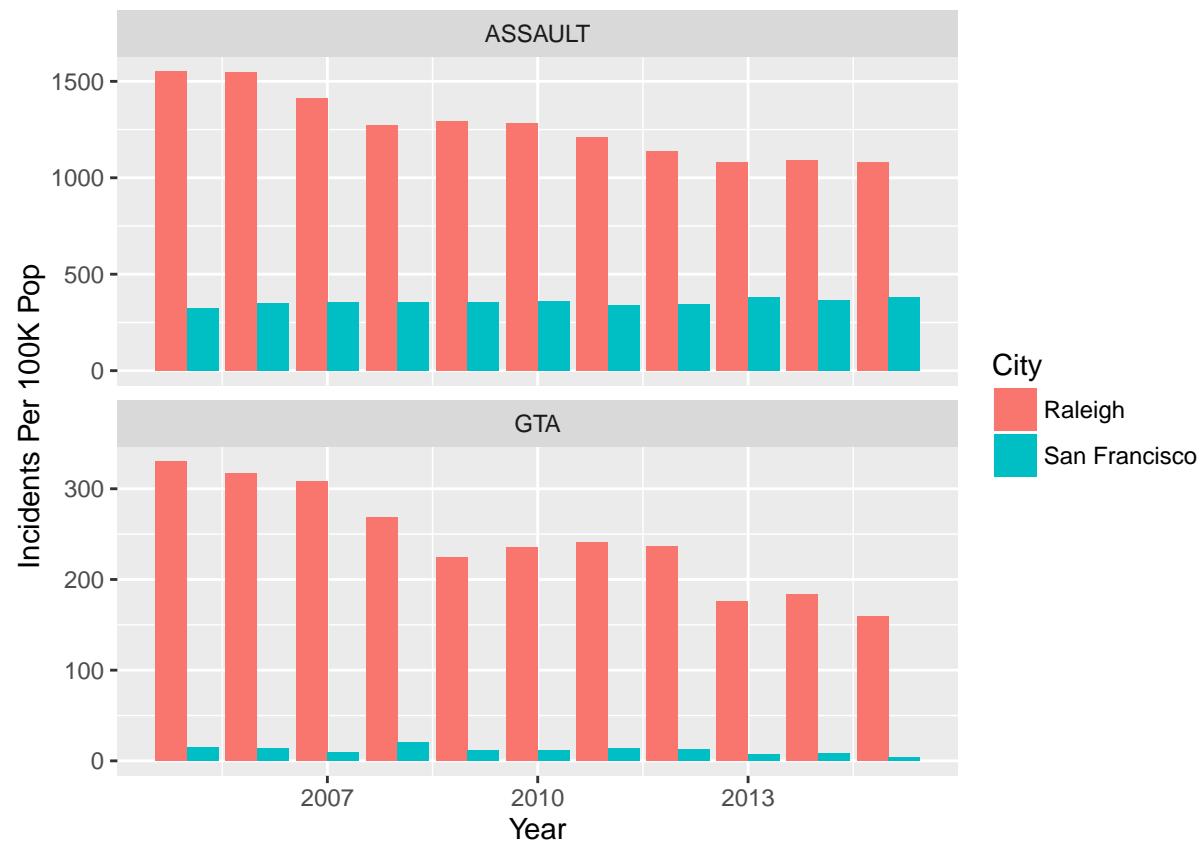
```

crimes[,Year := as.numeric(format(DateTime, '%Y'))]
crimes <- crimes[Year < 2016]
crimes <- merge(crimes,population,by=c('City','Year'))

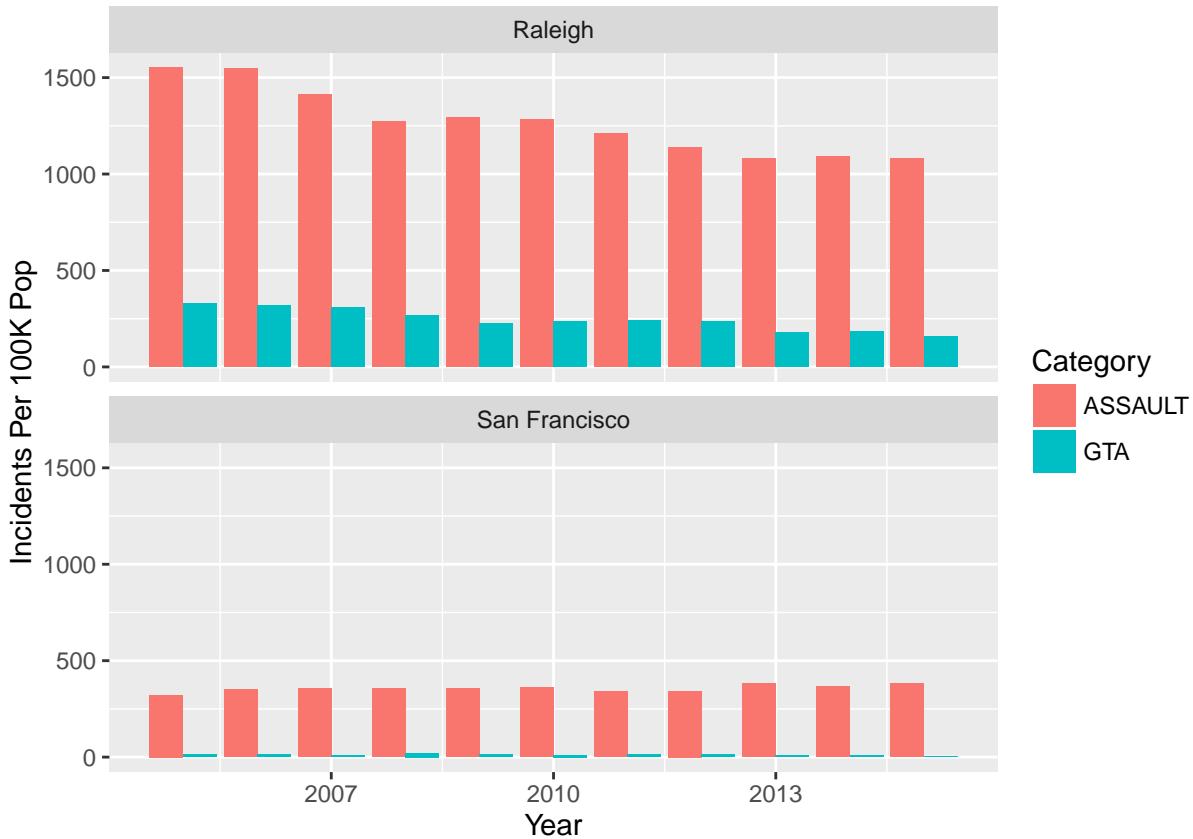
annual.crime <- crimes[,(Per.100K = .N/(Population/100000)),by=.(City,Category,Year)]

ggplot(annual.crime,aes(x=Year,y=Per.100K,fill=City))+geom_bar(stat='identity',position='dodge') + labs

```



```
ggplot(annual.crime,aes(x=Year,y=Per.100K,fill=Category))+geom_bar(stat='identity',position='dodge') +
```



```

library(ggmap)
city.centers <- crimes[!is.na(X) & !is.na(Y), lapply(.SD, function(V) (max(V) - min(V))/2 + min(V)), by=City]
center.sf <- city.centers[City=='San Francisco']
center.ral <- city.centers[City=='Raleigh']
map.sf <- get_map(location = c(center.sf$lon, center.sf$lat), zoom=12)

## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=37.764039,-122.439604&zoom=12&size=600x300
map.ral <- get_map(location = c(center.ral$lon, center.ral$lat), zoom=12)

## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=35.778305,-78.639414&zoom=12&size=600x300

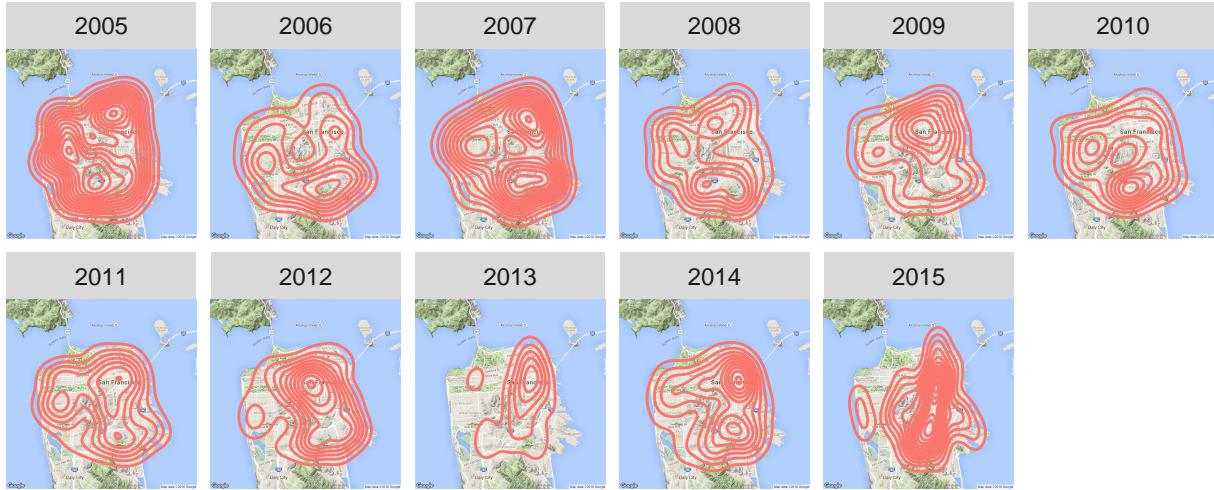
blankout <- theme(axis.line=element_blank(),
                   axis.text.x=element_blank(),
                   axis.text.y=element_blank(),
                   axis.ticks=element_blank(),
                   axis.title.x=element_blank(),
                   axis.title.y=element_blank(),
                   legend.position="none",
                   panel.background=element_blank(),
                   panel.border=element_blank(),
                   panel.grid.major=element_blank(),
                   panel.grid.minor=element_blank(),
                   plot.background=element_blank())
map_progression <- function(m,city,cat) {
  
```

```

print(ggmap(m) + geom_density_2d(aes(x = X, y = Y,color=Category),data=crimes[City==city & Category == "GTA"])
}
map_progression(map.sf,'San Francisco','GTA')

```

### GTA in San Francisco



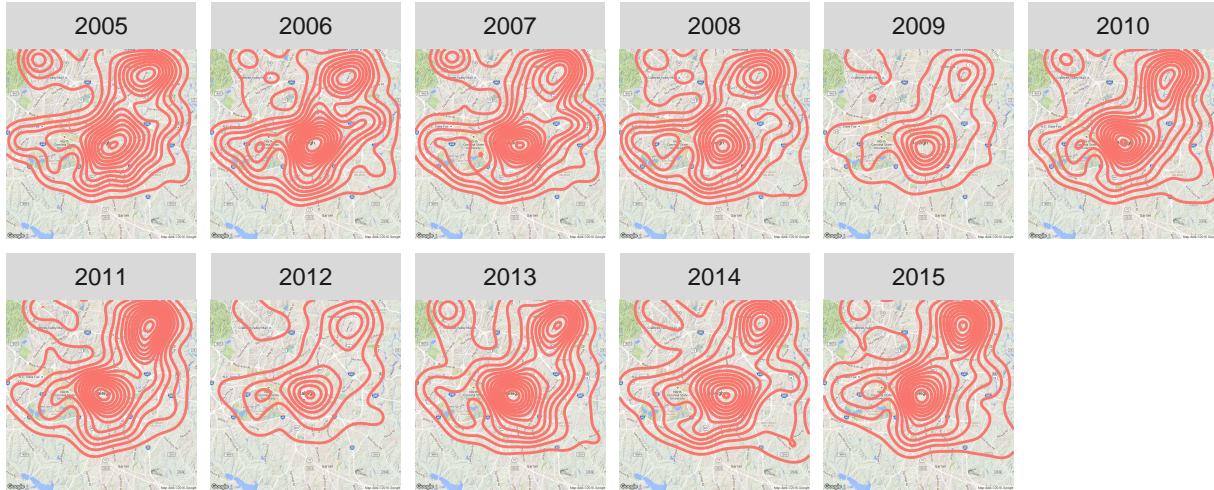
```

map_progression(map.ral,'Raleigh','GTA')

```

## Warning: Removed 1798 rows containing non-finite values (stat\_density2d).

### GTA in Raleigh



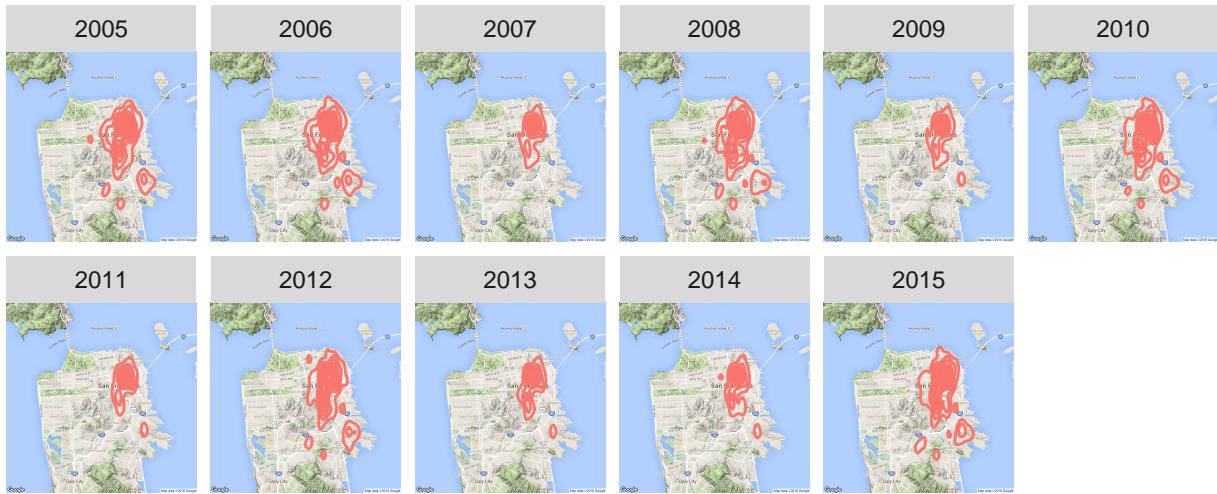
```

map_progression(map.sf,'San Francisco','ASSAULT')

```

## Warning: Removed 2 rows containing non-finite values (stat\_density2d).

## ASSAULT in San Francisco



```
map_progression(map.ral, 'Raleigh', 'ASSAULT')
```

## Warning: Removed 9310 rows containing non-finite values (stat\_density2d).

## ASSAULT in Raleigh

