Assignment 7.2

Visualize the correlation between all variables in a meaningful and clear way of representing. Find out top 3 reasons for having more crime in a city.

What is the difference between co-variance and correlation? Take an example from this dataset and show the differences if any?

```
COBRA_YTD2017<-read.csv('C:/Users/rakesh1/Desktop/COBRA-YTD2017.csv')
require(Amelia)
```

```
## Loading required package: Amelia
## Loading required package: Rcpp
## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.7.5, built: 2018-05-07)
## ## Copyright (C) 2005-2018 James Honaker, Gary King and Matthew Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##
```

```
library(Rcpp)
data<-COBRA_YTD2017
data[4:10,3] <- rep(NA,7)
data[1:5,4] <- NA
data <- data[-c(5,6)]
summary(data)
```

```
##     MI_PRINX          offense_id                rpt_date
##  Min.   :8838438   Min.   :1.608e+08   7/26/2017 :  106
```

```
##  1st Qu.:8904204   1st Qu.:1.711e+08   10/16/2017:  103
##  Median :8910894   Median :1.720e+08   11/1/2017 :  103
##  Mean   :8910851   Mean   :6.523e+08   9/21/2017 :  101
##  3rd Qu.:8917584   3rd Qu.:1.728e+08   11/28/2017:  100
##  Max.   :8924410   Max.   :1.730e+11   (Other)   :26239
##                                        NA's      :    7
##       occur_date       poss_time          beat       apt_offi
ce_prefix
##  11/17/2017:  110   8:00:00 :  526   Min.   :101.0          :
26213
##  10/7/2017 :  106   7:00:00 :  430   1st Qu.:208.0   APT    :
314
##  8/19/2017 :  105   12:00:00:  426   Median :312.0   STE    :
25
##  10/28/2017:  102   10:00:00:  376   Mean   :355.6   ROOM   :
21
##  10/31/2017:   99   9:00:00 :  376   3rd Qu.:505.0   BLDG   :
12
##  (Other)   :26232   16:00:00:  375   Max.   :710.0   UNIT   :
12
##  NA's      :    5   (Other) :24250                   (Other):
162
##  apt_office_num                                      location
##         :22133   1801 HOWELL MILL RD NW                   :  1
42
##  A      :  120   3393 PEACHTREE RD NE @LENOX MALL         :  1
40
##  B      :  108   1275 CAROLINE ST NE @TARGET - CAROLINE   :  1
36
##  1      :   61   3393 PEACHTREE RD NE                     :  1
29
##  2      :   48   835 MARTIN L KING JR DR NW               :  1
08
##  5      :   46   2841 GREENBRIAR PKWY SW @GREENBRIAR MALL:
95
```

```
##  (Other): 4243   (Other)                                    :260
09
##     MinOfucr     MinOfibr_code    dispo_code    MaxOfnum_vict
ims
##  Min.   :110.0   2305   :9024          :22959   Min.   : 0.00
##  1st Qu.:521.0   2404   :2774   10     : 2893   1st Qu.: 1.00
##  Median :640.0   2303   :2486   20     :  632   Median : 1.00
##  Mean   :598.8   2399   :1946   30     :  210   Mean   : 1.16
##  3rd Qu.:660.0   2202   :1802   40     :   36   3rd Qu.: 1.00
##  Max.   :730.0   2308   :1381   60     :   20   Max.   :27.00
##                  (Other):7346   (Other):    9   NA's   :75
##   Shift         Avg.Day        loc_type                   UC2
.Literal
##  Day :6882   Sat    :3713   Min.   : 1.00   LARCENY-FROM VEHI
CLE:9840
##  Eve :9151   Sun    :3569   1st Qu.:13.00   LARCENY-NON VEHIC
LE :6589
##  Morn:7014   Tue    :3542   Median :18.00   AUTO THEFT
:3197
##  Unk :3712   Wed    :3539   Mean   :20.76   BURGLARY-RESIDENC
E  :2635
##              Mon    :3492   3rd Qu.:20.00   AGG ASSAULT
:2024
##              Thu    :3455   Max.   :99.00   ROBBERY-PEDESTRIA
N  :1126
##              (Other):5449   NA's   :3344    (Other)
:1348
##             neighborhood        npu              x
##  Downtown         : 1828   M      : 3077   Min.   :-84.55
##  Midtown          : 1410   E      : 2742   1st Qu.:-84.43
##                   : 1185   B      : 2716   Median :-84.40
##  Old Fourth Ward  :  697   D      : 1281   Mean   :-83.69
##  Lindbergh/Morosgo:  595   V      : 1281   3rd Qu.:-84.37
```

```
##  West End          :  571   T       : 1140   Max.    :  0.00
##  (Other)           :20473   (Other):14522
##         y
##  Min.    : 0.00
##  1st Qu.:33.73
##  Median :33.76
##  Mean    :33.47
##  3rd Qu.:33.79
##  Max.    :33.88
##
```

```
pMiss <- function(x){sum(is.na(x))/length(x)*100}
apply(data,2,pMiss)
```

```
##          MI_PRINX        offense_id          rpt_date
occur_date
##        0.00000000        0.00000000        0.02615942
0.01868530
##         poss_time              beat apt_office_prefix    apt_
office_num
##        0.00000000        0.00000000        0.00000000
0.00000000
##          location          MinOfucr     MinOfibr_code
dispo_code
##        0.00000000        0.00000000        0.00000000
0.00000000
##  MaxOfnum_victims             Shift           Avg.Day
loc_type
##        0.28027953        0.00000000        0.00000000       1
2.49673007
##       UC2.Literal      neighborhood               npu
x
##        0.00000000        0.00000000        0.00000000
0.00000000
##                 y
```

```
##        0.00000000
```

```
apply(data,1,pMiss)
```

```
##     [1] 4.761905 4.761905 4.761905 9.523810 9.523810 4.761905
4.761905
##     [8] 4.761905 4.761905 4.761905 0.000000 4.761905 4.761905
0.000000
##    [15] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [22] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [29] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [36] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [43] 0.000000 4.761905 0.000000 0.000000 0.000000 0.000000
0.000000
##    [50] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000
##    [57] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [64] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000
##    [71] 4.761905 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [78] 4.761905 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##    [85] 0.000000 4.761905 0.000000 4.761905 0.000000 0.000000
0.000000
##    [92] 4.761905 0.000000 0.000000 0.000000 4.761905 0.000000
4.761905
##    [99] 0.000000 0.000000 4.761905 0.000000 0.000000 0.000000
0.000000
##   [106] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
##   [113] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
```

```
##   [120] 4.761905 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

##   [127] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

##   [134] 0.000000 4.761905 4.761905 0.000000 0.000000 0.000000
0.000000

##   [141] 0.000000 4.761905 0.000000 0.000000 0.000000 0.000000
0.000000

##   [148] 0.000000 0.000000 4.761905 0.000000 0.000000 0.000000
0.000000

##   [155] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
4.761905

##   [162] 0.000000 0.000000 0.000000 4.761905 0.000000 4.761905
4.761905

##   [169] 0.000000 4.761905 0.000000 0.000000 4.761905 0.000000
0.000000

##   [176] 4.761905 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

##   [183] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

##   [190] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

##   [197] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

##   [204] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26419] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26426] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26433] 0.000000 0.000000 4.761905 0.000000 0.000000 4.761905
0.000000

## [26440] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26447] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
```

```
## [26454] 0.000000 0.000000 0.000000 0.000000 4.761905 0.000000
0.000000

## [26461] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26468] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
4.761905

## [26475] 0.000000 0.000000 4.761905 0.000000 0.000000 0.000000
0.000000

## [26482] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26489] 4.761905 4.761905 0.000000 0.000000 0.000000 4.761905
0.000000

## [26496] 0.000000 4.761905 0.000000 0.000000 4.761905 0.000000
0.000000

## [26503] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26510] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26517] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26524] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26531] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26538] 4.761905 0.000000 0.000000 4.761905 0.000000 0.000000
0.000000

## [26545] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26552] 0.000000 4.761905 0.000000 0.000000 0.000000 0.000000
0.000000

## [26559] 0.000000 0.000000 0.000000 0.000000 4.761905 0.000000
0.000000

## [26566] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26573] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
```

```
## [26580] 0.000000 4.761905 0.000000 0.000000 0.000000 0.000000
0.000000

## [26587] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26594] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26601] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26608] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26615] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26622] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26629] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26636] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26643] 0.000000 0.000000 0.000000 4.761905 4.761905 0.000000
0.000000

## [26650] 0.000000 0.000000 0.000000 0.000000 4.761905 0.000000
0.000000

## [26657] 0.000000 0.000000 0.000000 4.761905 0.000000 0.000000
0.000000

## [26664] 4.761905 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26671] 0.000000 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26678] 9.523810 4.761905 0.000000 0.000000 4.761905 0.000000
4.761905

## [26685] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26692] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26699] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000
```

```
## [26706] 4.761905 0.000000 0.000000 0.000000 0.000000 4.761905
0.000000

## [26713] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26720] 0.000000 0.000000 0.000000 0.000000 4.761905 0.000000
0.000000

## [26727] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26734] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26741] 0.000000 0.000000 0.000000 4.761905 0.000000 4.761905
0.000000

## [26748] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000

## [26755] 0.000000 0.000000 0.000000 0.000000 0.000000
```

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 3.5.1

## Loading required package: lattice

##

## Attaching package: 'mice'

## The following objects are masked from 'package:base':

##

##     cbind, rbind
```

```
md.pattern(data)
```

```
##       MI_PRINX offense_id poss_time beat apt_office_prefix ap
t_office_num

## 23405        1          1         1    1                 1
1

## 3269         1          1         1    1                 1
1

## 75           1          1         1    1                 1
1
```

```
## 5            1          1         1    1                  1
1

## 3            1          1         1    1                  1
1

## 2            1          1         1    1                  1
1

##             0          0         0    0                  0
0

##       location MinOfucr MinOfibr_code dispo_code Shift Avg.Da
y UC2.Literal

## 23405        1        1             1          1     1
1           1

## 3269         1        1             1          1     1
1           1

## 75           1        1             1          1     1
1           1

## 5            1        1             1          1     1
1           1

## 3            1        1             1          1     1
1           1

## 2            1        1             1          1     1
1           1

##             0        0             0          0     0
0           0

##       neighborhood npu x y occur_date rpt_date MaxOfnum_victi
ms loc_type

## 23405            1   1 1 1          1        1
1        1

## 3269             1   1 1 1          1        1
1        0

## 75               1   1 1 1          1        1
0        0

## 5                1   1 1 1          1        0
1        1

## 3                1   1 1 1          0        1
1        1
```

```
## 2                 1   1 1 1          0        0
1        1
##                   0   0 0 0          5        7
75     3344
##
## 23405    0
## 3269     1
## 75       2
## 5        1
## 3        1
## 2        2
##       3431
```

```
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 3.5.1
## Loading required package: colorspace
## Loading required package: grid
## Loading required package: data.table
## VIM is ready to use.
##  Since version 4.0.0 the GUI is in its own package VIMGUI.
##
##           Please use the package to use the new (and old) GUI
.
## Suggestions and bug-reports can be submitted at: https://gith
ub.com/alexkowa/VIM/issues
##
## Attaching package: 'VIM'
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
aggr_plot <- aggr(data, col=c('navyblue','red'), numbers=TRUE, s
ortVars=TRUE, labels=names(data), cex.axis=.7, gap=3, ylab=c("Hi
stogram of missing data","Pattern"))
```

```
## Warning in plot.aggr(res, ...): not enough horizontal space t
o display
## frequencies
```

```
##
##  Variables sorted by number of missings:
##           Variable        Count
##           loc_type 0.1249673007
##   MaxOfnum_victims 0.0028027953
##           rpt_date 0.0002615942
##         occur_date 0.0001868530
##           MI_PRINX 0.0000000000
##         offense_id 0.0000000000
##          poss_time 0.0000000000
##               beat 0.0000000000
##  apt_office_prefix 0.0000000000
##     apt_office_num 0.0000000000
##           location 0.0000000000
##           MinOfucr 0.0000000000
##      MinOfibr_code 0.0000000000
##         dispo_code 0.0000000000
##              Shift 0.0000000000
##            Avg.Day 0.0000000000
##        UC2.Literal 0.0000000000
##       neighborhood 0.0000000000
##                npu 0.0000000000
##                  x 0.0000000000
##                  y 0.0000000000
marginplot(data[c(1,2)])
```

```r
# All below charts provide the visualization of missing data in
the data set
m <- matrix(data=cbind(rnorm(30, 0), rnorm(30, 2), rnorm(30, 5))
, nrow=30, ncol=3)
apply(m, 1, mean)
```

```
##  [1] 3.6966102 2.5742466 2.7391286 2.1355486 2.0897085 2.2097
172 2.5066403
##  [8] 1.3674533 1.2135926 2.3049017 1.5394682 2.4264711 2.3560
555 1.4429536
## [15] 1.9525326 2.8921570 2.8218232 2.0948454 2.9282604 1.6813
430 2.8007640
## [22] 2.4313354 2.7598386 2.5998863 3.1127215 2.0842223 1.5925
865 0.5778122
## [29] 2.3238416 1.2541749
```

```r
apply(m, 2, function(x) length(x[x<0]))
```

```
## [1] 14  0  0
```

```r
apply(m, 2, function(x) is.matrix(x))
```

```
## [1] FALSE FALSE FALSE
```

```r
apply(m, 2, is.vector)
```

```
## [1] TRUE TRUE TRUE
```

```r
apply(m, 2, function(x) mean(x[x>0]))
```

```
## [1] 0.5386839 1.9773260 4.7891772
```

```r
sapply(1:3, function(x) x^2)
```

```
## [1] 1 4 9
```

```r
lapply(1:3, function(x) x^2)
```

```
## [[1]]
## [1] 1
##
## [[2]]
## [1] 4
##
```

```
## [[3]]
## [1] 9
sapply(1:3, function(x) mean(m[,x]))
## [1] -0.1154391  1.9773260  4.7891772
sapply(1:3, function(x, y) mean(y[,x]), y=m)
## [1] -0.1154391  1.9773260  4.7891772
library(tidyverse)
## -- Attaching packages ---------------------------------------
------ tidyverse 1.2.1 --
## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.1     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0
## -- Conflicts ------------------------------------------------
tidyverse_conflicts() --
## x dplyr::between()   masks data.table::between()
## x tidyr::complete()  masks mice::complete()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()
library(ggmap)
## Warning: package 'ggmap' was built under R version 3.5.1
library(readxl)
library(kableExtra)
## Warning: package 'kableExtra' was built under R version 3.5.1
library(knitr)
str(COBRA_YTD2017)
## 'data.frame':    26759 obs. of  23 variables:
```

```
##  $ MI_PRINX         : int  8924155 8924156 8924157 8924158 89
24159 8924160 8924161 8924162 8924163 8924164 ...

##  $ offense_id       : num  1.74e+08 1.74e+08 1.74e+08 1.74e+0
8 1.74e+08 ...

##  $ rpt_date         : Factor w/ 365 levels "1/1/2017","1/10/2
017",..: 117 117 117 117 117 117 117 117 117 117 ...

##  $ occur_date       : Factor w/ 471 levels "1/1/2008","1/1/20
15",..: 174 145 174 174 176 174 176 176 174 176 ...

##  $ occur_time       : Factor w/ 1355 levels "","0:00:00","0:0
1:00",..: 955 290 883 763 43 940 112 2 2 2 ...

##  $ poss_date        : Factor w/ 412 levels "1/1/2015","1/1/20
17",..: 147 145 147 147 147 147 147 147 147 147 ...

##  $ poss_time        : Factor w/ 1434 levels "","0:00:00","0:0
1:00",..: 32 902 62 68 50 88 121 722 1024 1056 ...

##  $ beat             : int  510 501 303 507 409 612 605 603 60
5 304 ...

##  $ apt_office_prefix: Factor w/ 88 levels "","#8","1","10",..
: 1 1 1 1 1 1 1 1 1 1 ...

##  $ apt_office_num   : Factor w/ 2044 levels "","#5","]","`",.
.: 1 1 1 1 1 1 213 1 1 1372 ...

##  $ location         : Factor w/ 13865 levels ": 565 Main St N
E",..: 9394 1133 10955 7860 5557 1525 8250 9706 9456 455 ...

##  $ MinOfucr         : int  640 640 640 640 640 650 311 640 64
0 531 ...

##  $ MinOfibr_code    : Factor w/ 68 levels "","1101","1101A",.
.: 51 51 51 51 51 50 30 51 51 42 ...

##  $ dispo_code       : Factor w/ 8 levels "","10","20","30",..
: 1 1 1 1 1 1 1 1 1 1 ...

##  $ MaxOfnum_victims : int  2 1 1 1 2 1 1 1 1 1 ...

##  $ Shift            : Factor w/ 4 levels "Day","Eve","Morn",.
.: 3 4 3 2 3 3 3 3 4 3 ...

##  $ Avg.Day          : Factor w/ 8 levels "Fri","Mon","Sat",..
: 3 7 3 3 4 4 4 4 3 4 ...

##  $ loc_type         : int  13 13 18 18 18 18 26 18 13 26 ...

##  $ UC2.Literal      : Factor w/ 11 levels "AGG ASSAULT",..: 6
6 6 6 6 6 10 6 6 4 ...
```

```
##  $ neighborhood     : Factor w/ 239 levels "","Adair Park",..
: 80 117 145 64 3 83 103 164 103 175 ...
##  $ npu              : Factor w/ 26 levels "","A","B","C",..:
14 6 22 14 19 23 23 14 23 22 ...
##  $ x                : num  -84.4 -84.4 -84.4 -84.4 -84.5 ...
##  $ y                : num  33.8 33.8 33.7 33.8 33.7 ...
```

```
COBRA_YTD2017$long <- COBRA_YTD2017$x %>%
  as.numeric()
COBRA_YTD2017$lat <- COBRA_YTD2017$y %>%
  as.numeric()


COBRA_YTD2017$loc_type <- COBRA_YTD2017$UC2.Literal %>%   as.fac
tor()


COBRA_YTD2017$days <- COBRA_YTD2017$Avg.Day %>%
  as.factor()
kable(count(COBRA_YTD2017, loc_type, sort=TRUE), "html", col.nam
es=c("Crime Type", "Frequency")) %>%
kable_styling(bootstrap_options="striped", full_width=FALSE)
```

Crime Type	Frequency
LARCENY-FROM VEHICLE	9840
LARCENY-NON VEHICLE	6589
AUTO THEFT	3197
BURGLARY-RESIDENCE	2635

Crime Type	Frequency
AGG ASSAULT	2024
ROBBERY-PEDESTRIAN	1126
BURGLARY-NONRES	758
RAPE	226
ROBBERY-COMMERCIAL	157
ROBBERY-RESIDENCE	132
HOMICIDE	75

```
COBRA_YTD2017 %>%
  group_by(days, loc_type) %>%
  summarize(freq=n()) %>%
  ggplot(aes(reorder(days, -freq), freq)) +
  geom_bar(aes(fill=loc_type), position="dodge", stat="identity"
, width=0.8, color="black") +
  xlab("Day of Week") +
  ylab("Frequency") +
  labs(fill="Crime Type") +
  ggtitle("Crime by Day of the Week")
```

```
kable
```

```
## function (x, format, digits = getOption("digits"), row.names
= NA,
```

```
##     col.names = NA, align, caption = NULL, format.args = list
(),
##     escape = TRUE, ...)
## {
##     if (missing(format) || is.null(format))
##         format = getOption("knitr.table.format")
##     if (is.null(format))
##         format = if (is.null(pandoc_to()))
##             switch(out_format() %n% "markdown", latex = "late
x",
##                 listings = "latex", sweave = "latex", html =
"html",
##                 markdown = "markdown", rst = "rst", stop("tab
le format not implemented yet!"))
##         else if (isTRUE(opts_knit$get("kable.force.latex")) &
&
##             is_latex_output()) {
##             "latex"
##         }
##         else "pandoc"
##     if (is.function(format))
##         format = format()
##     if (format != "latex" && !missing(align) && length(align)
==
##         1L)
##         align = strsplit(align, "")[[1]]
##     if (!is.null(caption) && !is.na(caption))
##         caption = paste0(create_label("tab:", opts_current$ge
t("label"),
##             latex = (format == "latex")), caption)
##     if (inherits(x, "list")) {
##         if (format == "pandoc" && is_latex_output())
```

```
##             format = "latex"
##         res = lapply(x, kable, format = format, digits = digi
ts,
##             row.names = row.names, col.names = col.names, ali
gn = align,
##             caption = NA, format.args = format.args, escape =
escape,
##             ...)
##         res = unlist(lapply(res, paste, collapse = "\n"))
##         res = if (format == "latex") {
##             kable_latex_caption(res, caption)
##         }
##         else if (format == "html" || (format == "pandoc" && i
s_html_output()))
##             kable_html(matrix(paste0("\n\n", res, "\n\n"), 1)
,
##                 caption = caption, escape = FALSE, table.attr
= "class=\"kable_wrapper\"")
##         else {
##             res = paste(res, collapse = "\n\n")
##             if (format == "pandoc")
##                 kable_pandoc_caption(res, caption)
##             else res
##         }
##         return(structure(res, format = format, class = "knitr
_kable"))
##     }
##     if (!is.matrix(x))
##         x = as.data.frame(x)
##     if (identical(col.names, NA))
##         col.names = colnames(x)
##     m = ncol(x)
```

```
##     isn = if (is.matrix(x))
##         rep(is.numeric(x), m)
##     else sapply(x, is.numeric)
##     if (missing(align) || (format == "latex" && is.null(align
)))
##         align = ifelse(isn, "r", "l")
##     digits = rep(digits, length.out = m)
##     for (j in seq_len(m)) {
##         if (is_numeric(x[, j]))
##             x[, j] = round(x[, j], digits[j])
##     }
##     if (any(isn)) {
##         if (is.matrix(x)) {
##             if (is.table(x) && length(dim(x)) == 2)
##                 class(x) = "matrix"
##             x = format_matrix(x, format.args)
##         }
##         else x[, isn] = format_args(x[, isn], format.args)
##     }
##     if (is.na(row.names))
##         row.names = has_rownames(x)
##     if (!is.null(align))
##         align = rep(align, length.out = m)
##     if (row.names) {
##         x = cbind(` ` = rownames(x), x)
##         if (!is.null(col.names))
##             col.names = c(" ", col.names)
##         if (!is.null(align))
##             align = c("l", align)
##     }
```

```
##     n = nrow(x)
##     x = replace_na(to_character(as.matrix(x)), is.na(x))
##     if (!is.matrix(x))
##         x = matrix(x, nrow = n)
##     x = trimws(x)
##     colnames(x) = col.names
##     if (format != "latex" && length(align) && !all(align %in%
##         c("l", "r", "c")))
##         stop("'align' must be a character vector of possible
values 'l', 'r', and 'c'")
##     attr(x, "align") = align
##     res = do.call(paste("kable", format, sep = "_"), list(x =
x,
##         caption = caption, escape = escape, ...))
##     structure(res, format = format, class = "knitr_kable")
## }
## <bytecode: 0x0000000024a52558>
## <environment: namespace:knitr>
```

#The data provides crime type frequency and crime by da y of the week.#Among the high crime categories, larceny tend to increase on Fridays and Saturdays. while burgla ry residence generally occurred more often during the w eekdays than the weekends. Auto theft were least report ed on Thursdays and increase for the weekends.

```
atlanta_map <- qmap("atlanta",
                    zoom=12,
                    source="stamen",
                    maptype="toner",
                    color="bw")
```

```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?
center=atlanta&zoom=12&size=640x640&scale=2&maptype=terrain&sens
or=false
```

```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=atlanta&sensor=false
## Map from URL : http://tile.stamen.com/toner/12/1086/1638.png
## Map from URL : http://tile.stamen.com/toner/12/1087/1638.png
## Map from URL : http://tile.stamen.com/toner/12/1088/1638.png
## Map from URL : http://tile.stamen.com/toner/12/1089/1638.png
## Map from URL : http://tile.stamen.com/toner/12/1086/1639.png
## Map from URL : http://tile.stamen.com/toner/12/1087/1639.png
## Map from URL : http://tile.stamen.com/toner/12/1088/1639.png
## Map from URL : http://tile.stamen.com/toner/12/1089/1639.png
## Map from URL : http://tile.stamen.com/toner/12/1086/1640.png
## Map from URL : http://tile.stamen.com/toner/12/1087/1640.png
## Map from URL : http://tile.stamen.com/toner/12/1088/1640.png
## Map from URL : http://tile.stamen.com/toner/12/1089/1640.png
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
atlanta_map
## Theme element panel.border missing
## Theme element axis.line.x.bottom missing
## Theme element axis.ticks.x.bottom missing
## Theme element axis.line.x.top missing
## Theme element axis.ticks.x.top missing
## Theme element axis.line.y.left missing
## Theme element axis.ticks.y.left missing
## Theme element axis.line.y.right missing
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
```

```
## Theme element plot.caption missing
```

```
library(dplyr)
library(data.table)
library(ggplot2)
at <- COBRA_YTD2017
str(at)
```

```
## 'data.frame':    26759 obs. of  26 variables:
##  $ MI_PRINX         : int  8924155 8924156 8924157 8924158 8924159 8924160 8924161 8924162 8924163 8924164 ...
##  $ offense_id       : num  1.74e+08 1.74e+08 1.74e+08 1.74e+08 1.74e+08 ...
##  $ rpt_date         : Factor w/ 365 levels "1/1/2017","1/10/2017",..: 117 117 117 117 117 117 117 117 117 117 ...
##  $ occur_date       : Factor w/ 471 levels "1/1/2008","1/1/2015",..: 174 145 174 174 176 174 176 176 174 176 ...
##  $ occur_time       : Factor w/ 1355 levels "","0:00:00","0:01:00",..: 955 290 883 763 43 940 112 2 2 2 ...
##  $ poss_date        : Factor w/ 412 levels "1/1/2015","1/1/2017",..: 147 145 147 147 147 147 147 147 147 147 ...
##  $ poss_time        : Factor w/ 1434 levels "","0:00:00","0:01:00",..: 32 902 62 68 50 88 121 722 1024 1056 ...
##  $ beat             : int  510 501 303 507 409 612 605 603 605 304 ...
##  $ apt_office_prefix: Factor w/ 88 levels "","#8","1","10",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ apt_office_num   : Factor w/ 2044 levels "","#5","]","`",..: 1 1 1 1 1 1 213 1 1 1372 ...
##  $ location         : Factor w/ 13865 levels ": 565 Main St NE",..: 9394 1133 10955 7860 5557 1525 8250 9706 9456 455 ...
##  $ MinOfucr         : int  640 640 640 640 640 650 311 640 640 531 ...
##  $ MinOfibr_code    : Factor w/ 68 levels "","1101","1101A",..: 51 51 51 51 51 50 30 51 51 42 ...
```

```
##  $ dispo_code       : Factor w/ 8 levels "","10","20","30",..
: 1 1 1 1 1 1 1 1 1 1 ...
##  $ MaxOfnum_victims : int  2 1 1 1 2 1 1 1 1 1 ...
##  $ Shift            : Factor w/ 4 levels "Day","Eve","Morn",.
.: 3 4 3 2 3 3 3 3 4 3 ...
##  $ Avg.Day          : Factor w/ 8 levels "Fri","Mon","Sat",..
: 3 7 3 3 4 4 4 4 3 4 ...
##  $ loc_type         : Factor w/ 11 levels "AGG ASSAULT",..: 6
6 6 6 6 6 10 6 6 4 ...
##  $ UC2.Literal      : Factor w/ 11 levels "AGG ASSAULT",..: 6
6 6 6 6 6 10 6 6 4 ...
##  $ neighborhood     : Factor w/ 239 levels "","Adair Park",..
: 80 117 145 64 3 83 103 164 103 175 ...
##  $ npu              : Factor w/ 26 levels "","A","B","C",..:
14 6 22 14 19 23 23 14 23 22 ...
##  $ x                : num  -84.4 -84.4 -84.4 -84.4 -84.5 ...
##  $ y                : num  33.8 33.8 33.7 33.8 33.7 ...
##  $ long             : num  -84.4 -84.4 -84.4 -84.4 -84.5 ...
##  $ lat              : num  33.8 33.8 33.7 33.8 33.7 ...
##  $ days             : Factor w/ 8 levels "Fri","Mon","Sat",..
: 3 7 3 3 4 4 4 4 3 4 ...
```

```
at$MI_PRINX <- at$apt_office_prefix <- at$apt_office_num <- at$l
ocation <- at$dispo_code <- at$loc_type <- at$npu <- NULL
library(chron)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:chron':
##
##     days, hours, minutes, seconds, years
## The following objects are masked from 'package:data.table':
##
```

```
##     hour, isoweek, mday, minute, month, quarter, second, wday
,
##     week, yday, year
## The following object is masked from 'package:base':
##
##     date
at$lon <- at$x
at$lat <- at$y
at$occur_date <- mdy(at$occur_date)
at$rpt_date <- mdy(at$rpt_date)
at$occur_time <- chron(times=at$occur_time)
at$lon <- as.numeric(at$lon)
at$lat <- as.numeric(at$lat)
at$x <- at$y <- NULL
library(xts)
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Attaching package: 'xts'
## The following objects are masked from 'package:dplyr':
##
##     first, last
## The following objects are masked from 'package:data.table':
##
##     first, last
```

```
by_Date <- na.omit(at) %>% group_by(occur_date) %>% summarise(To
tal = n())

tseries <- xts(by_Date$Total, order.by= by_Date$occur_date)

library(highcharter)
```

```
## Warning: package 'highcharter' was built under R version 3.5.
1

## Highcharts (www.highcharts.com) is a Highsoft software produc
t which is

## not free for commercial and Governmental use
```

```
hchart(tseries, name = "Crimes") %>%

  hc_add_theme(hc_theme_darkunica()) %>%

  hc_credits(enabled = TRUE, text = "Sources: Atlanta Police Dep
artment", style = list(fontSize = "12px")) %>%

  hc_title(text = "Time Series of Atlanta Crimes") %>%

  hc_legend(enabled = TRUE)
```

Zoom1m3m6mYTD1yAllFromDec 30, 1916ToDec 31, 2017Time Series of Atlanta CrimesCrimes2016201602550751001 25Sources: Atlanta Police Department

```
hchart
```

```
## function (object, ...)

## {

##     UseMethod("hchart")

## }

## <bytecode: 0x0000000021bb6d30>

## <environment: namespace:highcharter>
```

```
#Graph provides the data spread of the crime during the year

at$dayofWeek <- weekdays(as.Date(at$occur_date))

at$hour <- sub(":.*", "", at$occur_time)

at$hour <- as.numeric(at$hour)

ggplot(aes(x = hour), data = at) + geom_histogram(bins = 24, col
or='white', fill='black') +
```

```
  ggtitle('Histogram of Crime Time')
```

```
## Warning: Removed 11 rows containing non-finite values (stat_b
in).
```

```
#The crime time distribution appears bimodal with peaki
ng around midnight and again at the noon, then again be
tween 6pm and 8pm.
#topCrimes_1 <- topCrimes %>% group_by(`UC2 Literal`,occur_time)
%>%
  #summarise(total = n())
#ggplot(aes(x = occur_time, y = total), data = topCrimes_1) +
  #geom_point(colour="blue", size=1) +
  #geom_smooth(method="loess") +
  #xlab('Hour(24 hour clock)') +
 # ylab('Number of Crimes') +
  #ggtitle('Top Crimes Time of the Day') +
  #facet_wrap(~`UC2 Literal`)


#Downtown and midtown are the most common locations whe
re crimes take place, followed by Old Fourth Ward and W
est End.
topLocations <- subset(at, neighborhood =="Downtown"|neighborhoo
d =="Midtown" | neighborhood=="Old Fourth Ward" | neighborhood==
"West End" | neighborhood=="Vine City" | neighborhood=="North Bu
ckhead")
topLocations <- within(topLocations,  neighborhood <- factor(nei
ghborhood, levels = names(sort(table(neighborhood), decreasing =
T))))
topLocations$days <- ordered(topLocations$days,
                                     levels = c('Monday', 'Tuesday
', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'))
ggplot(data = topLocations, aes(x = days, fill = neighborhood))
+
```

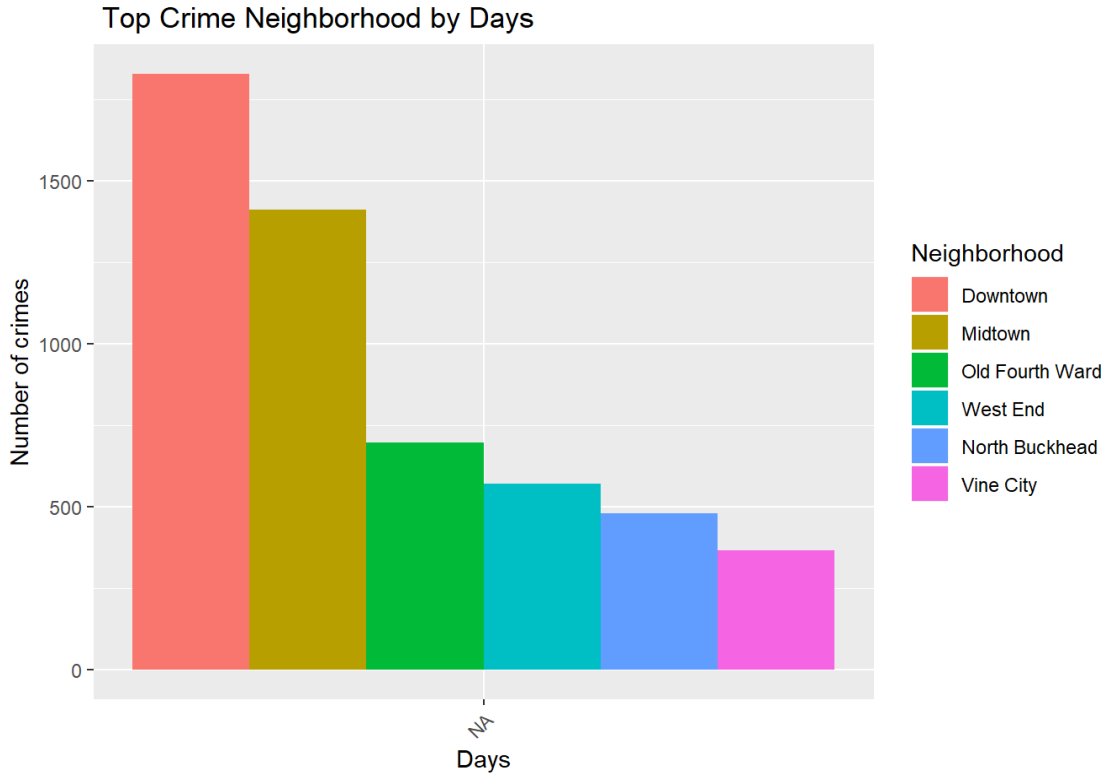
```
  geom_bar(width = 0.9, position = position_dodge()) + ggtitle("
Top Crime Neighborhood by Days") +

  labs(x = "Days", y = "Number of crimes", fill = guide_legend(t
itle = "Neighborhood")) + theme(axis.text.x = element_text(angle
= 45, hjust = 1))
```

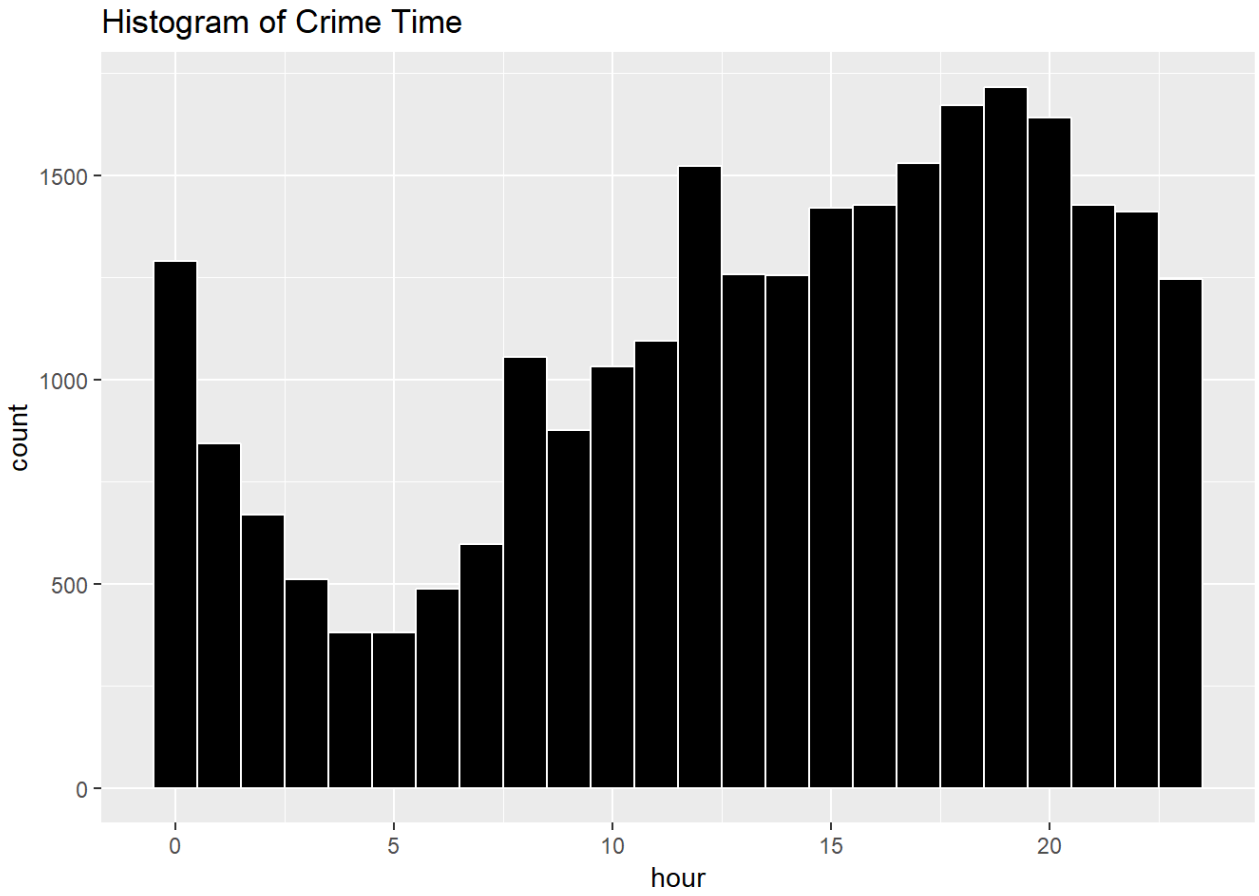
```
#among the high crime categories, larceny tend to incre
ase on Fridays and Saturdays. while burglary residence
generally occurred more often during the weekdays than
the weekends. Auto theft were least reported on Thursda
ys and increase for the weekends.
```

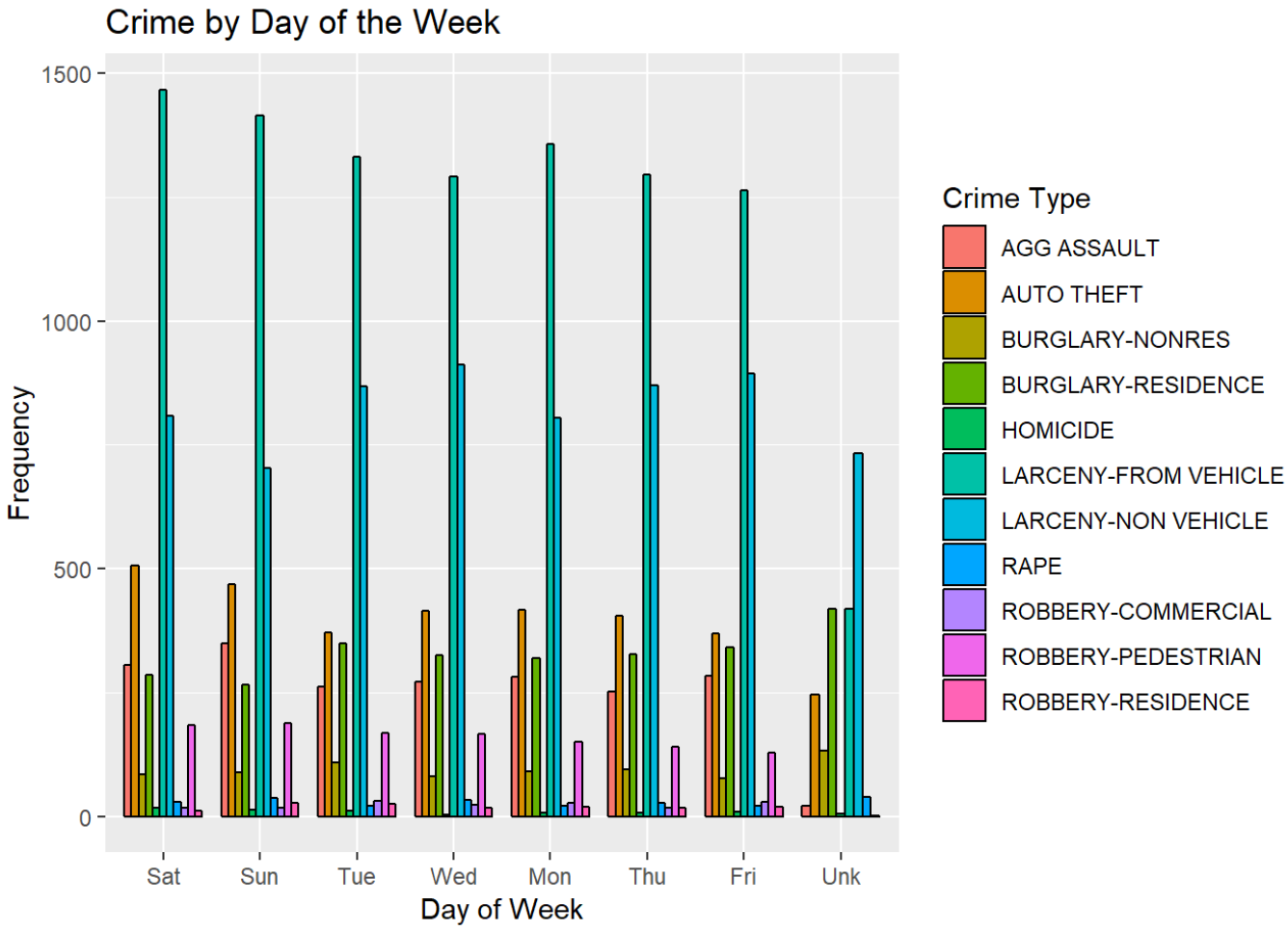
Plots and graphs are attached in the HTML document attached along with the session 13 Assignment```

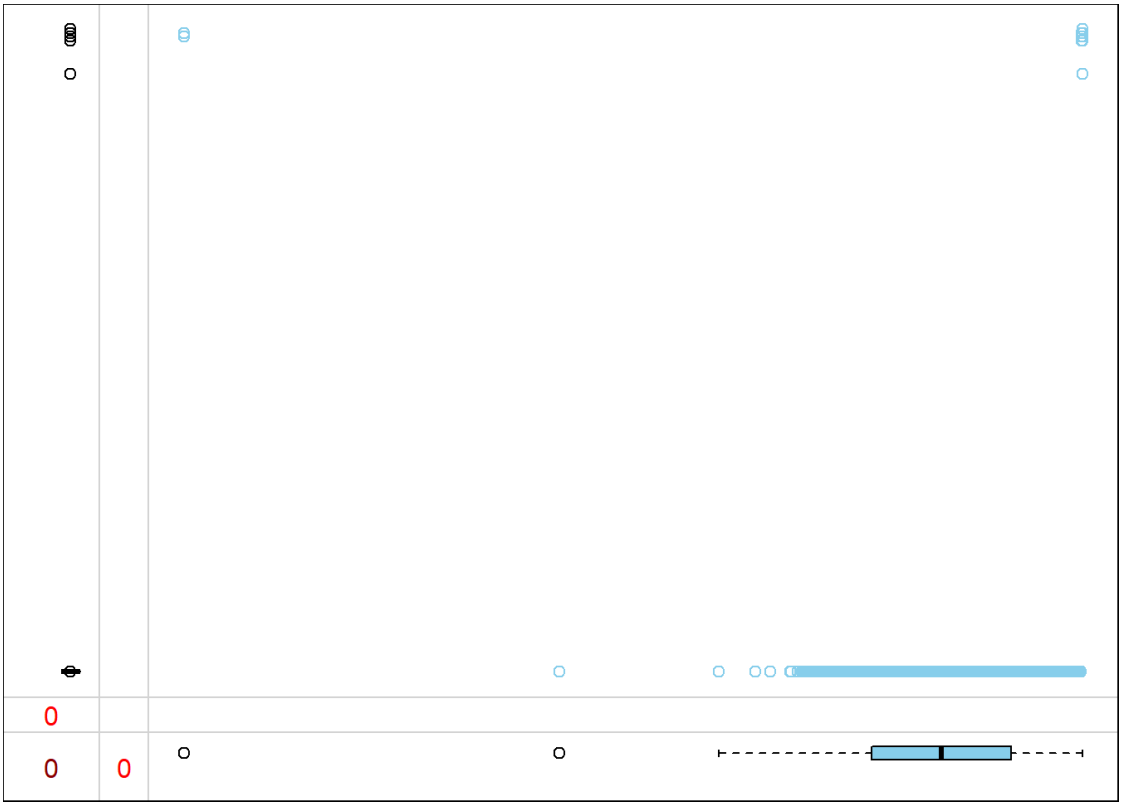
Visualize the correlation between all variables in a meaningful and clear way of representing. Find out top 3 reasons for having more crime in a city.

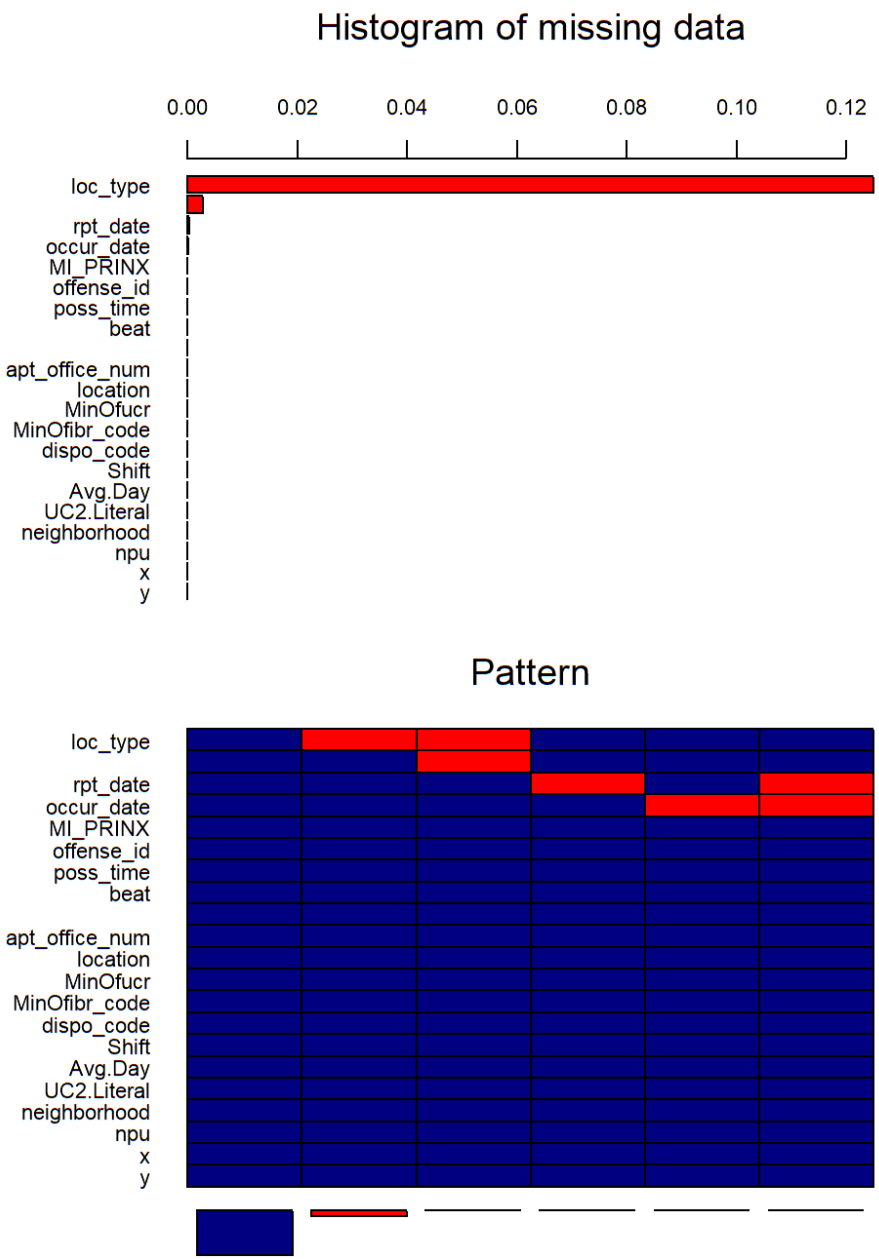
Crime Type	Frequency
LARCENY-FROM VEHICLE	9840
LARCENY-NON VEHICLE	6589
AUTO THEFT	3197
BURGLARY-RESIDENCE	2635
AGG ASSAULT	2024
ROBBERY-PEDESTRIAN	1126
BURGLARY-NONRES	758
RAPE	226
ROBBERY-COMMERCIAL	157

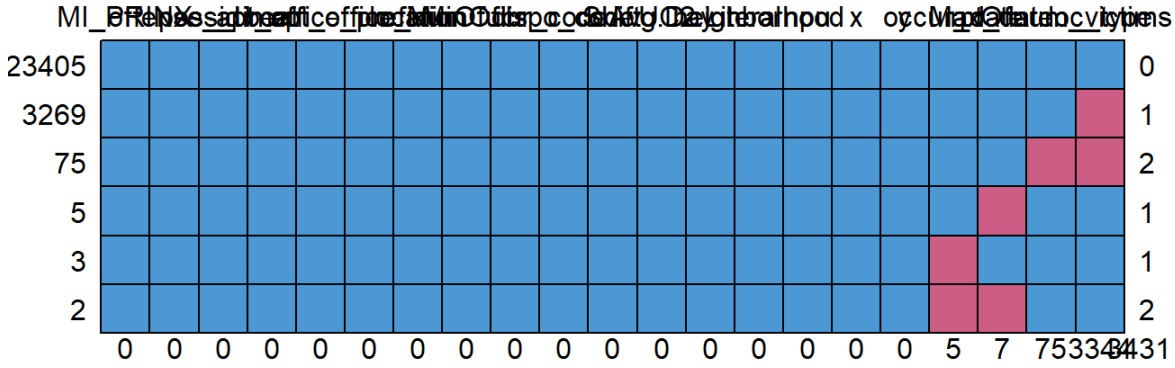



Histogram of Crime Time
1500
1000
500
0
count
0
5
10
15
20
hour


Crime by Day of the Week
1500
1000
500
0
Frequency
Sat
Sun
Tue
Wed
Mon
Thu
Fri
Unk
Day of Week
Crime Type
AGG ASSAULT
AUTO THEFT
BURGLARY-NONRES
BURGLARY-RESIDENCE
HOMICIDE
LARCENY-FROM VEHICLE
LARCENY-NON VEHICLE
RAPE
ROBBERY-COMMERCIAL
ROBBERY-PEDESTRIAN
ROBBERY-RESIDENCE


0
0
0

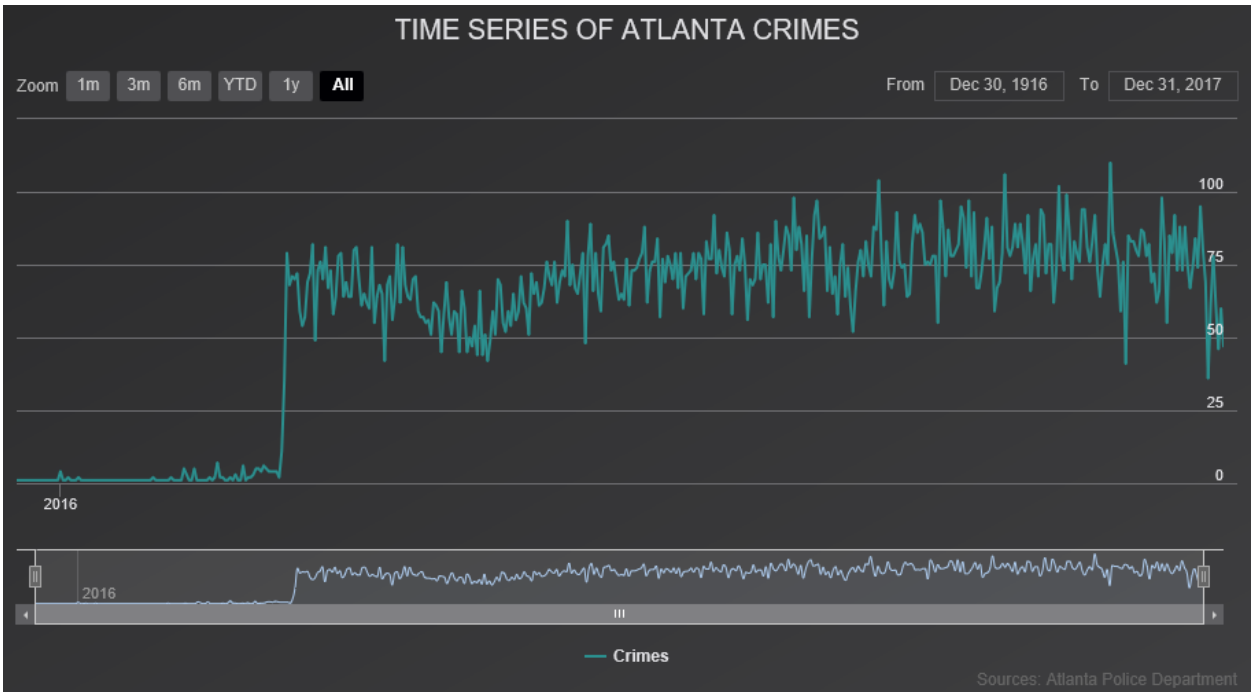

Histogram of missing data
0.00
0.02
0.04
0.06
0.08
0.10
0.12
loc_type
rpt_date
occur_date
MI_PRINX
offense_id
poss_time
beat
apt_office_num
location
MinOfucr
MinOfibr_code
dispo_code
Shift
Avg.Day
UC2.Literal
neighborhood
npu
x
y
Pattern
loc_type
rpt_date
occur_date
MI_PRINX
offense_id
poss_time
beat
apt_office_num
location
MinOfucr
MinOfibr_code
dispo_code
Shift
Avg.Day
UC2.Literal
neighborhood
npu
x
y



Crime Type	Frequency
LARCENY-FROM VEHICLE	9840
LARCENY-NON VEHICLE	6589
AUTO THEFT	3197
BURGLARY-RESIDENCE	2635
AGG ASSAULT	2024
ROBBERY-PEDESTRIAN	1126
BURGLARY-NONRES	758
RAPE	226
ROBBERY-COMMERCIAL	157

graph6.svg

Time series graphs for crime during the period



What is the difference between co-variance and correlation? Take an example from this dataset and show the differences if any?

Covariance and **Correlation** are two mathematical concepts which are quite commonly used in business statistics. Both of these two determine the

relationship and measures the dependency between two random variables. Despite, some similarities between these two mathematical terms, they are different from each other. Correlation is when the change in one item may result in the change in another item.

Correlation is considered as the best tool for for measuring and expressing the quantitative relationship between two variables in formula. On the other hand, covariance is when two items vary together. Read the given article to know the differences between covariance and correlation.

BASIS FOR COMPARISON	COVARIANCE	CORRELATION
Meaning	Covariance is a measure indicating the extent to which two random variables change in tandem.	Correlation is a statistical measure that indicates how strongly two variables are related.
What is it?	Measure of correlation	Scaled version of covariance
Values	Lie between -∞ and +∞	Lie between -1 and +1
Change in scale	Affects covariance	Does not affects correlation
Unit free measure	No	Yes

Similarities

Both measures only linear relationship between two variables, i.e. when the correlation coefficient is zero, covariance is also zero. Further, the two measures are unaffected by the change in location.

Correlation is a special case of covariance which can be obtained when the data is standardized. Now, when it comes to making a choice, which is a better measure of the relationship between two variables*, correlation is preferred over covariance, because it remains unaffected by the change in location and scale, and can also be used to make a comparison between two pairs of variables.*

Take an example from this dataset and show the differences if any?

#Correlation & covariance

#Correlation & covariance

```
cor(COBRA_YTD2017$x,COBRA_YTD2017$y)

cov(COBRA_YTD2017$x,COBRA_YTD2017$y)

cor.test(COBRA_YTD2017$x,COBRA_YTD2017$y)

cor(COBRA_YTD2017$long,COBRA_YTD2017$lat)

cor.test(COBRA_YTD2017$long,COBRA_YTD2017$lat)

cov(COBRA_YTD2017$long,COBRA_YTD2017$lat)

plot(COBRA_YTD2017$x,COBRA_YTD2017$y)

mod=lm(COBRA_YTD2017$long~COBRA_YTD2017$lat)

summary(mod)

predict(mod)
```

pred= predict(mod)

COBRA_YTD2017$predicted=NA

COBRA_YTD2017$predicted=pred

COBRA_YTD2017$error=COBRA_YTD2017$residuals

library(car)

dwt(mod)

plot(COBRA_YTD2017$long,COBRA_YTD2017$lat,abline(COBRA_YTD2017$long~COBRA_YTD2017$lat),col='red')

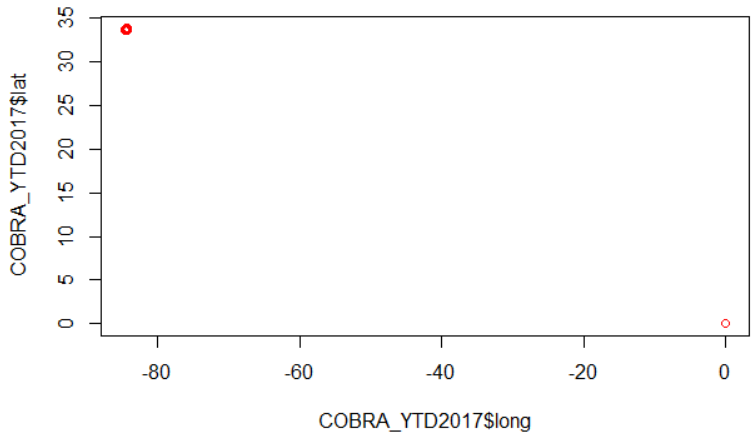
```
[1] -0.9998355
[1] -23.86342

     Pearson's product-moment correlation

data:  COBRA_YTD2017$x and COBRA_YTD2017$y
t = -9017.2, df = 26757, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal
to 0
95 percent confidence interval:
 -0.9998394 -0.9998315
sample estimates:
       cor
-0.9998355

[1] -0.9998355

     Pearson's product-moment correlation

data:  COBRA_YTD2017$long and COBRA_YTD2017$lat
t = -9017.2, df = 26757, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal
to 0
95 percent confidence interval:
 -0.9998394 -0.9998315
```

```
sample estimates:
       cor
-0.9998355

[1] -23.86342

        156         157         158         159         160
-84.42579683 -84.51468279 -84.35395817 -84.32176325 -84.62601522
        161         162         163         164         165
-84.24112598 -84.34355981 -84.61686666 -84.52210662 -84.55457650
        166         167         168         169         170
-84.41107415 -84.52540610 -84.43749498 -84.36698111 -84.53340484
        171         172         173         174         175
-84.31936363 -84.41764811 -84.43677009 -84.36185692 -84.47736369
        176         177         178         179         180
-84.42814646 -84.39302700 -84.11039662 -84.14436626 -84.41507352
        181         182         183         184         185
-84.41789807 -84.39345193 -84.35360822 -84.39540163 -84.39000248
        186         187         188         189         190
-84.31583919 -84.30746551 -84.54732764 -84.49833538 -84.40007589
        191         192         193         194         195
-84.57079894 -84.27072131 -84.38625307 -84.52508115 -84.29791702
        196         197         198         199         200
-84.38047898 -84.51438284 -84.19998248 -84.40202558 -84.27777020
        201         202         203         204         205
-84.52418130 -84.35438310 -84.42687166 -84.39625149 -84.38500327
        206         207         208         209         210
 -0.02197167 -84.47451414 -84.48048819 -84.41507352 -84.29656723
        211         212         213         214         215
-84.37737947 -84.39345193 -84.40407526 -84.39315198 -84.21048082
        216         217         218         219         220
-84.29579235 -84.40952440 -84.43936968 -84.35825749 -84.35383319
        221         222         223         224         225
-84.53747920 -84.53502958 -84.62551530 -84.39052740 -84.49731054
        226         227         228         229         230
-84.42054766 -84.63816330 -84.53415472 -84.39392686 -84.41342378
        231         232         233         234         235
-84.49196138 -84.43989460 -84.21553002 -84.40719976 -84.51833222
        236         237         238         239         240
-84.41532348 -84.31583919 -84.46421576 -84.35043372 -84.41179904
        241         242         243         244         245
-84.38017903 -84.26067290 -84.41802305 -84.40050082 -84.41952282
        246         247         248         249         250
-84.23052765 -84.47738868 -84.49191139 -84.48818698 -84.21835458
        251         252         253         254         255
-84.38622807 -84.55887582 -84.60241894 -84.32358796 -84.28719371
        256         257         258         259         260
-84.27984487 -84.54230343 -84.32371294 -84.39055239 -84.41917287
        261         262         263         264         265
-84.39442678 -84.45599206 -84.38162880 -84.65446073 -84.55635122
        266         267         268         269         270
-84.20898106 -84.60816804 -84.45214267 -84.30629069 -84.36395659
        271         272         273         274         275
-84.30826538 -84.54475305 -84.39625149 -84.56537479 -84.35955728
        276         277         278         279         280
-84.31356455 -84.41579841 -84.46339089 -84.23057765 -84.28134463
```

```
         281          282          283          284          285
-84.18293517 -84.19333353 -84.27127122 -84.42034769 -84.39312698
         286          287          288          289          290
-84.44826828 -84.51308305 -84.41889792 -84.56869927 -84.32543767
         291          292          293          294          295
-84.34570947 -84.29084313 -84.63991302 -84.45231764 -84.34728422
         296          297          298          299          300
-84.40375031 -84.46004142 -84.44054450 -84.41414867 -84.32133832
         301          302          303          304          305
-84.21700479 -84.62551530 -84.50588418 -84.35433311 -84.41237395
         306          307          308          309          310
-84.41452361 -84.25629859 -84.68728055 -84.15311488 -84.42184745
         311          312          313          314          315
-84.26084787 -84.29046819 -84.62551530 -84.64116283 -84.51833222
         316          317          318          319          320
-84.35440810 -84.18495985 -84.39165222 -84.40517508 -84.34943388
         321          322          323          324          325
-84.38160380 -84.27779519 -84.21553002 -84.22970278 -84.40215056
         326          327          328          329          330
-84.68663065 -84.22970278 -84.22560343 -84.54260339 -84.48048819
         331          332          333          334          335
-84.40395028 -84.32476278 -84.31073999 -84.38280361 -84.57717293
         336          337          338          339          340
-84.63108942 -84.45221766 -84.43951966 -84.51833222 -84.46486566
         341          342          343          344          345
-84.48978673 -84.38730290 -84.43127096 -84.41257391 -84.41969779
         346          347          348          349          350
-84.44914314 -84.41184903 -84.53003037 -84.33776073 -84.40410025
         351          352          353          354          355
-84.55367664 -84.55750104 -84.48253787 -84.68920524 -84.56992407
         356          357          358          359          360
-84.44214425 -84.56907421 -84.31683903 -84.36780598 -84.48018824
         361          362          363          364          365
-84.41479856 -84.25817329 -84.44739342 -84.49701058 -84.35080867
         366          367          368          369          370
-84.59796965 -84.49233632 -84.28214451 -84.62551530 -84.17831090
         371          372          373          374          375
-84.51833222 -84.40050082 -84.52305647 -84.41404868 -84.26879661
         376          377          378          379          380
-84.40050082 -84.31738894 -84.62551530 -84.40337537 -84.41784808
         381          382          383          384          385
-84.51183324 -84.61924129 -84.39072736 -84.59854456 -84.45254260
         386          387          388          389          390
-84.44069448 -84.39145225 -84.24954965 -84.37987908 -84.57672300
         391          392          393          394          395
-84.44284414 -84.26667195 -84.40262549 -84.63016456 -84.43989460
         396          397          398          399          400
-84.36748103 -84.62066606 -84.70332801 -84.31468937 -84.55627623
         401          402          403          404          405
-84.44179430 -84.41449861 -84.46544057 -84.38017903 -84.44494380
         406          407          408          409          410
-84.49191139 -84.46079130 -84.42114756 -84.44831827 -84.36680614
         411          412          413          414          415
-84.41889792 -84.55887582 -84.23717660 -84.32443783 -84.16458806
         416          417          418          419          420
-84.39820118 -84.34548451 -84.46381583 -84.39162722 -84.46171616
         421          422          423          424          425
```

```
-84.40977436 -84.33368637 -84.46506563 -84.21700479 -84.53875399
         426          427          428          429          430
-84.31786387 -84.38760286 -84.39100232 -84.42057265 -84.36668116
         431          432          433          434          435
-84.40322539 -84.38347851 -84.27612046 -84.47823855 -84.39622650
         436          437          438          439          440
-84.28366926 -84.10247287 -84.61531691 -84.56557476 -84.29739210
         441          442          443          444          445
-84.27817013 -84.37747946 -84.45769179 -84.40487513 -84.47823855
         446          447          448          449          450
-84.57097391 -84.45919156 -84.18293517 -84.45176773 -84.44466885
         451          452          453          454          455
-84.45334248 -84.55597628 -84.42512193 -84.54177852 -84.39942599
         456          457          458          459          460
-84.37173036 -84.19158380 -84.43314567 -84.39625149 -84.14796569
         461          462          463          464          465
-84.44536874 -84.35265837 -84.49811041 -84.51833222 -84.55877583
         466          467          468          469          470
-84.41917287 -84.59886950 -84.56869927 -84.34913393 -84.22050424
         471          472          473          474          475
-84.39477672 -84.61839142 -84.31231474 -84.62551530 -84.40217556
         476          477          478          479          480
-84.43032111 -84.43841983 -84.53642936 -84.56869927 -84.53775415
         481          482          483          484          485
-84.53357981 -84.35998222 -84.42507194 -84.60919287 -84.25712346
         486          487          488          489          490
-84.39170221 -84.50133490 -84.39032743 -84.64116283 -84.49068658
         491          492          493          494          495
-84.37837931 -84.64983646 -84.43117098 -84.43514535 -84.57097391
         496          497          498          499          500
 -0.02197167 -84.38667800 -84.42582182 -84.38737789 -84.18293517
         501          502          503          504          505
-84.29589234 -84.29046819 -84.28134463 -84.36815593 -84.53457965
         506          507          508          509          510
-84.42189744 -84.30731553 -84.42314725 -84.19020902 -84.38095391
         511          512          513          514          515
-84.40634990 -84.38935258 -84.31528927 -84.35443309 -84.42132253
         516          517          518          519          520
-84.37710451 -84.40357534 -84.41854797 -84.23755155 -84.35443309
         521          522          523          524          525
-84.44256918 -84.27944493 -84.52678088 -84.43289571 -84.38900264
         526          527          528          529          530
-84.48018824 -84.30391607 -84.42814646 -84.33518613 -84.68013168
         531          532          533          534          535
-84.41644830 -84.34403474 -84.40050082 -84.39625149 -84.55427654
         536          537          538          539          540
-84.62536532 -84.35443309 -84.23450203 -84.48341273 -84.39345193
         541          542          543          544          545
-84.62551530 -84.54485303 -84.53080525 -84.62751498 -84.35295833
         546          547          548          549          550
-84.41754813 -84.51833222 -84.41739815 -84.46331590 -84.62579025
         551          552          553          554          555
-84.36568132 -84.54340326 -84.41439863 -84.15229001 -84.14786570
         556          557          558          559          560
-84.40050082 -84.40050082 -84.26562211 -84.63816330 -84.22560343
         561          562          563          564          565
-84.35998222 -84.15881398 -84.48253787 -84.70070343 -84.40050082
```

```
         566          567          568          569          570
-84.41832301 -84.38107889 -84.41384871 -84.63991302 -84.47026481
         571          572          573          574          575
-84.48346272 -84.45249261 -84.21553002 -84.45889160 -84.41917287
         576          577          578          579          580
-84.47493907 -84.62551530 -84.61839142 -84.25734842 -84.45889160
         581          582          583          584          585
-84.35118361 -84.39640147 -84.60261891          990
-84.43074605 -84.63256418 -84.45636700 -84.52885556 -84.11329616
         991          992          993          994          995
-84.41797306 -84.54272837 -84.56177536 -84.63261418 -84.38785282
         996          997          998          999         1000
-84.39625149 -84.36688113 -84.44059449 -84.45641699 -84.17181192
 [ reached getOption("max.print") -- omitted 25759 entries ]


lag Autocorrelation D-W Statistic p-value
   1      0.02809992      1.943799       0
 Alternative hypothesis: rho != 0
```



R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.