ada final project

Yuyao Wang yw3395

4/5/2020

```r
library(dplyr)
```

```
##
## 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
```

```r
library(tidyr)
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

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

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
covid19_confirmed_global=read.csv("~/Desktop/time_series_covid19_confirmed_global.csv",
                                  header=T)

dim(covid19_confirmed_global)
```

```
## [1] 259  78
```

```r
covid19_confirmed_global=covid19_confirmed_global%>%
  select(-"Province.State")%>%
  mutate(cases_sum=rowSums(covid19_confirmed_global[,4:77]))



population_by_country_2020 <- read.csv("~/Desktop/population_by_country_2020.csv",
                                       header=T)

data1=covid19_confirmed_global%>%
  left_join(population_by_country_2020,
            by=c("Country.Region"="Country..or.dependency."))%>%
  select(-c("Lat","Long",))%>%
  group_by(Country.Region)%>%
  mutate(cases_country=sum(cases_sum))
```

```r
write.csv(data1,file="~/Desktop/data1.csv")

# Country (or dependency):
# This column contains different country's name (235 countries)

# Population (2020):
# This columns contains the population of different countries

# Yearly Change:
# This columns contains the population change by yearly

# Net Change:
# This columns contains the net change of the population

# Density (P/Km²):
# The column contains the density of the population

# Land Area (Km²):
# This column contain the land area in terms of kilometer square

# Migrants (net):
# This column represents the migrants of the countries

# Fert. Rate:
# This column represents the fertility or the growth rate of individual countries

# Med. Age:
# This column represents the median age
# (Middle Age or the average age) lifespan of the country

# Urban Pop %:
# This column represents the urban population

# World Share:
# This column represents the population
# contributed to the world's share by individual country
```

```r
data_global=data1[,-c(2:76)]

data_global=data_global[,c(12,1,2:11)]%>%
  select(-c("Net.Change","Land.Area..Km.."))%>%
  rename(Population=Population..2020.)%>%
  distinct()%>%
  mutate(Fert..Rate=as.double(Fert..Rate),
         Urban.Pop..=as.double(Urban.Pop..),
         World.Share=as.double(World.Share),
         Yearly.Change=as.double(Yearly.Change),
         Med..Age=as.double(Med..Age))%>%
  drop_na()%>%
  rename(cases=cases_country,
         Density=Density..P.Km..,
         Popchange=Yearly.Change,
         Country=Country.Region,
         Fert=Fert..Rate,
         MedAge=Med..Age,
         Migrant=Migrants..net.,
         Urban=Urban.Pop..,
         WorldShare=World.Share)%>%
  mutate(log_cases=log(cases))%>%
  drop_na()


data_global=data_global%>%
  mutate(log_casespop=log(cases)/log(Population))
dim(data_global)
```

```
## [1] 158  12
```

```r
write.csv(data_global,file="~/Desktop/data_global.csv")
```

```r
set.seed(0)
index=sample(1:158,10)

data_global=read.csv("~/Desktop/data_global.csv")

data_train=data_global[-index,]
newdata=data_global[index,][,-1]
newdata
```

```
##           cases          Country Population Popchange Density Migrant Fert
## 142   3731.2342          Uruguay    3473730        46      20   -3000   10
## 68    8733.0000             Iraq   40222493       138      93    7834   27
## 129     61.9722         Suriname     586632        76       4   -1000   14
## 43     186.1035      El Salvador    6486205        55     313  -40539   11
## 14     326.4568         Barbados     287375        33     668     -79    6
## 51  593808.0351           France   65273511        40     119   36527    9
## 85   37672.5000         Malaysia   32365999       103      99   50000   10
## 21   61016.0747           Brazil  212559417        67      25   21200    7
## 106    158.9555 Papua New Guinea    8947024       130      20    -800   26
```

```
## 74    3344.5100           Jordan   10203134        83     115   10220   18
##     MedAge Urban WorldShare log_cases log_casespop
## 142     22    78          5  8.224494    0.5460883
## 68       7    57         42  9.074864    0.5182694
## 129     15    49          2  4.126686    0.3106940
## 43      14    57          9  5.226303    0.3331999
## 14      26    19          1  5.788298    0.4605385
## 51      28    65         54 13.294311    0.7388152
## 85      16    61         35 10.536686    0.6093169
## 21      19    71         71 11.018893    0.5746569
## 106      8     4         12  5.068624    0.3166538
## 74      10    73         14  8.115075    0.5028487
```

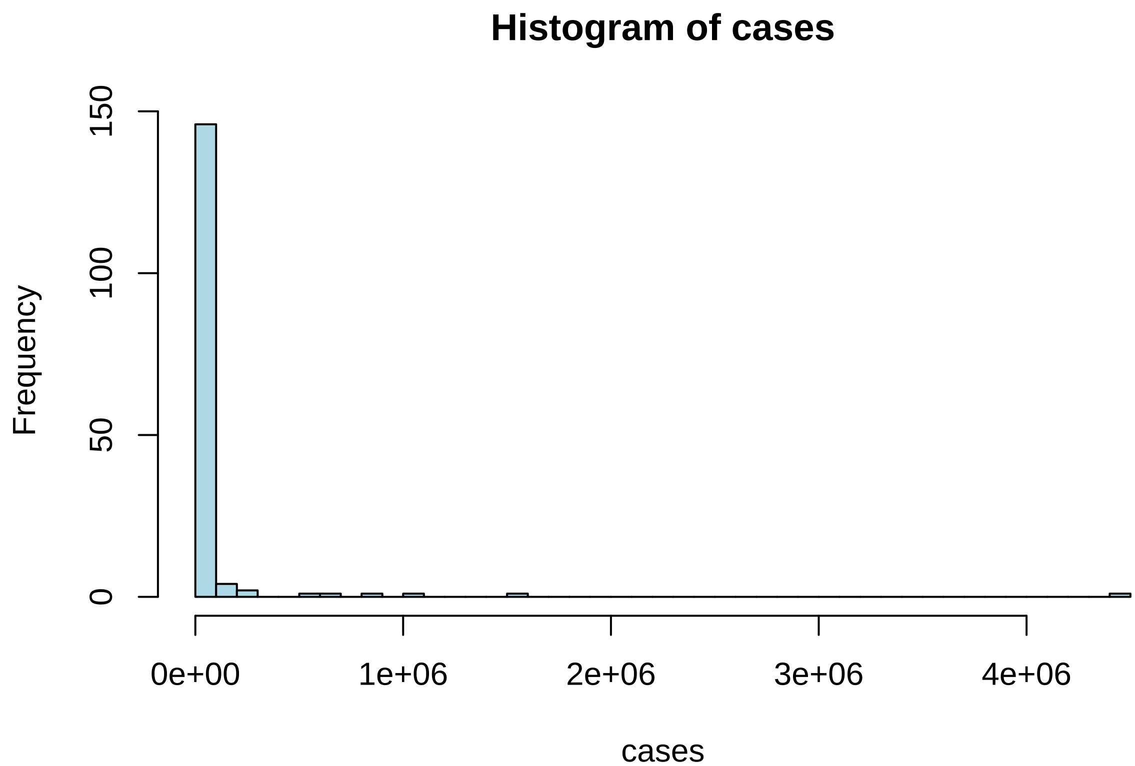
```r
head(data_global)
```

```
##   X     cases             Country Population Popchange Density Migrant Fert
## 1 1 2081.0000         Afghanistan   38928346       139      60  -62920   36
## 2 2 3092.1683             Albania    2877797         4     105  -14000    6
## 3 3 7833.6596             Algeria   43851044       125      18  -10000   21
## 4 4   89.8739              Angola   32866272       170      26    6413   45
## 5 5   30.2036 Antigua and Barbuda      97929        73     223       0   10
## 6 6 9917.3833           Argentina   45195774        79      17    4800   13
##   MedAge Urban WorldShare log_cases log_casespop
## 1      4    13         41  7.640604    0.4371747
## 2     22    47          5  8.036628    0.5403670
## 3     15    57         43  8.966185    0.5095492
## 4      3    51         35  4.498408    0.2599040
## 5     20    14          1  3.407961    0.2965508
## 6     18    75         44  9.202044    0.5220570
```

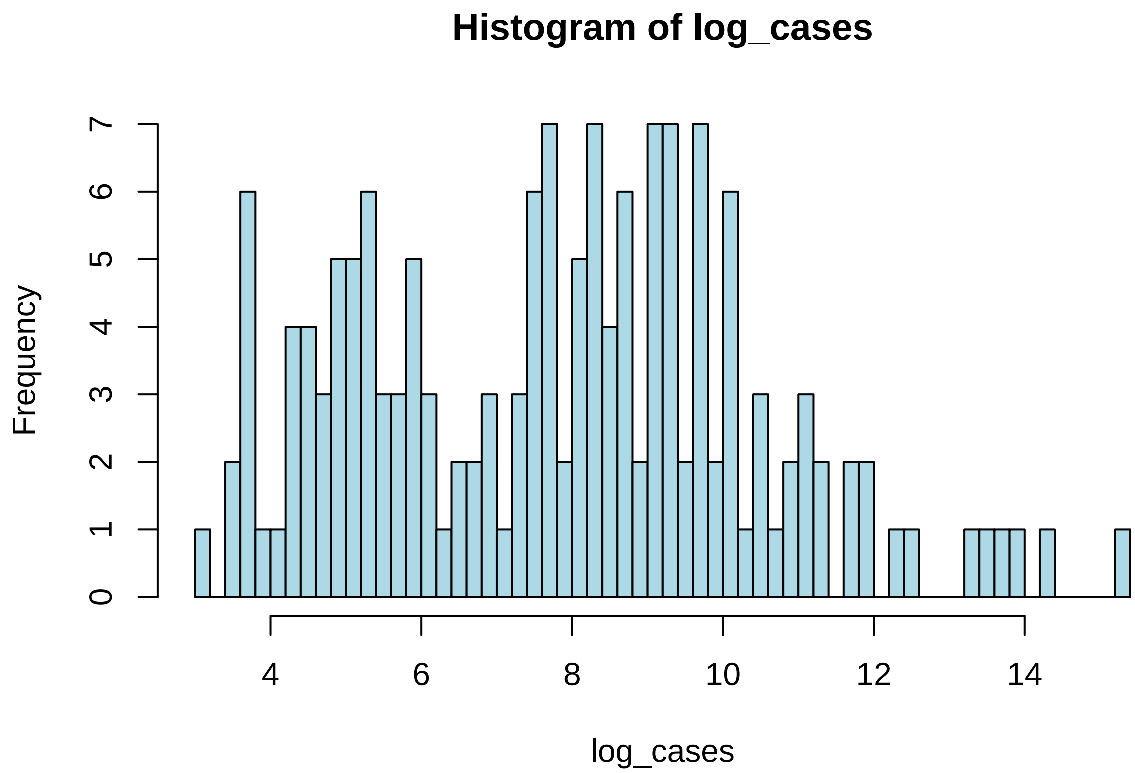
```r
names(data_global)
```

```
##  [1] "X"            "cases"        "Country"      "Population"   "Popchange"
##  [6] "Density"      "Migrant"      "Fert"         "MedAge"       "Urban"
## [11] "WorldShare"   "log_cases"    "log_casespop"
```

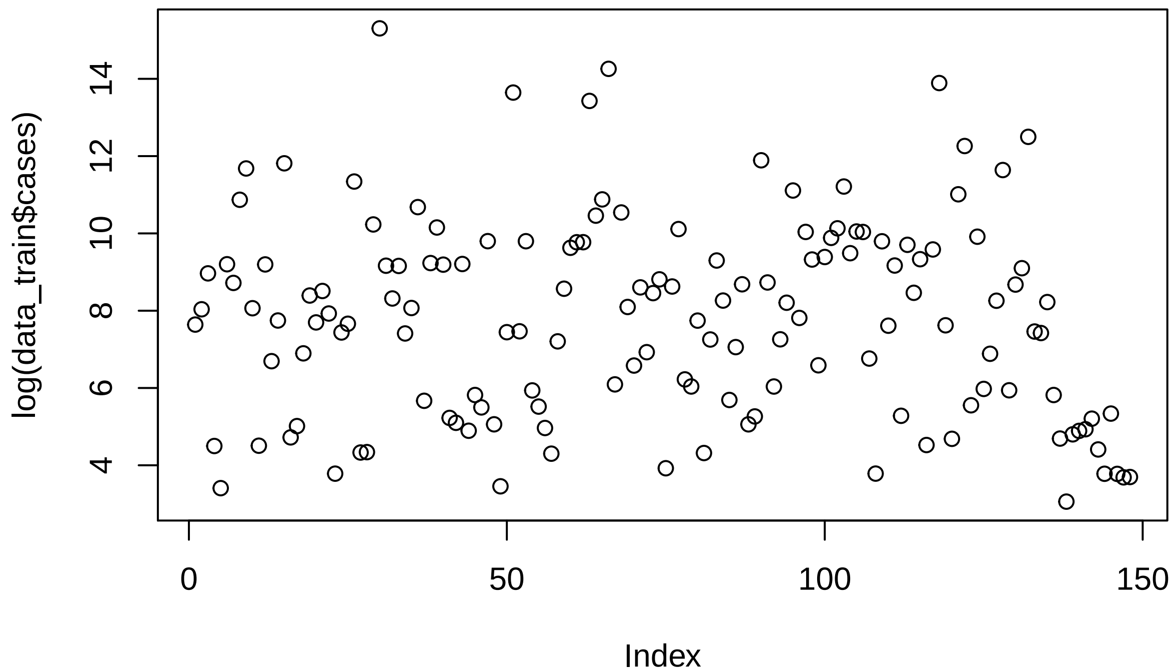
```r
hist(data_global$cases,xlab = "cases", main="Histogram of cases",breaks=50,col="light blue")
```




```
hist(log(data_global$cases),xlab = "log_cases", main="Histogram of log_cases",breaks=50,col="light blue
```




```
plot(log(data_train$cases))
```

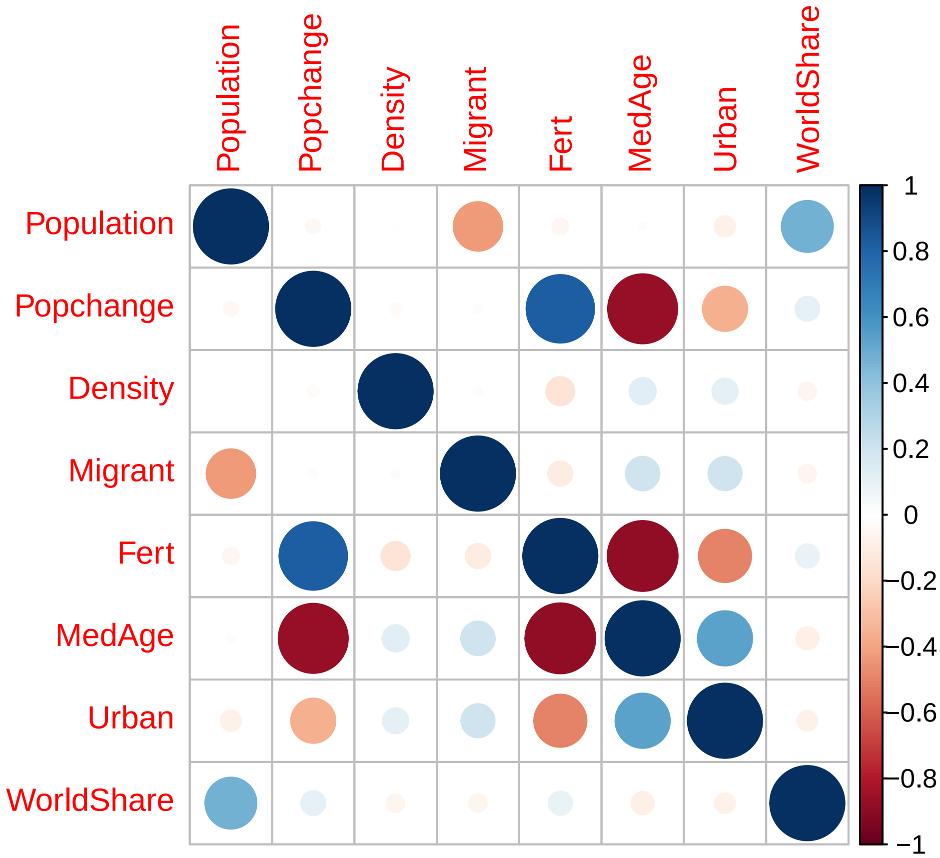


#EDA correlation matrix between continuous variables

```
myvars <- c("Population","Popchange","Density","Migrant",
             "Fert","MedAge","Urban","WorldShare")
data_global2 <-data_global[myvars]
data_global2.cor=cor(data_global2)
data_global2.cor
```

```
##             Population   Popchange      Density     Migrant        Fert
## Population  1.000000000 -0.03597347  0.008614403 -0.42843813 -0.04645276
## Popchange  -0.035973473  1.00000000 -0.024937388 -0.01583095  0.82862994
## Density     0.008614403 -0.02493739  1.000000000 -0.01545121 -0.14418058
## Migrant    -0.428438126 -0.01583095 -0.015451214  1.00000000 -0.10768481
## Fert       -0.046452763  0.82862994 -0.144180578 -0.10768481  1.00000000
## MedAge      0.010985879 -0.86698339  0.126431823  0.20471956 -0.88712830
## Urban      -0.076279936 -0.35551833  0.115546428  0.20011585 -0.49480444
## WorldShare  0.472831594  0.10529316 -0.055289933 -0.05644882  0.09871369
##                  MedAge       Urban  WorldShare
## Population  0.01098588 -0.07627994  0.47283159
## Popchange  -0.86698339 -0.35551833  0.10529316
## Density     0.12643182  0.11554643 -0.05528993
## Migrant     0.20471956  0.20011585 -0.05644882
## Fert       -0.88712830 -0.49480444  0.09871369
## MedAge      1.00000000  0.53445134 -0.08983236
## Urban       0.53445134  1.00000000 -0.07201397
## WorldShare -0.08983236 -0.07201397  1.00000000
```

```r
corrplot(data_global2.cor)
```



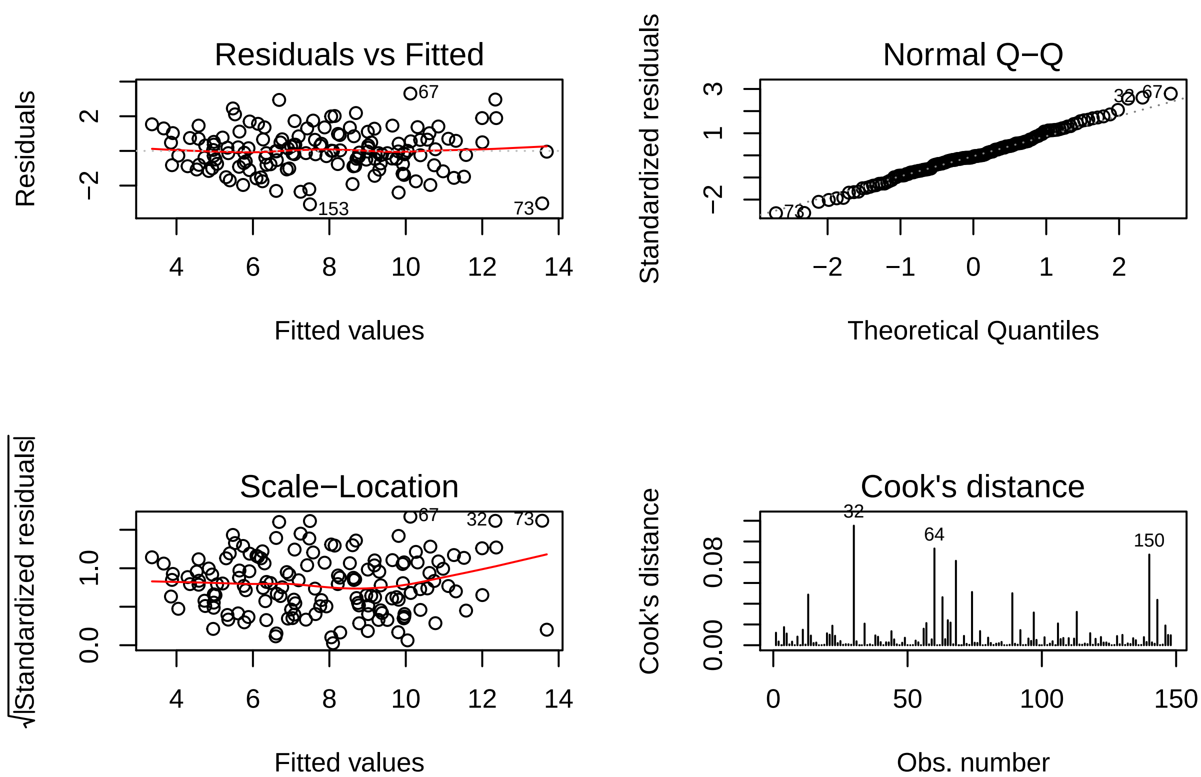

```r
# Since Country is the state with larger scale.
# We decided to drop the Country variable since it has too many levels.
```

```r
#after log transformation, the normality is better than before
m.full=lm(log(cases)~log(Population)+Popchange+log(Density)+
            Migrant+Fert+MedAge+Urban+
          WorldShare,data=data_train)

summary(m.full)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + log(Density) +
##     Migrant + Fert + MedAge + Urban + WorldShare, data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0825 -0.7634 -0.0710  0.6785  3.3087
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.087e+01  2.081e+00  -5.226 6.22e-07 ***
```

```
## log(Population)  8.028e-01  1.378e-01   5.825 3.80e-08 ***
## Popchange         1.304e-02  4.681e-03   2.787  0.00607 **
## log(Density)      1.110e-01  7.769e-02   1.428  0.15549
## Migrant           1.695e-06  9.632e-07   1.760  0.08067 .
## Fert             -2.100e-02  1.869e-02  -1.124  0.26308
## MedAge            2.152e-01  3.393e-02   6.342 2.97e-09 ***
## Urban             2.118e-02  6.714e-03   3.155  0.00197 **
## WorldShare       -4.092e-03  1.171e-02  -0.349  0.72735
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.218 on 139 degrees of freedom
## Multiple R-squared:  0.788,  Adjusted R-squared:  0.7758
## F-statistic: 64.59 on 8 and 139 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(m.full, which = 1:4)
```




```r
# Adjusted R-squared:  0.7781
# p-value: < 2.2e-16
```

```r
m.reduced1=lm(log(cases)~log(Population)+Popchange+
                log(Density)+Fert+MedAge+Urban,data=data_train)
anova(m.reduced1,m.full)
```

```
## Analysis of Variance Table
##
## Model 1: log(cases) ~ log(Population) + Popchange + log(Density) + Fert +
##     MedAge + Urban
## Model 2: log(cases) ~ log(Population) + Popchange + log(Density) + Migrant +
##     Fert + MedAge + Urban + WorldShare
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    141 210.84
## 2    139 206.25  2     4.598 1.5494  0.216
```

```
# after dropping migrants and wordshare, p-value is 0.4887,
# thus it is ok to drop it.

summary(m.reduced1)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + log(Density) +
##     Fert + MedAge + Urban, data = data_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.09968 -0.83284 -0.08497  0.76399  3.14961
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -10.617723   1.267596  -8.376 4.95e-14 ***
## log(Population)   0.745220   0.058399  12.761  < 2e-16 ***
## Popchange         0.015623   0.004459   3.504 0.000615 ***
## log(Density)      0.088486   0.076403   1.158 0.248762
## Fert             -0.016850   0.018442  -0.914 0.362444
## MedAge            0.237460   0.031596   7.515 5.98e-12 ***
## Urban             0.021132   0.006685   3.161 0.001923 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.223 on 141 degrees of freedom
## Multiple R-squared:  0.7833, Adjusted R-squared:  0.7741
## F-statistic: 84.95 on 6 and 141 DF,  p-value: < 2.2e-16
```

```
# now we have 6 predictors to complete our inference and prediction

# Adjusted R-squared: 0.7741

# m.reduced1 is ok.
```

```
m.reduced2=lm(log(cases)~log(Population)+Popchange+MedAge
               +Urban,data=data_train)
anova(m.reduced2,m.reduced1)
```

```
## Analysis of Variance Table
##
## Model 1: log(cases) ~ log(Population) + Popchange + MedAge + Urban
```

```
## Model 2: log(cases) ~ log(Population) + Popchange + log(Density) + Fert +
##     MedAge + Urban
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    143 214.70
## 2    141 210.84  2     3.856 1.2893 0.2787
```

```
# However, because p-value here is 1.414e-08 from ANOVA F-test,
# so there is strong evidence of a difference that m.reduced1 is ok.
# thus we finally decided not to drop urban factor.

# p-value is 0.2787, thus m.reduced2 is ok.

# thus this is our final model

cor(cbind(log(data_train$Population),data_train$Popchange,

          data_train$MedAge,data_train$Urban))
```

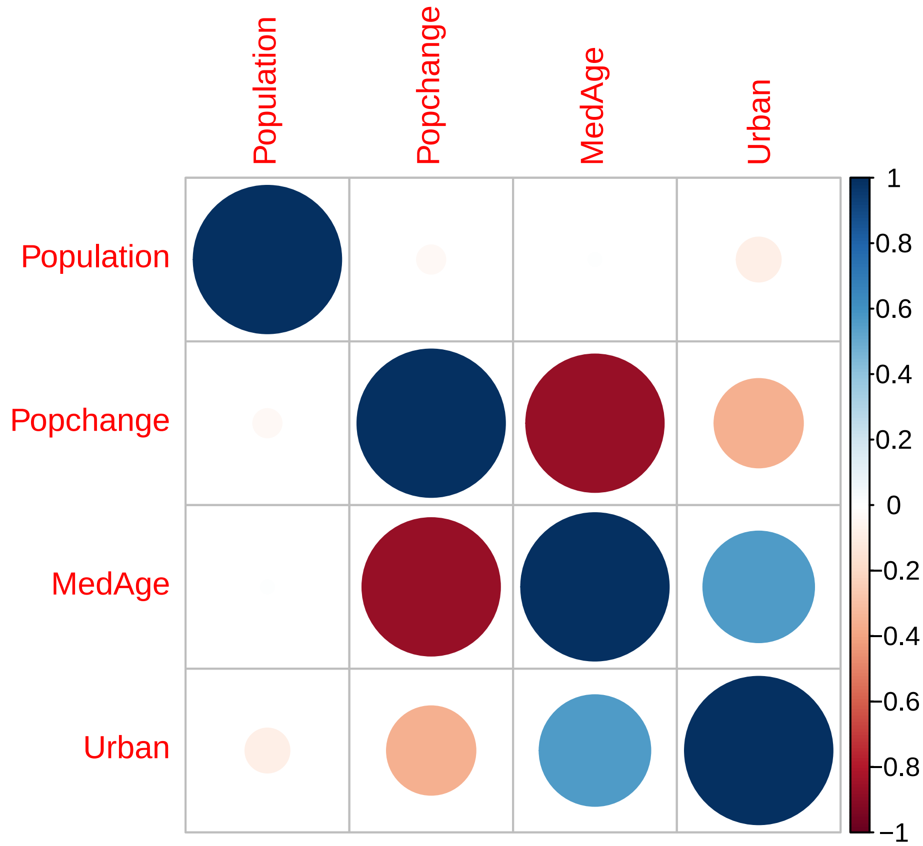
```
##             [,1]       [,2]       [,3]        [,4]
## [1,]  1.00000000  0.1393194 -0.1272541 -0.07587069
## [2,]  0.13931940  1.0000000 -0.8693872 -0.35918191
## [3,] -0.12725408 -0.8693872  1.0000000  0.56316609
## [4,] -0.07587069 -0.3591819  0.5631661  1.00000000
```

```
# corr between Popchange, and MedAge is -0.8693872.
```

```
myvars2 <- c("Population","Popchange",
             "MedAge","Urban")

data_train3 <-data_train[myvars2]
data_train3.cor=cor(data_train3)
data_train3.cor
```

```
##              Population   Popchange       MedAge       Urban
## Population  1.000000000 -0.03664143  0.007799532 -0.08885572
## Popchange  -0.036641431  1.00000000 -0.869387203 -0.35918191
## MedAge      0.007799532 -0.86938720  1.000000000  0.56316609
## Urban      -0.088855724 -0.35918191  0.563166085  1.00000000
```

```
corrplot(data_train3.cor)
```




```r
# also, we consider the migrant_level
# but we find that it is not very related to the model construction

data_train$migrant_level=ifelse(data_train$Migrant<=0,"out","in")
data_train$migrant_level=as.factor(data_train$migrant_level)

m.reduced3=lm(log(cases)~log(Population)+Popchange+
              MedAge+Urban+migrant_level,data=data_train)

anova(m.reduced2,m.reduced3)
```

```
## Analysis of Variance Table
##
## Model 1: log(cases) ~ log(Population) + Popchange + MedAge + Urban
## Model 2: log(cases) ~ log(Population) + Popchange + MedAge + Urban + migrant_level
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    143 214.70
## 2    142 214.65  1  0.048857 0.0323 0.8576
```

```r
# p-value:  0.8576, m.reduced2 is ok

# thus it is our final model.
```

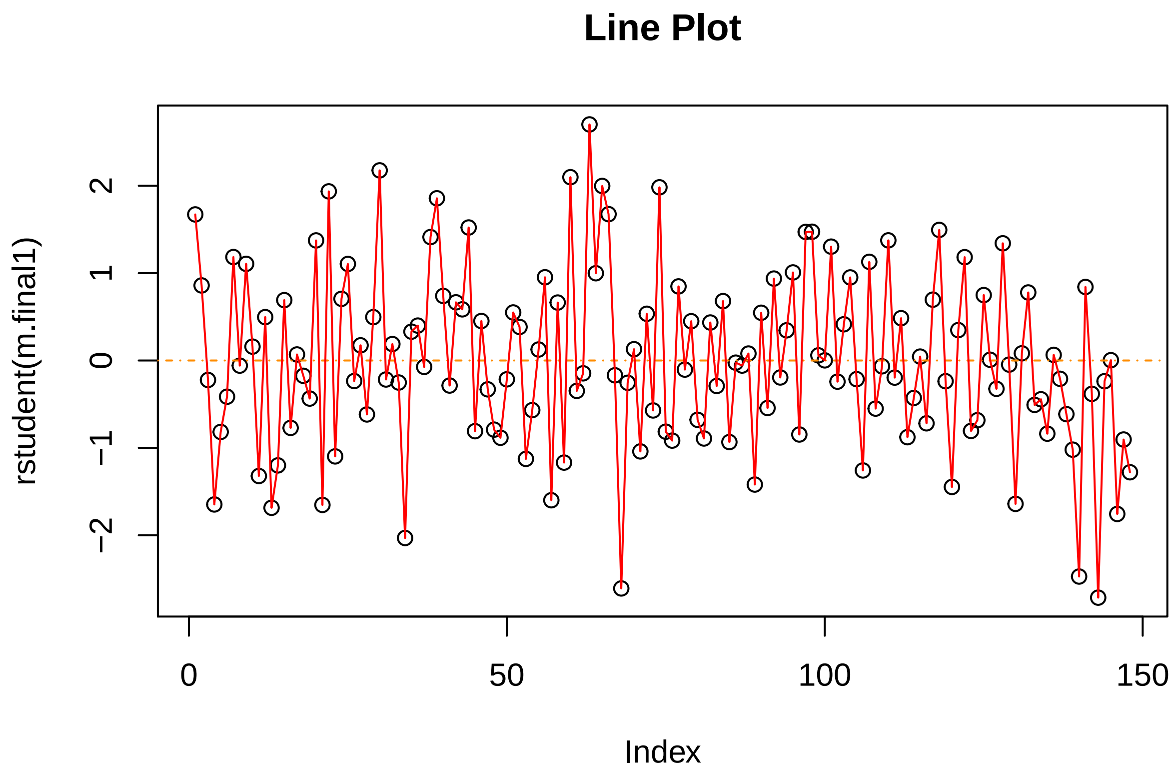
```r
m.final1=lm(log(cases)~log(Population)+Popchange+MedAge
               +Urban,data=data_train)
summary(m.final1)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + MedAge +
##     Urban, data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2281 -0.8179 -0.0759  0.8212  3.2011
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -10.742816   1.167969  -9.198    4e-16 ***
## log(Population)   0.743256   0.058030  12.808  < 2e-16 ***
## Popchange         0.015054   0.004297   3.503 0.000613 ***
## MedAge            0.259579   0.026786   9.691  < 2e-16 ***
## Urban             0.019754   0.006441   3.067 0.002586 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.225 on 143 degrees of freedom
## Multiple R-squared:  0.7793, Adjusted R-squared:  0.7732
## F-statistic: 126.3 on 4 and 143 DF,  p-value: < 2.2e-16
```

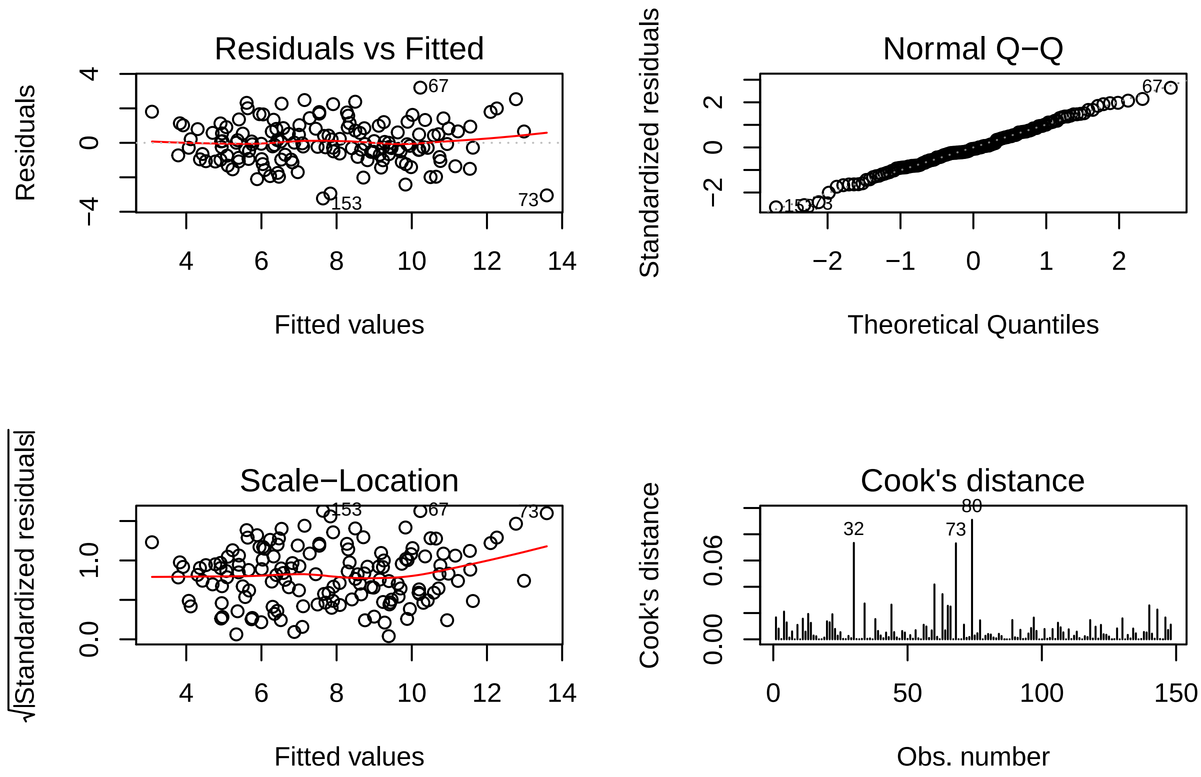
```r
# Adjusted R-squared: 0.7732
# p-value: < 2.2e-16
```

#model diagnostics

```r
#line plot of the studentized deleted residuals

plot(rstudent(m.final1),main="Line Plot")
abline(h=0,lty=10,col="dark orange")
lines(rstudent(m.final1),col=2)
```




```r
par(mfrow=c(2,2))
plot(m.final1, which = 1:4,sub.caption = "Final Model Diagnostic Plots")
```




```r
# 1. pretty close to 0, good
# 2. looks normal
# 3. pretty random points
# 4. only three influencial less than 10% , it is ok
```

```r
m.full<- lm(log(cases)~log(Population)+Popchange+log(Density)+
              Migrant+Fert+MedAge+Urban+
            WorldShare,data=data_train)


m0<- lm(log(cases)~1,data=data_train)

# this time we try to use stepwise backward method


step(m.full,scope=m0,direction=c("backward"))
```

```
## Start:  AIC=67.11
## log(cases) ~ log(Population) + Popchange + log(Density) + Migrant +
##     Fert + MedAge + Urban + WorldShare
##
##                   Df Sum of Sq    RSS    AIC
## - WorldShare       1     0.181 206.43 65.245
## - Fert             1     1.874 208.12 66.453
## <none>                         206.25 67.115
```

```
## - log(Density)     1     3.026 209.27  67.271
## - Migrant          1     4.594 210.84  68.375
## - Popchange        1    11.523 217.77  73.161
## - Urban            1    14.765 221.01  75.348
## - log(Population)  1    50.339 256.58  97.437
## - MedAge           1    59.681 265.93 102.729
##
## Step:  AIC=65.24
## log(cases) ~ log(Population) + Popchange + log(Density) + Migrant +
##     Fert + MedAge + Urban
##
##                   Df Sum of Sq    RSS     AIC
## - Fert             1     1.738 208.17  64.486
## <none>                         206.43  65.245
## - log(Density)     1     2.875 209.30  65.292
## - Migrant          1     4.417 210.84  66.378
## - Popchange        1    11.586 218.01  71.327
## - Urban            1    15.416 221.84  73.904
## - MedAge           1    60.963 267.39 101.541
## - log(Population)  1   247.908 454.33 180.000
##
## Step:  AIC=64.49
## log(cases) ~ log(Population) + Popchange + log(Density) + Migrant +
##     MedAge + Urban
##
##                   Df Sum of Sq    RSS     AIC
## <none>                         208.17  64.486
## - log(Density)     1     3.626 211.79  65.042
## - Migrant          1     3.927 212.09  65.252
## - Popchange        1    10.208 218.37  69.571
## - Urban            1    16.972 225.14  74.086
## - MedAge           1    95.555 303.72 118.397
## - log(Population)  1   246.672 454.84 178.164
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + log(Density) +
##     Migrant + MedAge + Urban, data = data_train)
##
## Coefficients:
##     (Intercept)  log(Population)        Popchange     log(Density)
##      -1.080e+01        7.511e-01        1.199e-02        1.189e-01
##         Migrant           MedAge            Urban
##       1.537e-06        2.347e-01        2.236e-02
```

```
m1<-lm(formula = log(cases) ~ log(Population) + Popchange +  log(Density) + Migrant +
    MedAge + Urban, data = data_train)
summary(m1)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + log(Density) +
##     Migrant + MedAge + Urban, data = data_train)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2527 -0.7954 -0.0473  0.7227  3.3381
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.080e+01  1.186e+00  -9.105 7.55e-16 ***
## log(Population)  7.511e-01  5.811e-02  12.926  < 2e-16 ***
## Popchange        1.199e-02  4.559e-03   2.630 0.009500 **
## log(Density)     1.189e-01  7.588e-02   1.567 0.119317
## Migrant          1.537e-06  9.422e-07   1.631 0.105135
## MedAge           2.347e-01  2.917e-02   8.045 3.20e-13 ***
## Urban            2.236e-02  6.595e-03   3.391 0.000905 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.215 on 141 degrees of freedom
## Multiple R-squared:  0.7861, Adjusted R-squared:  0.777
## F-statistic: 86.34 on 6 and 141 DF,  p-value: < 2.2e-16
```

```r
anova(m.final1,m1)
```

```
## Analysis of Variance Table
##
## Model 1: log(cases) ~ log(Population) + Popchange + MedAge + Urban
## Model 2: log(cases) ~ log(Population) + Popchange + log(Density) + Migrant +
##     MedAge + Urban
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    143 214.70
## 2    141 208.16  2    6.5347 2.2131 0.1131
```

```r
# p-value is 0.1201, thus m.final1 is ok
```

```r
# interaction plot
m.interact<- lm(log(cases) ~ log(Population) + Popchange + MedAge + Urban
                +log(Population)*Popchange+log(Population)*MedAge
                  +log(Population)* Urban + Popchange*MedAge+
                  Popchange* Urban+MedAge*Urban,data=data_train)

summary(m.interact)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + MedAge +
##     Urban + log(Population) * Popchange + log(Population) * MedAge +
##     log(Population) * Urban + Popchange * MedAge + Popchange *
##     Urban + MedAge * Urban, data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3654 -0.7151 -0.0758  0.7422  2.7824
```

```
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 1.0063546  7.8393142   0.128  0.89804
## log(Population)             0.1121438  0.5143479   0.218  0.82773
## Popchange                  -0.0655181  0.0491842  -1.332  0.18504
## MedAge                     -0.1916795  0.2871079  -0.668  0.50550
## Urban                       0.0484295  0.0751428   0.645  0.52033
## log(Population):Popchange   0.0042859  0.0031661   1.354  0.17807
## log(Population):MedAge      0.0229382  0.0186108   1.233  0.21987
## log(Population):Urban      -0.0024976  0.0036022  -0.693  0.48927
## Popchange:MedAge            0.0008491  0.0003112   2.729  0.00719 **
## Popchange:Urban            -0.0000209  0.0002328  -0.090  0.92860
## MedAge:Urban                0.0005434  0.0013026   0.417  0.67719
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.208 on 137 degrees of freedom
## Multiple R-squared:  0.7946, Adjusted R-squared:  0.7796
## F-statistic: 53.01 on 10 and 137 DF,  p-value: < 2.2e-16
```

```
step(m.interact,scope=m0,direction=c("backward"))
```

```
## Start:  AIC=66.43
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + log(Population) *
##     Popchange + log(Population) * MedAge + log(Population) *
##     Urban + Popchange * MedAge + Popchange * Urban + MedAge *
##     Urban
##
##                             Df Sum of Sq    RSS    AIC
## - Popchange:Urban            1    0.0118 199.83 64.435
## - MedAge:Urban               1    0.2539 200.07 64.615
## - log(Population):Urban      1    0.7011 200.52 64.945
## - log(Population):MedAge     1    2.2156 202.03 66.059
## - log(Population):Popchange  1    2.6727 202.49 66.393
## <none>                                  199.81 66.427
## - Popchange:MedAge           1   10.8599 210.68 72.259
##
## Step:  AIC=64.44
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + log(Population):Popchange +
##     log(Population):MedAge + log(Population):Urban + Popchange:MedAge +
##     MedAge:Urban
##
##                             Df Sum of Sq    RSS    AIC
## - log(Population):Urban      1    0.6895 200.52 62.945
## - MedAge:Urban               1    1.0018 200.83 63.175
## - log(Population):MedAge     1    2.2516 202.08 64.094
## - log(Population):Popchange  1    2.6860 202.51 64.411
## <none>                                  199.83 64.435
## - Popchange:MedAge           1   12.0924 211.92 71.131
##
## Step:  AIC=62.95
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + log(Population):Popchange +
##     log(Population):MedAge + Popchange:MedAge + MedAge:Urban
```

```
##
##                              Df Sum of Sq    RSS    AIC
## - MedAge:Urban                1    1.3083 201.82 61.908
## - log(Population):MedAge      1    1.5627 202.08 62.094
## - log(Population):Popchange   1    2.0486 202.56 62.450
## <none>                                    200.52 62.945
## - Popchange:MedAge            1   11.9708 212.49 69.527
##
## Step:  AIC=61.91
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + log(Population):Popchange +
##     log(Population):MedAge + Popchange:MedAge
##
##                              Df Sum of Sq    RSS    AIC
## - log(Population):MedAge      1    1.6504 203.47 61.113
## - log(Population):Popchange   1    2.3409 204.17 61.614
## <none>                                    201.82 61.908
## - Urban                       1    8.3433 210.17 65.903
## - Popchange:MedAge            1   10.8020 212.63 67.624
##
## Step:  AIC=61.11
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + log(Population):Popchange +
##     Popchange:MedAge
##
##                              Df Sum of Sq    RSS    AIC
## - log(Population):Popchange   1    0.6908 204.17 59.615
## <none>                                    203.47 61.113
## - Urban                       1    8.0089 211.48 64.827
## - Popchange:MedAge            1   10.7290 214.20 66.718
##
## Step:  AIC=59.61
## log(cases) ~ log(Population) + Popchange + MedAge + Urban + Popchange:MedAge
##
##                    Df Sum of Sq    RSS     AIC
## <none>                          204.17  59.615
## - Urban             1     7.346 211.51  62.846
## - Popchange:MedAge  1    10.534 214.70  65.060
## - log(Population)   1   256.555 460.72 178.066
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + MedAge +
##     Urban + Popchange:MedAge, data = data_train)
##
## Coefficients:
##      (Intercept)   log(Population)         Popchange            MedAge
##       -9.7867167         0.7697621         0.0035745         0.2049520
##            Urban  Popchange:MedAge
##        0.0148287         0.0007072
```

```r
m.final2=lm(formula = log(cases) ~ log(Population) + Popchange + MedAge +
    Urban + Popchange:MedAge, data = data_train)
summary(m.final2)
```

```
##
```

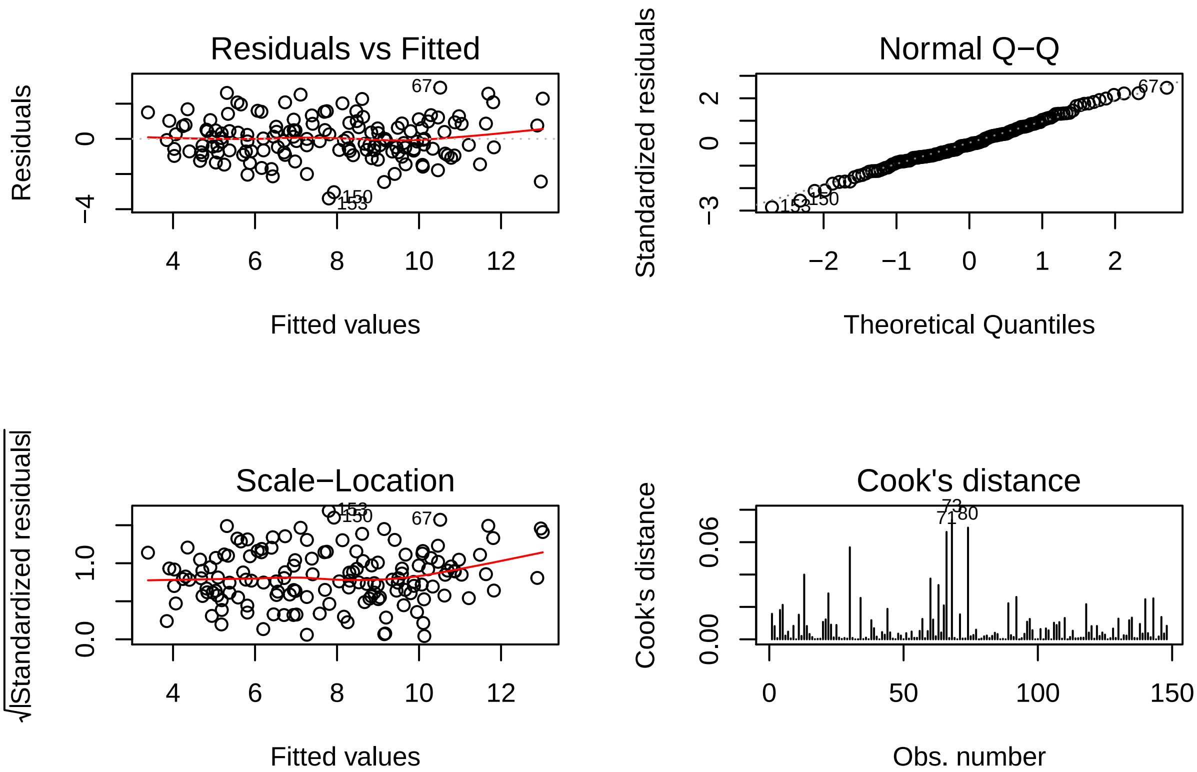
```
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + MedAge +
##     Urban + Popchange:MedAge, data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3900 -0.7189 -0.0822  0.7686  2.9115
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -9.7867167  1.1962954  -8.181 1.44e-13 ***
## log(Population)   0.7697621  0.0576253  13.358  < 2e-16 ***
## Popchange         0.0035745  0.0059724   0.599  0.55045
## MedAge            0.2049520  0.0330816   6.195 5.93e-09 ***
## Urban             0.0148287  0.0065603   2.260  0.02532 *
## Popchange:MedAge  0.0007072  0.0002613   2.707  0.00763 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.199 on 142 degrees of freedom
## Multiple R-squared:  0.7902, Adjusted R-squared:  0.7828
## F-statistic: 106.9 on 5 and 142 DF,  p-value: < 2.2e-16
```

```r
anova(m.final1,m.final2)
```

```
## Analysis of Variance Table
##
## Model 1: log(cases) ~ log(Population) + Popchange + MedAge + Urban
## Model 2: log(cases) ~ log(Population) + Popchange + MedAge + Urban + Popchange:MedAge
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1    143 214.70
## 2    142 204.17  1    10.534 7.3266 0.007628 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# p-value is 0.007628 **
# there is some suggestive evidence that m.final1 should be rejected
# and the interaction model m.final2 is more appropriate.
```

```r
par(mfrow=c(2,2))
plot(m.final2, which = 1:4,sub.caption = "Final Model Diagnostic Plots")
```

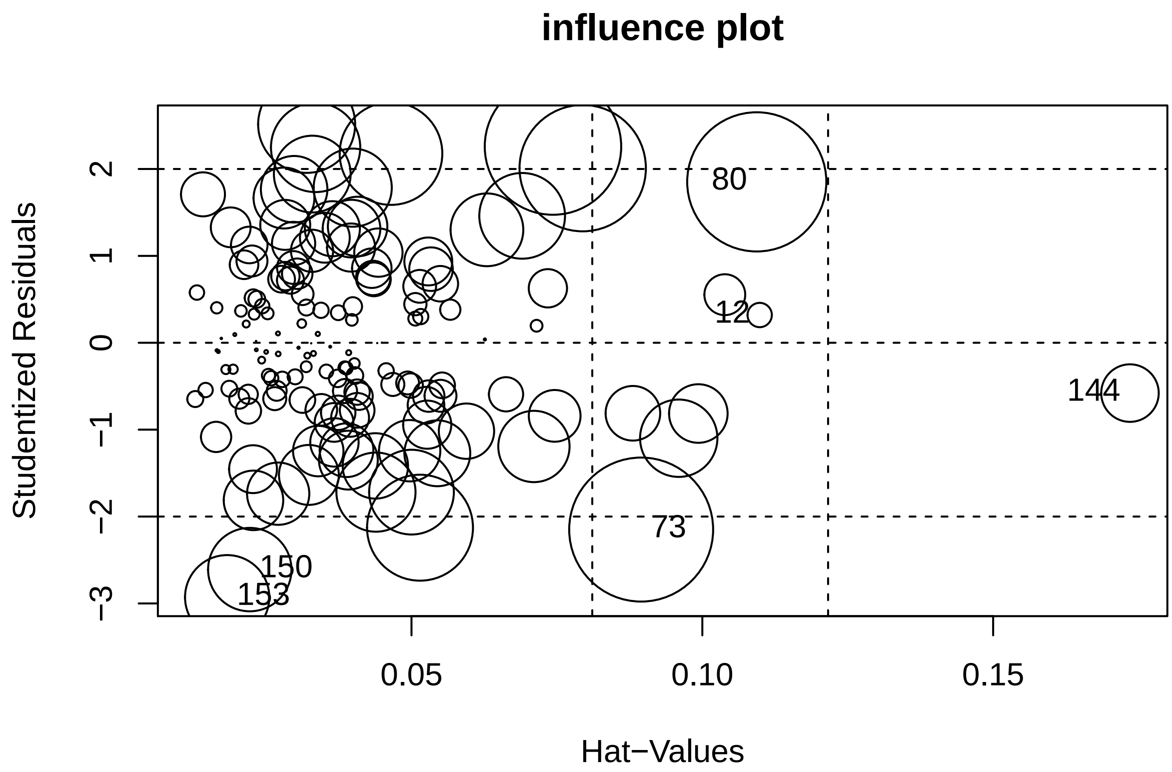



```r
# 1. pretty close to 0, good
# 2. looks normal
# 3. pretty random points
# 4. only three influencial less than 10% , it is ok
```

```r
outlierTest(m.final2)
```

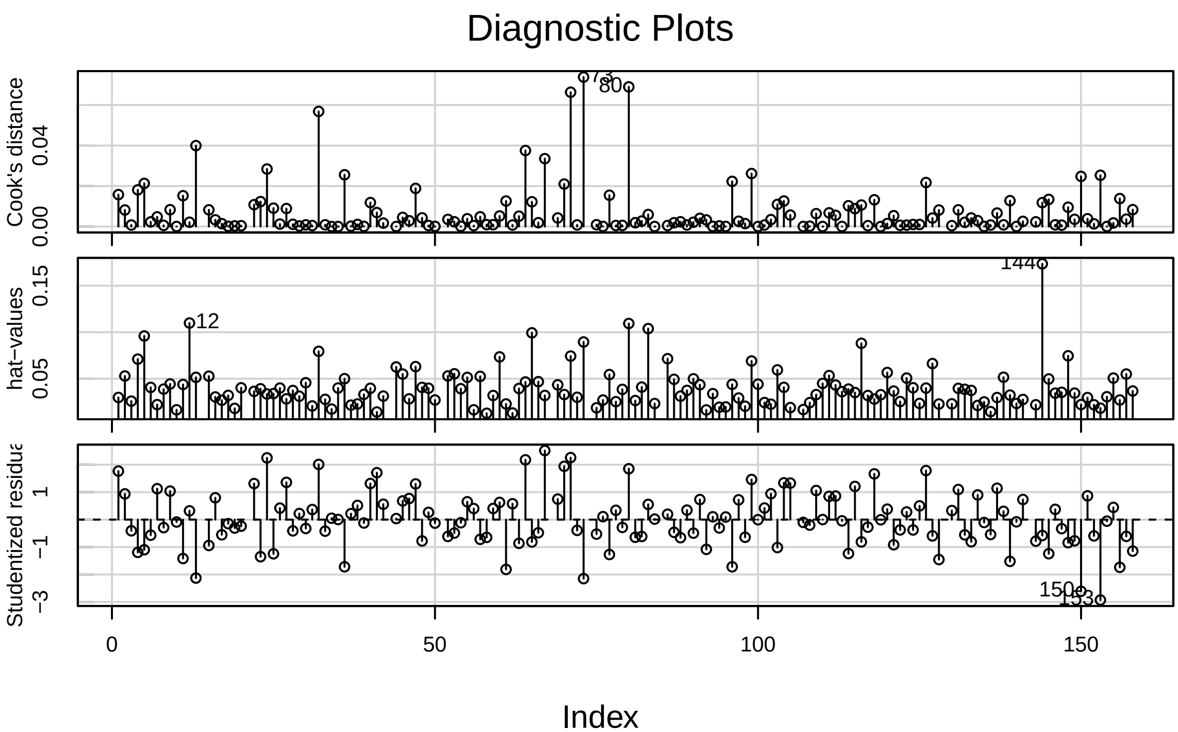
```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 153 -2.928565          0.0039723       0.5879
```

```r
influencePlot(m.final2,main="influence plot")
```




```
##         StudRes        Hat       CookD
## 12   0.3193408 0.10986133 0.002111058
## 73  -2.1499279 0.08948401 0.073826996
## 80   1.8516540 0.10935734 0.068983929
## 144 -0.5790187 0.17352214 0.011786786
## 150 -2.6103648 0.02219563 0.024765065
## 153 -2.9285652 0.01830887 0.025308721
```

```
infIndexPlot(m.final2, vars=c("Cook","hat","Student"))
```




```r
# question about:
# relationship between Confirmed Cases and Urbanization

exp(coef(m.final2)["Urban"])
```

```
##    Urban
## 1.014939
```

```r
exp(confint(m.final2)[5,])
```

```
##    2.5 %   97.5 %
## 1.001862 1.028187
```

```r
# For the same Population,Popchange, MedAge,
# the cases will be increased by 1.015  times as the Urban increased by one unit.
# 95% confidence interval is between 1.002 and  1.028
```

```r
summary(m.full)
```

```
##
## Call:
## lm(formula = log(cases) ~ log(Population) + Popchange + log(Density) +
##     Migrant + Fert + MedAge + Urban + WorldShare, data = data_train)
```

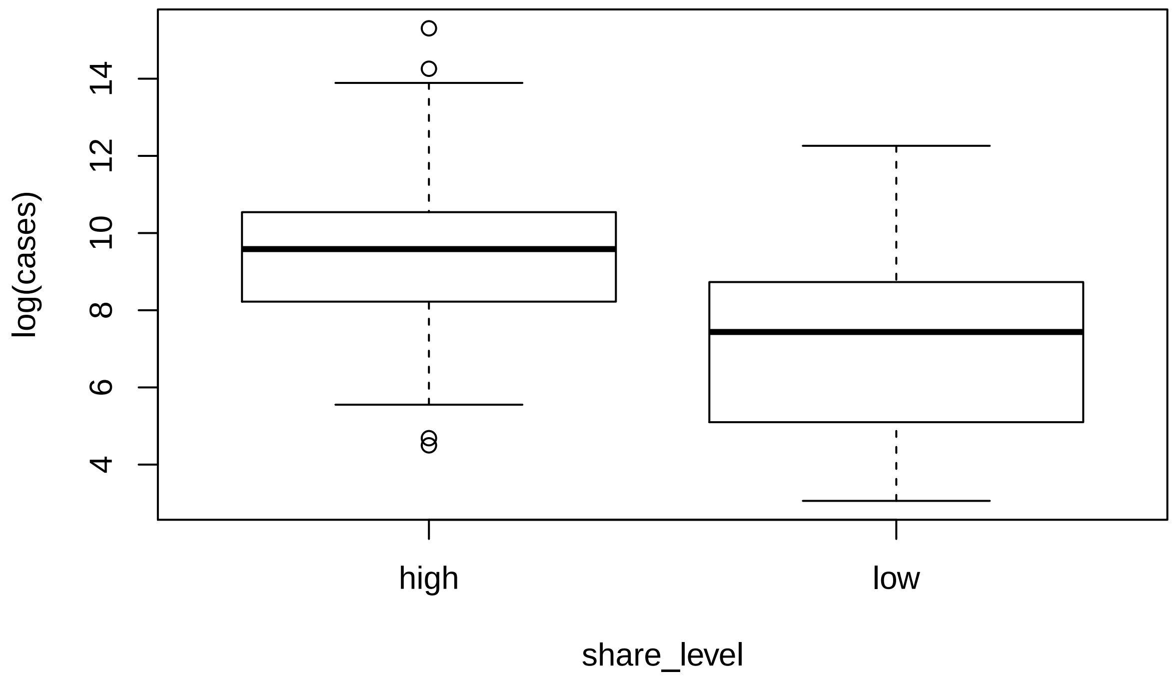
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0825 -0.7634 -0.0710  0.6785  3.3087
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.087e+01  2.081e+00  -5.226 6.22e-07 ***
## log(Population)  8.028e-01  1.378e-01   5.825 3.80e-08 ***
## Popchange        1.304e-02  4.681e-03   2.787  0.00607 **
## log(Density)     1.110e-01  7.769e-02   1.428  0.15549
## Migrant          1.695e-06  9.632e-07   1.760  0.08067 .
## Fert            -2.100e-02  1.869e-02  -1.124  0.26308
## MedAge           2.152e-01  3.393e-02   6.342 2.97e-09 ***
## Urban            2.118e-02  6.714e-03   3.155  0.00197 **
## WorldShare      -4.092e-03  1.171e-02  -0.349  0.72735
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.218 on 139 degrees of freedom
## Multiple R-squared:  0.788,  Adjusted R-squared:  0.7758
## F-statistic: 64.59 on 8 and 139 DF,  p-value: < 2.2e-16
```

```r
# we have known that the number of confirmed cases is related to
# Population,Popchange, Median Age and Urbanization

# question: is  there some relationship
# beween the number of confirmed cases and WorldShare level?

# beween the number of confirmed cases and Migrant level?
```

```r
# explore: relationship beween the number of confirmed cases and WorldShare level
sort(data_global$WorldShare)
```

```
##   [1]  1  1  1  1  1  1  2  2  2  2  2  2  2  2  2  2  2  2  2  3  3  3  3  3  3
##  [26]  3  3  4  4  4  4  4  4  4  4  5  5  5  5  5  5  5  6  6  6  6  6  7  7  7
##  [51]  7  7  7  8  8  8  8  8  8  9  9  9 10 10 10 10 10 12 12 12 12 12 13 13 13
##  [76] 14 14 14 14 14 14 14 15 16 16 16 16 16 16 17 18 18 19 20 21 21 21 22 22 23
## [101] 23 24 24 25 25 25 26 27 27 28 29 30 31 31 32 34 34 35 35 35 36 37 38 39 40
## [126] 41 42 43 43 43 44 45 46 47 49 51 52 53 54 55 56 57 58 58 60 61 62 63 64 65
## [151] 66 67 68 69 70 71 72 73
```

```r
data_global$share_level=as.factor(ifelse(data_global$WorldShare<35,"low","high"))
boxplot(log(cases)~share_level, data=data_global)
```




```r
with(data_global, tapply(log(cases), share_level, summary)) # Sumary statistics
```

```
## $high
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.498   8.223   9.585   9.569  10.542  15.306
##
## $low
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.060   5.099   7.437   7.187   8.730  12.261
```

```r
n <- with(data_global, tapply(log(cases), share_level, length))

ybar <- with(data_global, tapply(log(cases), share_level, mean))

s <- with(data_global, tapply(log(cases), share_level, sd))

round(cbind(n, ybar, s), 4)
```

```
##        n   ybar      s
## high  41 9.5690 2.5896
## low  117 7.1871 2.3008
```

```r
# Estimated difference in means
exp(as.numeric( ybar[1] - ybar[2] ))
```

```
## [1] 10.82526
```

```r
# that mean  the median number of confirmed cases is
# with high WorldShare level as same about 10.8 times
# as with low WorldShare level


t.test(log(cases)~share_level, data=data_global,alternative="greater",var.equal=T)
```

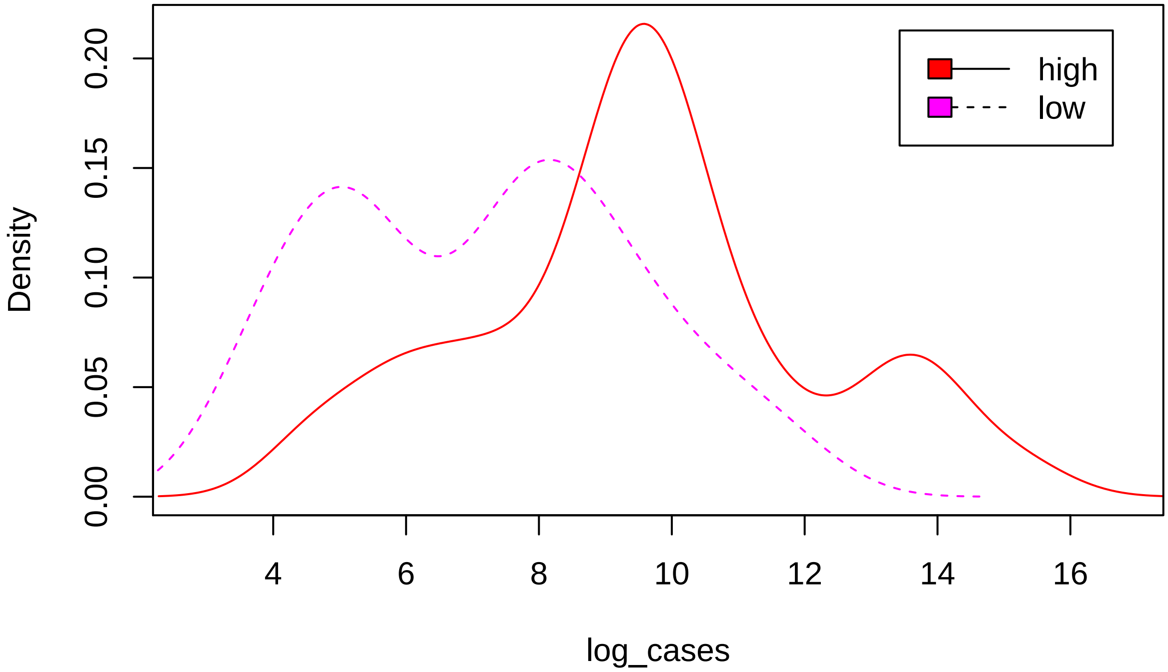
```
##
##  Two Sample t-test
##
## data:  log(cases) by share_level
## t = 5.5186, df = 156, p-value = 6.967e-08
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  1.667703      Inf
## sample estimates:
## mean in group high  mean in group low
##           9.568971           7.187089
```

```r
# p-value = 6.967e-08
# Do a 95% confidence interval for the median difference
log_CI=t.test(log(cases)~share_level, data=data_global,var.equal=T)$conf.int
CI=exp(log_CI)
CI
```

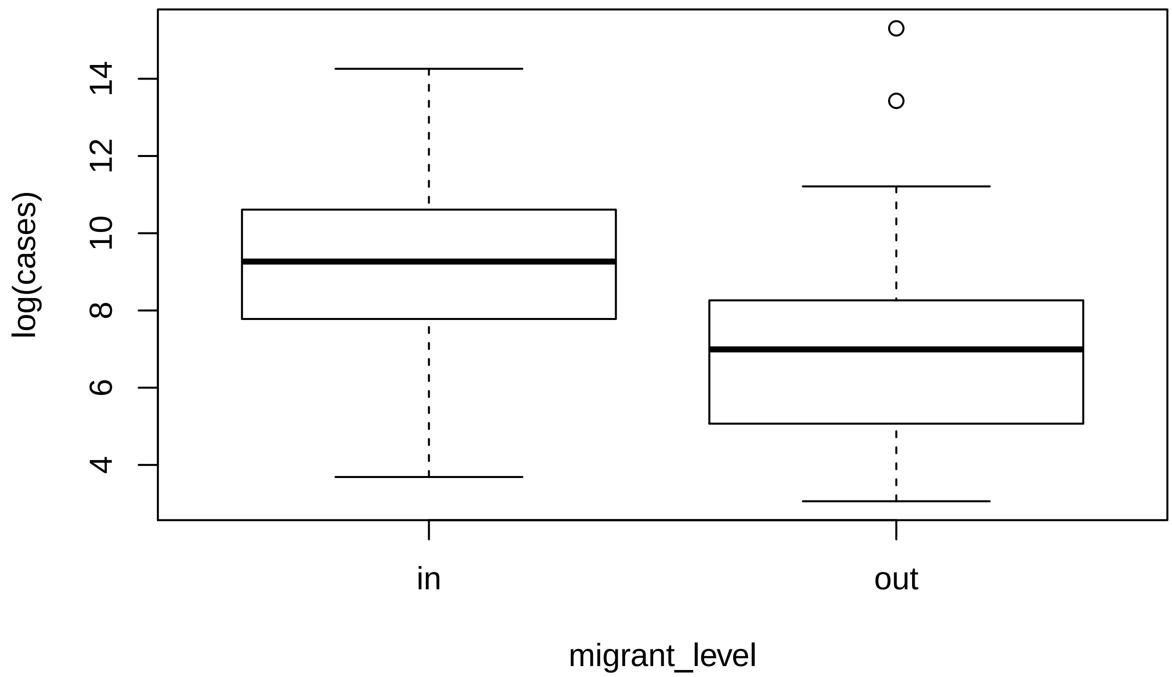
```
## [1]  4.615058 25.392169
## attr(,"conf.level")
## [1] 0.95
```

```r
# 95% confidence interval is between 4.6 and 25.4
```

```r
# Plot "density curves" (smoothed-out histograms) of high and low share_level
xr <- range(data_global$log_cases) * c(0.9, 1.1)
den.high <- with(data_global, density(log(cases)[share_level=="high"]))
den.low<- with(data_global, density(log(cases)[share_level=="low"]))


plot(den.high$y ~ den.high$x, type="l",
     xlim=xr, xlab="log_cases", ylab="Density",col=2)
lines(den.low, lty=2,col=6)
legend("topright", inset=.05, lty=1:2, legend=c("high","low"),fill=c(2,6))
```




```r
# same with Migrant level

data_global$migrant_level=
  as.factor(ifelse(data_global$Migrant<=0,"out","in"))

boxplot(log(cases)~migrant_level, data=data_global)
```




```
n <- with(data_global, tapply(log(cases), migrant_level, length))

ybar <- with(data_global, tapply(log(cases), migrant_level, mean))

s <- with(data_global, tapply(log(cases), migrant_level, sd))

round(cbind(n, ybar, s), 4)
```

```
##      n   ybar      s
## in  68 8.9711 2.5761
## out 90 6.9243 2.2427
```

```
# Estimated difference in means
exp(as.numeric( ybar[1] - ybar[2] ))
```

```
## [1] 7.743381
```

```
# that mean  the median number of confirmed cases is
# with high WorldShare level as same about 7.7 times
# as with low WorldShare level

t.test(log(cases)~migrant_level, data=data_global,alternative="greater",var.equal=T)
```

```
##
```

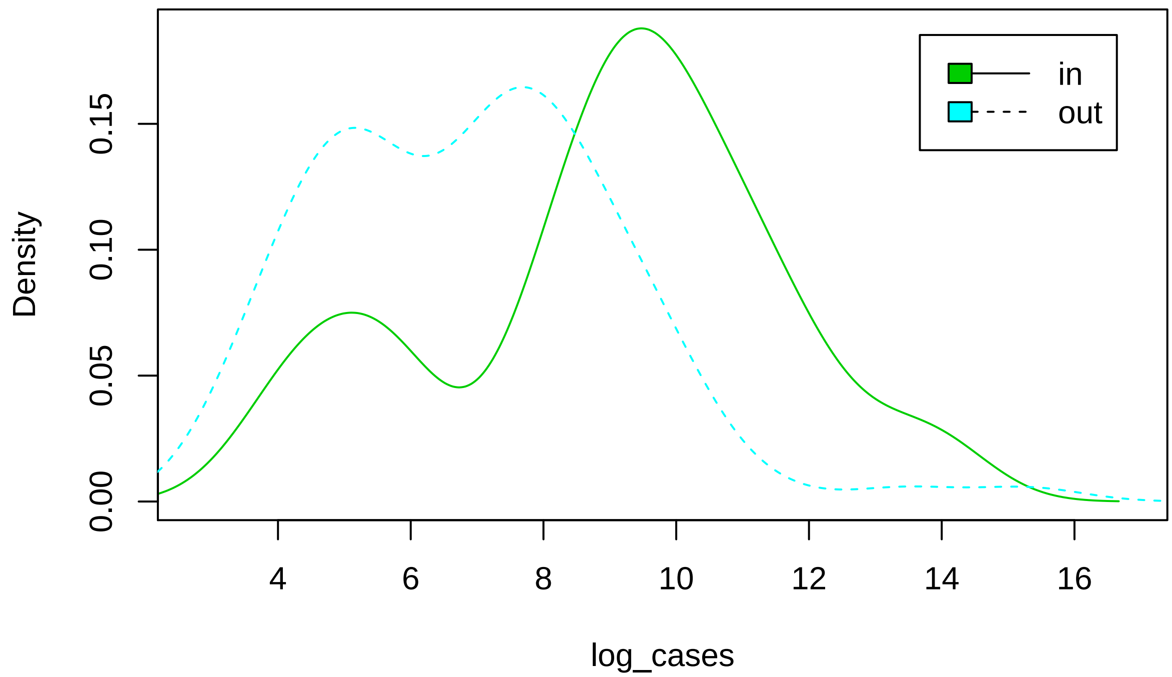
```
##  Two Sample t-test
##
## data:  log(cases) by migrant_level
## t = 5.3265, df = 156, p-value = 1.721e-07
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  1.410986      Inf
## sample estimates:
##  mean in group in mean in group out
##          8.971093          6.924254
```

```
# p-value = 0.0000001721
# Do a 95% confidence interval for the median difference
log_CI=t.test(log(cases)~migrant_level, data=data_global,var.equal=T)$conf.int
CI=exp(log_CI)
CI
```

```
## [1]  3.624745 16.541837
## attr(,"conf.level")
## [1] 0.95
```

```
# 95% confidence interval is between 3.6and 16.5
```

```
# Plot "density curves" (smoothed-out histograms) of in and out migrant_level
den.in <- with(data_global, density(log(cases)[migrant_level=="in"]))
den.out<- with(data_global, density(log(cases)[migrant_level=="out"]))


plot(den.in$y ~ den.in$x, type="l",
     xlim=xr, xlab="log_cases", ylab="Density",col=3)
lines(den.out, lty=2,col=5)
legend("topright", inset=.05, lty=1:2, legend=c("in","out"),fill=c(3,5))
```




```
ci_test=predict(m.final2, newdata=newdata, interval="confidence")%>%
  as_tibble()
head(ci_test)
```

```
## # A tibble: 6 x 3
##     fit   lwr   upr
##   <dbl> <dbl> <dbl>
## 1  8.35  7.88  8.83
## 2  7.15  6.70  7.60
## 3  5.32  4.87  5.76
## 4  6.74  6.18  7.31
## 5  6.22  5.58  6.86
## 6 11.7  11.3  12.1
```

```
ci_train=predict(m.final2, newdata=data_train, interval="confidence")%>%
  as_tibble()
head(ci_train)
```

```
## # A tibble: 6 x 3
##     fit   lwr   upr
##   <dbl> <dbl> <dbl>
## 1  5.57  5.16  5.98
## 2  6.94  6.40  7.49
## 3  9.45  9.07  9.83
## 4  5.88  5.24  6.51
```

```
## 5  4.66  3.93  5.39
## 6  9.87  9.39 10.3
```

```r
# now we use this model formula to do some predictions

predict1=predict(m.final2, newdata=newdata, interval="prediction")%>%
  as_tibble()

t1=predict1%>%
  mutate(true=newdata$log_cases)

head(t1)
```

```
## # A tibble: 6 x 4
##     fit   lwr   upr  true
##   <dbl> <dbl> <dbl> <dbl>
## 1  8.35  5.93 10.8   8.22
## 2  7.15  4.74  9.56  9.07
## 3  5.32  2.90  7.73  4.13
## 4  6.74  4.31  9.18  5.23
## 5  6.22  3.77  8.68  5.79
## 6 11.7   9.30 14.1  13.3
```

```r
mse1=mean((t1$fit-t1$true)^2)
mse1
```

```
## [1] 1.412156
```

```r
# mean squared error of test data is 1.412156

predict2=predict(m.final2, newdata=data_train, interval="prediction")%>%
  as_tibble()


t2=predict2%>%
  mutate(true=data_train$log_cases)
head(t2)
```

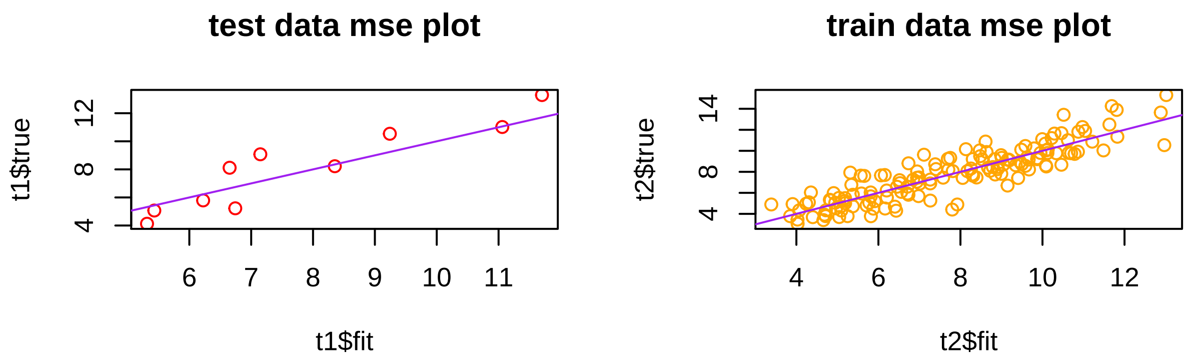
```
## # A tibble: 6 x 4
##     fit   lwr   upr  true
##   <dbl> <dbl> <dbl> <dbl>
## 1  5.57  3.16  7.97  7.64
## 2  6.94  4.51  9.38  8.04
## 3  9.45  7.05 11.9   8.97
## 4  5.88  3.42  8.33  4.50
## 5  4.66  2.18  7.14  3.41
## 6  9.87  7.45 12.3   9.20
```

```r
mse2=mean((t2$fit-t2$true)^2)
mse2
```

```
## [1] 1.379499
```

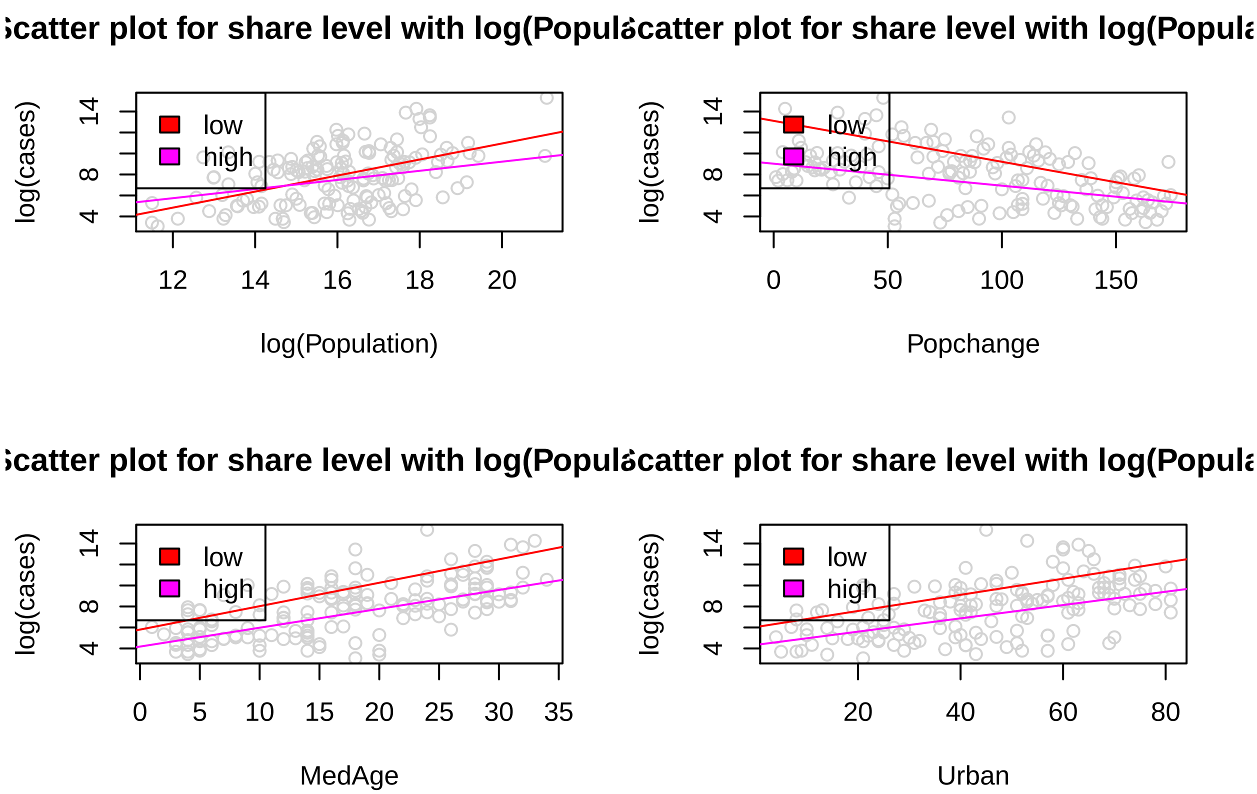
```
# mean squared error of train data is 1.379499,
# the model is good.
```

```
par(mfrow=c(2,2))
plot(t1$fit,t1$true,col="red",main="test data mse plot")
abline(0,1,col="purple")
plot(t2$fit,t2$true,col="orange",main="train data mse plot")
abline(0,1,col="purple")
# the two plots are both Consistent with the lines y=x+0
```



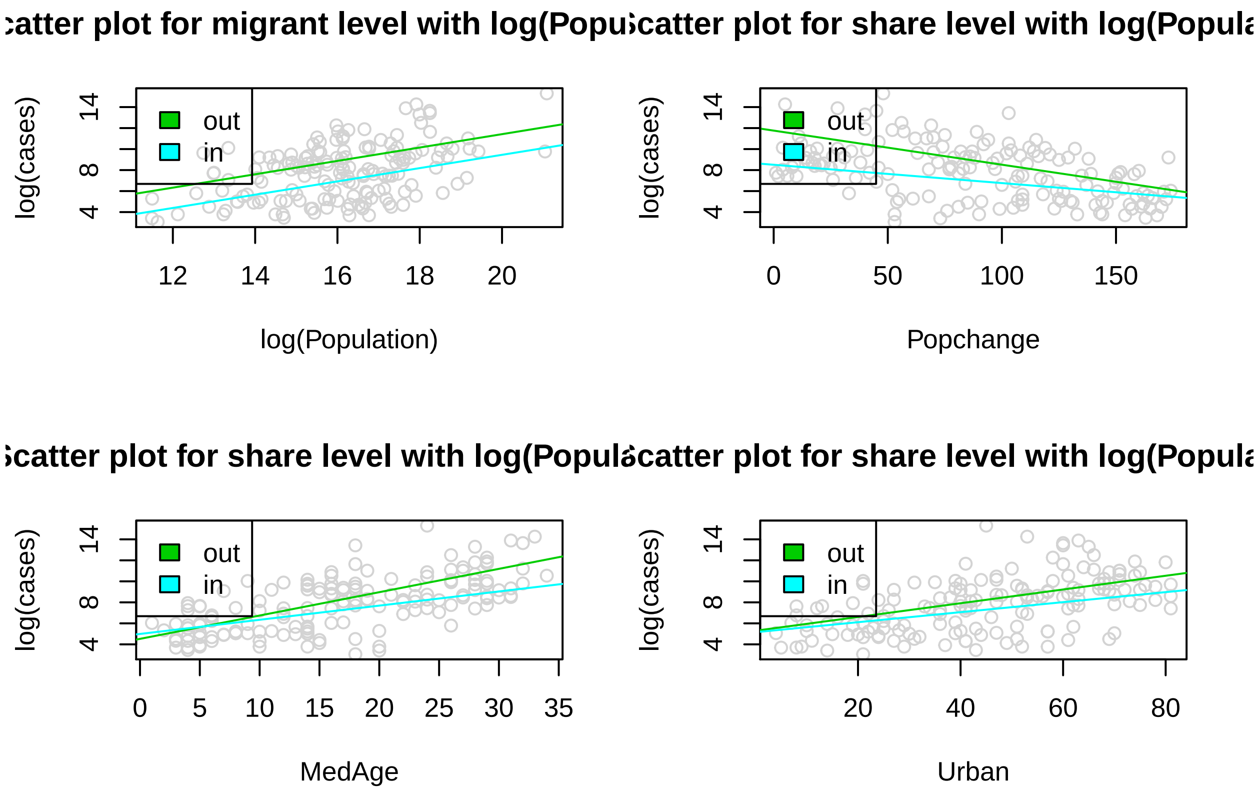

```
#interaction scatter plot for WorldShare level
par(mfrow=c(2,2))
#
plot(log(data_global$Population),log(data_global$cases),col="lightgrey",
     xlab="log(Population)",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$share_level=="high"])~
            log(data_global$Population[data_global$share_level=="high"])),
        col=2)
abline(lm(log(data_global$cases[data_global$share_level=="low"])~
            log(data_global$Population[data_global$share_level=="low"])),
        col=6)

legend("topleft",legend=c("low","high"),
```

```r
       fill=c(2,6))
#

plot(data_global$Popchange,log(data_global$cases),col="lightgrey",
     xlab="Popchange",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$share_level=="high"])~
            data_global$Popchange[data_global$share_level=="high"]),
       col=2)
abline(lm(log(data_global$cases[data_global$share_level=="low"])~
            data_global$Popchange[data_global$share_level=="low"]),
       col=6)

legend("topleft",legend=c("low","high"),
       fill=c(2,6))



#
plot(data_global$MedAge,log(data_global$cases),col="lightgrey",
     xlab="MedAge",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$share_level=="high"])~
            data_global$MedAge[data_global$share_level=="high"]),
       col=2)
abline(lm(log(data_global$cases[data_global$share_level=="low"])~
            data_global$MedAge[data_global$share_level=="low"]),
       col=6)

legend("topleft",legend=c("low","high"),
       fill=c(2,6))



#
plot(data_global$Urban,log(data_global$cases),col="lightgrey",
     xlab="Urban",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$share_level=="high"])~
            data_global$Urban[data_global$share_level=="high"]),
       col=2)
abline(lm(log(data_global$cases[data_global$share_level=="low"])~
            data_global$Urban[data_global$share_level=="low"]),
       col=6)

legend("topleft",legend=c("low","high"),
       fill=c(2,6))
```




```
#interaction scatter plot for migrant level
par(mfrow=c(2,2))
#
plot(log(data_global$Population),log(data_global$cases),col="lightgrey",
     xlab="log(Population)",ylab="log(cases)",
     main = "Scatter plot for migrant level with log(Population)")

abline(lm(log(data_global$cases[data_global$migrant_level=="in"])~
            log(data_global$Population[data_global$migrant_level=="in"])),
       col=3)
abline(lm(log(data_global$cases[data_global$migrant_level=="out"])~
            log(data_global$Population[data_global$migrant_level=="out"])),
       col=5)

legend("topleft",legend=c("out","in"),
       fill=c(3,5))
#

plot(data_global$Popchange,log(data_global$cases),col="lightgrey",
     xlab="Popchange",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$migrant_level=="in"])~
            data_global$Popchange[data_global$migrant_level=="in"]),
       col=3)
abline(lm(log(data_global$cases[data_global$migrant_level=="out"])~
```

```r
            data_global$Popchange[data_global$migrant_level=="out"]),
       col=5)

legend("topleft",legend=c("out","in"),
       fill=c(3,5))



#
plot(data_global$MedAge,log(data_global$cases),col="lightgrey",
     xlab="MedAge",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$migrant_level=="in"])~
            data_global$MedAge[data_global$migrant_level=="in"]),
       col=3)
abline(lm(log(data_global$cases[data_global$migrant_level=="out"])~
            data_global$MedAge[data_global$migrant_level=="out"]),
       col=5)

legend("topleft",legend=c("out","in"),
       fill=c(3,5))



#
plot(data_global$Urban,log(data_global$cases),col="lightgrey",
     xlab="Urban",ylab="log(cases)",
     main = "Scatter plot for share level with log(Population)")

abline(lm(log(data_global$cases[data_global$migrant_level=="in"])~
            data_global$Urban[data_global$migrant_level=="in"]),
       col=3)
abline(lm(log(data_global$cases[data_global$migrant_level=="out"])~
            data_global$Urban[data_global$migrant_level=="out"]),
       col=5)

legend("topleft",legend=c("out","in"),
       fill=c(3,5))
```




```r
m.cp1=lm(formula = log_casespop~  Popchange + log(Density) + MedAge +
    Urban + WorldShare, data = data_train)
summary(m.cp1)
```

```
##
## Call:
## lm(formula = log_casespop ~ Popchange + log(Density) + MedAge +
##     Urban + WorldShare, data = data_train)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.212259 -0.050063  0.001162  0.050051  0.197845
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0115019  0.0493963   0.233 0.816214
## Popchange    0.0010133  0.0002739   3.699 0.000308 ***
## log(Density) 0.0056335  0.0048250   1.168 0.244938
## MedAge       0.0157798  0.0017414   9.061 9.29e-16 ***
## Urban        0.0014767  0.0004228   3.493 0.000638 ***
## WorldShare   0.0011707  0.0003182   3.679 0.000332 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07784 on 142 degrees of freedom
```

```
## Multiple R-squared:  0.724,  Adjusted R-squared:  0.7143
## F-statistic: 74.49 on 5 and 142 DF,  p-value: < 2.2e-16
```

```
m.cp0<- lm(log_casespop~1,data=data_train)

# this time we try to use stepwise backward method


step(m.cp1,scope=m.cp0,direction=c("backward"))
```

```
## Start:  AIC=-749.82
## log_casespop ~ Popchange + log(Density) + MedAge + Urban + WorldShare
##
##                 Df Sum of Sq     RSS     AIC
## - log(Density)  1   0.00826 0.86875 -750.41
## <none>                      0.86049 -749.82
## - Urban         1   0.07392 0.93441 -739.63
## - WorldShare    1   0.08202 0.94251 -738.35
## - Popchange     1   0.08292 0.94342 -738.21
## - MedAge        1   0.49755 1.35805 -684.29
##
## Step:  AIC=-750.41
## log_casespop ~ Popchange + MedAge + Urban + WorldShare
##
##               Df Sum of Sq     RSS     AIC
## <none>                    0.86875 -750.41
## - Urban       1   0.06638 0.93514 -741.51
## - Popchange   1   0.08798 0.95673 -738.13
## - WorldShare  1   0.08914 0.95789 -737.95
## - MedAge      1   0.54912 1.41788 -679.91
```

```
##
## Call:
## lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare,
##     data = data_train)
##
## Coefficients:
## (Intercept)    Popchange       MedAge        Urban   WorldShare
##    0.030599     0.001040     0.016207     0.001359     0.001213
```

```
# p-value is 0.007628 **
# there is some suggestive evidence that m.final1 should be rejected
# and the interaction model m.final2 is more appropriate.
```

```
m.cp2=lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare,
    data = data_train)
summary(m.cp2)
```

```
##
## Call:
## lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare,
##     data = data_train)
```

```
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.207239 -0.050428 -0.001568  0.047881  0.182313
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0305986  0.0466690   0.656 0.513103
## Popchange   0.0010401  0.0002733   3.805 0.000209 ***
## MedAge      0.0162071  0.0017047   9.507  < 2e-16 ***
## Urban       0.0013591  0.0004112   3.306 0.001198 **
## WorldShare  0.0012126  0.0003166   3.830 0.000191 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07794 on 143 degrees of freedom
## Multiple R-squared:  0.7213, Adjusted R-squared:  0.7135
## F-statistic: 92.54 on 4 and 143 DF,  p-value: < 2.2e-16
```

```
#interaction:
m.cp3=lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare+
            Popchange *MedAge + Popchange * Urban + Popchange *WorldShare +
           MedAge* Urban +MedAge*WorldShare+ Urban*WorldShare,
    data = data_train)
summary(m.cp3)
```

```
##
## Call:
## lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare +
##     Popchange * MedAge + Popchange * Urban + Popchange * WorldShare +
##     MedAge * Urban + MedAge * WorldShare + Urban * WorldShare,
##     data = data_train)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.200820 -0.044021 -0.002126  0.047104  0.157094
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.784e-01  1.446e-01   1.233   0.2196
## Popchange             2.744e-06  8.383e-04   0.003   0.9974
## MedAge                9.152e-03  5.071e-03   1.805   0.0733 .
## Urban                 8.855e-04  2.627e-03   0.337   0.7366
## WorldShare           -1.840e-03  3.159e-03  -0.582   0.5612
## Popchange:MedAge      4.273e-05  1.991e-05   2.146   0.0336 *
## Popchange:Urban      -2.786e-06  1.473e-05  -0.189   0.8503
## Popchange:WorldShare  2.472e-05  1.916e-05   1.290   0.1993
## MedAge:Urban          5.320e-05  8.253e-05   0.645   0.5203
## MedAge:WorldShare     8.209e-05  1.105e-04   0.743   0.4587
## Urban:WorldShare     -1.151e-05  2.248e-05  -0.512   0.6093
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07715 on 137 degrees of freedom
```

```
## Multiple R-squared:  0.7384, Adjusted R-squared:  0.7193
## F-statistic: 38.68 on 10 and 137 DF,  p-value: < 2.2e-16
```

```r
step(m.cp3,scope=m.cp0,direction=c("backward"))
```

```
## Start:  AIC=-747.78
## log_casespop ~ Popchange + MedAge + Urban + WorldShare + Popchange *
##     MedAge + Popchange * Urban + Popchange * WorldShare + MedAge *
##     Urban + MedAge * WorldShare + Urban * WorldShare
##
##                        Df Sum of Sq     RSS     AIC
## - Popchange:Urban       1 0.0002129 0.81565 -749.75
## - Urban:WorldShare      1 0.0015619 0.81700 -749.50
## - MedAge:Urban          1 0.0024727 0.81791 -749.34
## - MedAge:WorldShare     1 0.0032865 0.81872 -749.19
## - Popchange:WorldShare  1 0.0099027 0.82534 -748.00
## <none>                              0.81544 -747.78
## - Popchange:MedAge      1 0.0274098 0.84285 -744.89
##
## Step:  AIC=-749.75
## log_casespop ~ Popchange + MedAge + Urban + WorldShare + Popchange:MedAge +
##     Popchange:WorldShare + MedAge:Urban + MedAge:WorldShare +
##     Urban:WorldShare
##
##                        Df Sum of Sq     RSS     AIC
## - Urban:WorldShare      1 0.0014413 0.81709 -751.48
## - MedAge:WorldShare     1 0.0031052 0.81876 -751.18
## - Popchange:WorldShare  1 0.0097010 0.82535 -750.00
## - MedAge:Urban          1 0.0107800 0.82643 -749.80
## <none>                              0.81565 -749.75
## - Popchange:MedAge      1 0.0295820 0.84523 -746.47
##
## Step:  AIC=-751.48
## log_casespop ~ Popchange + MedAge + Urban + WorldShare + Popchange:MedAge +
##     Popchange:WorldShare + MedAge:Urban + MedAge:WorldShare
##
##                        Df Sum of Sq     RSS     AIC
## - MedAge:WorldShare     1 0.0017568 0.81885 -753.17
## - Popchange:WorldShare  1 0.0082927 0.82538 -751.99
## <none>                              0.81709 -751.48
## - MedAge:Urban          1 0.0121871 0.82928 -751.29
## - Popchange:MedAge      1 0.0300613 0.84715 -748.14
##
## Step:  AIC=-753.17
## log_casespop ~ Popchange + MedAge + Urban + WorldShare + Popchange:MedAge +
##     Popchange:WorldShare + MedAge:Urban
##
##                        Df Sum of Sq     RSS     AIC
## <none>                              0.81885 -753.17
## - MedAge:Urban          1  0.012678 0.83153 -752.89
## - Popchange:WorldShare  1  0.017129 0.83598 -752.10
## - Popchange:MedAge      1  0.029530 0.84838 -749.92

##
```

```
## Call:
## lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare +
##     Popchange:MedAge + Popchange:WorldShare + MedAge:Urban, data = data_train)
##
## Coefficients:
##          (Intercept)             Popchange                MedAge
##            1.761e-01             5.812e-05             9.624e-03
##                Urban            WorldShare      Popchange:MedAge
##            1.263e-04             1.543e-04             4.127e-05
## Popchange:WorldShare          MedAge:Urban
##            1.227e-05             7.051e-05
```

```r
m.cp4=lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare +
    Popchange:MedAge + Popchange:WorldShare, data = data_train)
summary(m.cp4)
```

```
##
## Call:
## lm(formula = log_casespop ~ Popchange + MedAge + Urban + WorldShare +
##     Popchange:MedAge + Popchange:WorldShare, data = data_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20291 -0.04482 -0.00203  0.05155  0.17210
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          9.296e-02  5.599e-02   1.660  0.09906 .
## Popchange            3.483e-04  3.990e-04   0.873  0.38417
## MedAge               1.419e-02  2.114e-03   6.713 4.31e-10 ***
## Urban                1.301e-03  4.282e-04   3.038  0.00284 **
## WorldShare           1.481e-04  6.838e-04   0.217  0.82882
## Popchange:MedAge     2.926e-05  1.652e-05   1.771  0.07878 .
## Popchange:WorldShare 1.299e-05  7.186e-06   1.807  0.07287 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07679 on 141 degrees of freedom
## Multiple R-squared:  0.7333, Adjusted R-squared:  0.7219
## F-statistic: 64.61 on 6 and 141 DF,  p-value: < 2.2e-16
```