Case-Shiller 3-Tier City Index Assessment

```r
# load necessary library
library(forecast) # for prediction

# load all csv file into dataframe
temp = list.files(pattern="*.csv")
for (i in 1:length(temp)) {
  assign(strtrim(temp[i], (nchar(temp[i]) - 4)), read.csv(temp[i]))
}

# convert Boston Home Index data (3 tiers) into time series object
boxrhtnsats <- ts(BOXRHTNSA[,2], start = c(1987, 1), frequency = 12)
boxrltnsats <- ts(BOXRLTNSA[,2], start = c(1987, 1), frequency = 12)
boxrmtnsats <- ts(BOXRMTNSA[,2], start = c(1987, 1), frequency = 12)

# the cross-validation implied here is a rolling basis
# for example, the first training group is year 1 and test group is year 2
# then, the second training group is year 1 and 2 and test group is year 3
# the training group keeps rolling up each iteration and test group is always the next 12 months
# maybe except for the last year
# whatever left in the last year (< 12 months) will be added to the previous year as a test group

# to obtain the number of trainning groups and the number of months in the last test group
numy.bos = floor(length(boxrhtnsats) / 12) - 1
extra.bos = length(boxrhtnsats) - numy.bos * 12

# for high tier Index
# to create one numeric vector for storing mae value and one list for ARIMA parameters
models.bosht <- vector("list", numy.bos)
maes.bosht <- vector("numeric", numy.bos)

# loop through every year(except for only the frist year as a training set
# since the seasonal factor will fail on this set)
for (i in 2:numy.bos){

  # use auto.arima to find a best model on training set
  boxrhtnsafc <- auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D = 1)

  # store arima parameter
  models.bosht[[i]] <- boxrhtnsafc$arma

  # decide if the test group is the last year plus extra months
  if (i == numy.bos){
    pred <- forecast(boxrhtnsafc, extra.bos)

    # if the test group is the last year then test if it is a whole year
    if (extra.bos == 12){
      maes.bosht[i] <- mean(abs(pred$mean - window(boxrhtnsats, (1986 + i + 1), c(1986 + i + 1, 12))))
    } else {
      # if not a whole year then the extra months are added to the previous group
      maes.bosht[i] <- mean(abs(pred$mean - window(boxrhtnsats, (1986 + i + 1), c(1986 + i + 2, extra.b
  }

 # if the test group is not the last year then just predict the next 12 months
 } else {
  pred <- forecast(boxrhtnsafc, 12)
  maes.bosht[i] <- mean(abs(pred$mean - window(boxrhtnsats, (1986 + i + 1), c(1986 + i + 1, 12))))
 }
}
```

```
## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning in auto.arima(window(boxrhtnsats, 1987, c(1986 + i, 12)), D =
## 1): Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.
```

```
# the rolling basis cross-validation method may be improved in the future

# using mae criterion for the best fitting model
para.bosht <- models.bosht[[which.min(maes.bosht[2:numy.bos]) + 1]]

# fit the optimal model to the whole data set
# and predict for the next whole year
boxrhtnsafc <- arima(boxrhtnsats, order = c(para.bosht[1], para.bosht[6], para.bosht[2]), seasonal = li
prediction.bosht <- forecast(boxrhtnsafc, (24 - extra.bos))
pred.bosht <- prediction.bosht$mean
```