housing price index eda for EJW

Hongjie Wang

April 24, 2021

We show an example of getting data from web, perform some exploratory data analysis. It is only used as a simple demonstration for EJW to show high level steps.

In data analysis, one should also start with some questions or hypotheses that one hopes the data can provide some insights. Usually, more specific the questions are, the easier the task and more productive the process.

But sometimes, we are given the task to "find something interesting." Such tasks are actually very difficult to do. But they are good to practice some basic skills. Data science in my opinion is a bit like detective work. You want to bring vigorous logic, rich past experience, solid mathematical knowledge, versatile statistical techniques, a deep domain knowledge and finally your common sense to find useful patterns, make appropriate inferences and reach sensible conclusions and decisions.

In this demo, we show the steps of the high level analysis steps, not our underlying thinking, nor do we focus on any specific questions or findings.

First, we load some packages

```
rm(list = ls())
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.0.5
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
```

```
library(ggplot2)
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.0.5
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

We first obtain data from a table embedded in HTML page We use the functions in rvest package for this step.

This is a housing price data. You can look at the website to get more information of the data.

```
data_url<-"https://wiki.socr.umich.edu/index.php/SOCR_Data_Dinov_091609_SnP_HomePriceIndex"
wiki_url<- read_html(data_url)

mydata<-wiki_url%>%
  html_node("table")%>%
  html_table()
```

Some high level summary of the data to make sure all the types are correct. It is always a good idea to understand the definitions of data.But in that process, you will need to apply encapsulation. For example, you may want to know a particular field in the data is related to some medical risk factor. And you may want to know that the higher the worse the condition. But you may not need to get into the specific medical science part of it, at least not initially.

```
str(mydata)
```

```
## tibble [222 x 23] (S3: tbl_df/tbl/data.frame)
##  $ Index          : int [1:222] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Year           : int [1:222] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 ...
##  $ Month          : chr [1:222] "January" "February" "March" "April" ...
##  $ AZ-Phoenix     : num [1:222] 65.3 65.3 64.6 64.3 64.4 ...
##  $ CA-LosAngeles  : num [1:222] 95.3 94.1 92.8 92.8 93.4 ...
##  $ CA-SanDiego    : num [1:222] 83.1 81.9 80.9 80.7 81.4 ...
##  $ CA-SanFrancisco: num [1:222] 71.2 70.3 69.6 69.5 70.1 ...
##  $ CO-Denver      : num [1:222] 48.7 48.7 48.9 49.2 49.5 ...
##  $ DC-Washington  : num [1:222] 89.4 88.8 87.6 87.6 88.6 ...
##  $ FL-Miami       : num [1:222] 79.1 78.5 78.4 78.5 78 ...
##  $ FL-Tampa       : num [1:222] 81.8 81.8 81.4 81.5 81.3 ...
##  $ GA-Atlanta     : num [1:222] 69.6 69.2 69 69.4 69.7 ...
##  $ IL-Chicago     : num [1:222] 70 70.5 70.6 71.1 71.4 ...
##  $ MA-Boston      : num [1:222] 65 64.2 63.6 63.4 63.8 ...
##  $ MI-Detroit     : num [1:222] 58.2 57.8 57.6 57.9 58.4 ...
##  $ MN-Minneapolis : num [1:222] 64.2 64.2 64.2 64.3 64.8 ...
##  $ NC-Charlotte   : num [1:222] 73.3 73.3 72.8 72.9 73.3 ...
##  $ NV-LasVegas    : num [1:222] 81 81.6 81.7 81.7 82 ...
##  $ NY-NewYork     : num [1:222] 74.6 73.7 72.9 72.3 72.6 ...
##  $ OH-Cleveland   : num [1:222] 68.2 68 68.2 69.1 69.9 ...
##  $ OR-Portland    : num [1:222] 56.5 56.9 58 58.4 58.9 ...
##  $ WA-Seattle     : num [1:222] 65.5 64.6 64.5 65.1 66 ...
##  $ Composite-10   : num [1:222] 78.5 77.8 77 76.9 77.3 ...
```

```
head(mydata,10)
```

```
## # A tibble: 10 x 23
##    Index  Year Month `AZ-Phoenix` `CA-LosAngeles` `CA-SanDiego` `CA-SanFrancisc~
##    <int> <int> <chr>        <dbl>           <dbl>         <dbl>            <dbl>
##  1     1  1991 Janu~         65.3            95.3          83.1             71.2
##  2     2  1991 Febr~         65.3            94.1          81.9             70.3
##  3     3  1991 March         64.6            92.8          80.9             69.6
##  4     4  1991 April         64.4            92.8          80.7             69.5
##  5     5  1991 May           64.4            93.4          81.4             70.1
##  6     6  1991 June          64.9            94.2          82.2             70.8
##  7     7  1991 July          65.5            94.8          82.6             71.4
##  8     8  1991 Augu~         65.9            95.2          82.5             71.5
##  9     9  1991 Sept~         66.0            94.9          82.2             71.6
## 10    10  1991 Octo~         65.8            94.5          82.0             71.2
## # ... with 16 more variables: CO-Denver <dbl>, DC-Washington <dbl>,
## #   FL-Miami <dbl>, FL-Tampa <dbl>, GA-Atlanta <dbl>, IL-Chicago <dbl>,
## #   MA-Boston <dbl>, MI-Detroit <dbl>, MN-Minneapolis <dbl>,
## #   NC-Charlotte <dbl>, NV-LasVegas <dbl>, NY-NewYork <dbl>,
## #   OH-Cleveland <dbl>, OR-Portland <dbl>, WA-Seattle <dbl>, Composite-10 <dbl>
```

```
tail(mydata,5)
```

```
## # A tibble: 5 x 23
##   Index  Year Month  `AZ-Phoenix` `CA-LosAngeles` `CA-SanDiego` `CA-SanFranciscâ€¦
##   <int> <int> <chr>         <dbl>           <dbl>         <dbl>            <dbl>
## 1   218  2009 Februâ€¦         112.            163.          147.             120.
## 2   219  2009 March          107.            161.          145.             118.
## 3   220  2009 April          104.            159.          144.             118.
## 4   221  2009 May            104.            159.          145.             120.
## 5   222  2009 June           105.            161.          147.             125.
## # ... with 16 more variables: CO-Denver <dbl>, DC-Washington <dbl>,
## #   FL-Miami <dbl>, FL-Tampa <dbl>, GA-Atlanta <dbl>, IL-Chicago <dbl>,
## #   MA-Boston <dbl>, MI-Detroit <dbl>, MN-Minneapolis <dbl>,
## #   NC-Charlotte <dbl>, NV-LasVegas <dbl>, NY-NewYork <dbl>,
## #   OH-Cleveland <dbl>, OR-Portland <dbl>, WA-Seattle <dbl>, Composite-10 <dbl>
```

```
summary(mydata)
```

```
##      Index             Year          Month              AZ-Phoenix    
##  Min.   :  1.00   Min.   :1991   Length:222         Min.   : 64.35  
##  1st Qu.: 56.25   1st Qu.:1995   Class :character   1st Qu.: 77.75  
##  Median :111.50   Median :2000   Mode  :character   Median :101.78  
##  Mean   :111.50   Mean   :2000                      Mean   :114.39  
##  3rd Qu.:166.75   3rd Qu.:2004                      3rd Qu.:129.70  
##  Max.   :222.00   Max.   :2009                      Max.   :227.42  
##  CA-LosAngeles     CA-SanDiego     CA-SanFrancisco    CO-Denver     
##  Min.   : 73.07   Min.   : 71.22   Min.   : 65.79   Min.   : 48.67  
##  1st Qu.: 81.27   1st Qu.: 76.36   1st Qu.: 69.47   1st Qu.: 70.69  
##  Median :102.92   Median :104.34   Median :108.77   Median :102.53  
##  Mean   :135.83   Mean   :131.41   Mean   :119.18   Mean   : 99.17  
##  3rd Qu.:180.32   3rd Qu.:177.37   3rd Qu.:154.31   3rd Qu.:127.45  
##  Max.   :273.94   Max.   :250.34   Max.   :218.37   Max.   :140.28  
##  DC-Washington       FL-Miami         FL-Tampa        GA-Atlanta    
##  Min.   : 87.56   Min.   : 77.61   Min.   : 80.27   Min.   : 69.05  
##  1st Qu.: 89.19   1st Qu.: 87.04   1st Qu.: 87.05   1st Qu.: 79.65  
##  Median :102.52   Median :101.28   Median :101.39   Median :101.84  
##  Mean   :135.63   Mean   :135.34   Mean   :125.70   Mean   :100.51  
##  3rd Qu.:176.35   3rd Qu.:169.91   3rd Qu.:154.35   3rd Qu.:118.96  
##  Max.   :251.07   Max.   :280.87   Max.   :238.09   Max.   :136.47  
##    IL-Chicago       MA-Boston        MI-Detroit     MN-Minneapolis  
##  Min.   : 70.04   Min.   : 62.94   Min.   : 57.63   Min.   : 64.19  
##  1st Qu.: 83.41   1st Qu.: 70.10   1st Qu.: 70.50   1st Qu.: 76.02  
##  Median :102.16   Median :102.29   Median : 92.79   Median :101.30  
##  Mean   :111.44   Mean   :114.18   Mean   : 92.76   Mean   :110.41  
##  3rd Qu.:138.97   3rd Qu.:158.67   3rd Qu.:114.62   3rd Qu.:144.09  
##  Max.   :168.60   Max.   :182.45   Max.   :127.05   Max.   :171.12  
##   NC-Charlotte     NV-LasVegas       NY-NewYork      OH-Cleveland   
##  Min.   : 72.75   Min.   : 80.96   Min.   : 72.29   Min.   : 67.96  
##  1st Qu.: 83.96   1st Qu.: 88.68   1st Qu.: 78.88   1st Qu.: 84.15  
##  Median :101.59   Median :101.05   Median :101.84   Median : 99.68  
##  Mean   :100.36   Mean   :125.72   Mean   :125.10   Mean   : 98.21  
##  3rd Qu.:113.80   3rd Qu.:146.63   3rd Qu.:175.19   3rd Qu.:112.00  
##  Max.   :135.88   Max.   :234.78   Max.   :215.83   Max.   :123.49  
##   OR-Portland       WA-Seattle      Composite-10   
##  Min.   : 56.53   Min.   : 64.47   Min.   : 75.63  
##  1st Qu.: 81.28   1st Qu.: 72.49   1st Qu.: 77.94  
##  Median :101.45   Median :102.85   Median :103.12  
##  Mean   :110.39   Mean   :109.91   Mean   :125.40  
##  3rd Qu.:134.33   3rd Qu.:137.06   3rd Qu.:167.42  
##  Max.   :186.51   Max.   :192.30   Max.   :226.29
```

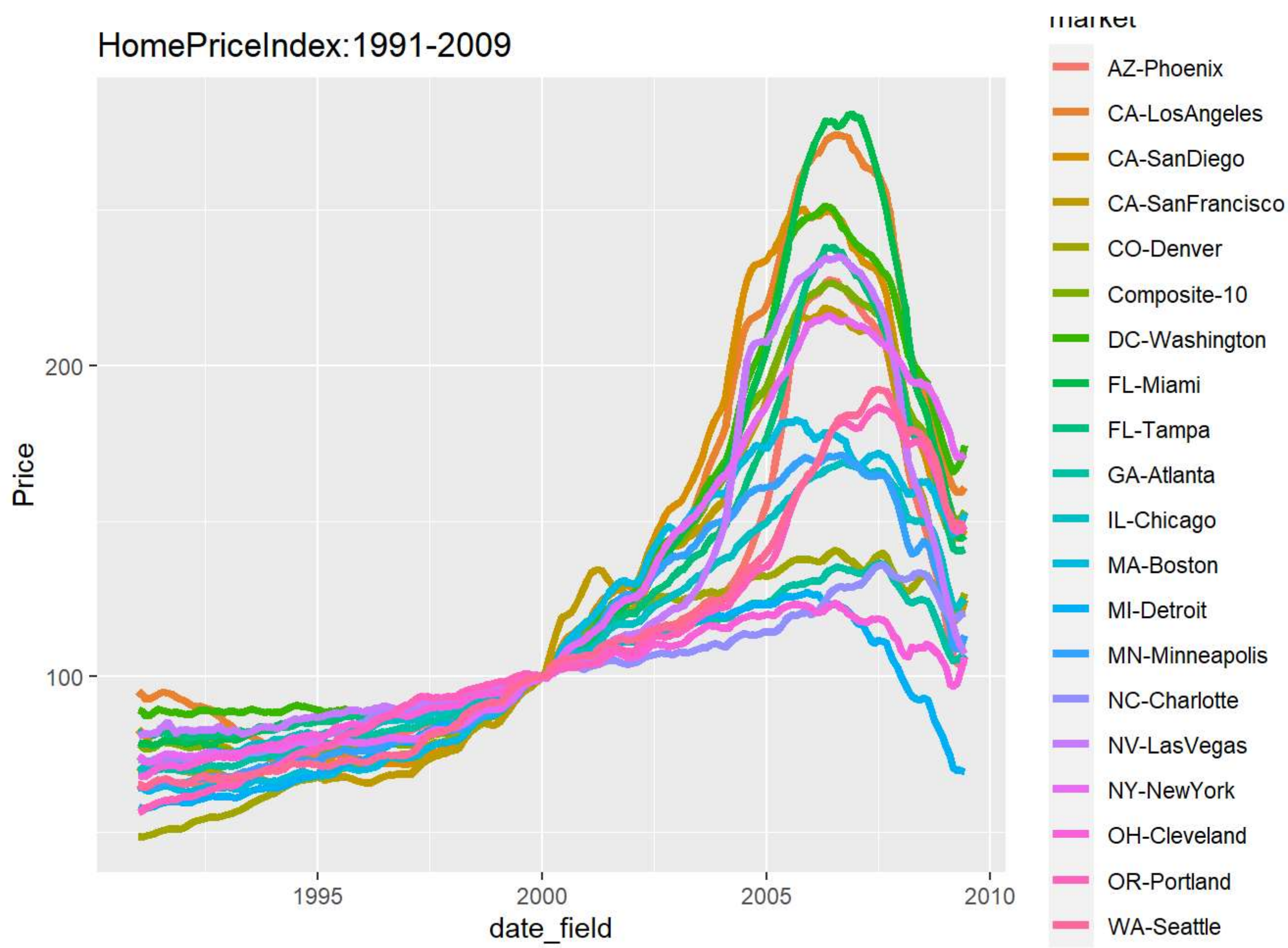
We replace the year and month with a date field.

```
temp=seq(as.Date('1991-01-01'),as.Date('2009-06-01'),by='month')
mydata<-mydata%>%
  mutate(date_field=temp)%>%
  select(-Year,-Month)
head(mydata)
```

```
## # A tibble: 6 x 22
##   Index `AZ-Phoenix` `CA-LosAngeles` `CA-SanDiego` `CA-SanFrancisco` `CO-Denver`
##   <int>        <dbl>           <dbl>         <dbl>              <dbl>       <dbl>
## 1     1         65.3            95.3          83.1               71.2        48.7
## 2     2         65.3            94.1          81.9               70.3        48.7
## 3     3         64.6            92.8          80.9               69.6        48.8
## 4     4         64.4            92.8          80.7               69.5        49.2
## 5     5         64.4            93.4          81.4               70.1        49.5
## 6     6         64.9            94.2          82.2               70.8        50.1
## # ... with 16 more variables: DC-Washington <dbl>, FL-Miami <dbl>,
## #   FL-Tampa <dbl>, GA-Atlanta <dbl>, IL-Chicago <dbl>, MA-Boston <dbl>,
## #   MI-Detroit <dbl>, MN-Minneapolis <dbl>, NC-Charlotte <dbl>,
## #   NV-LasVegas <dbl>, NY-NewYork <dbl>, OH-Cleveland <dbl>, OR-Portland <dbl>,
## #   WA-Seattle <dbl>, Composite-10 <dbl>, date_field <date>
```

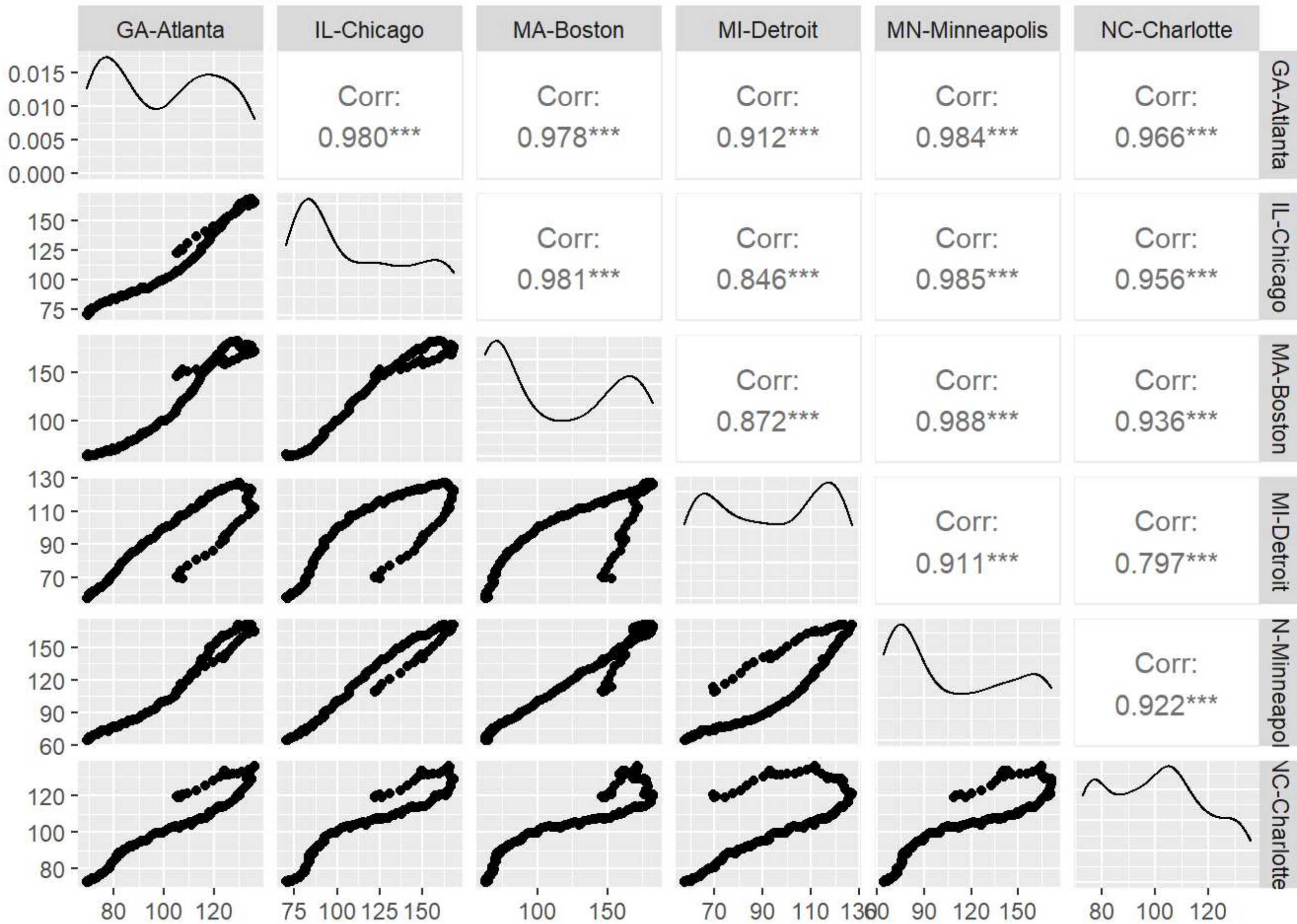
we change the data from wide format to long format so that we can plot price curves by location

```
mydata%>%
  select(-Index)%>%
  gather(-date_field,key="market",value="Price")%>%
  ggplot(aes(x=date_field, y=Price, color=market)) +
geom_line(size=1.5) + ggtitle("HomePriceIndex:1991-2009")
```



we change the data from wide format to long format so that we can plot price curves by location

```
subset<-mydata[,10:15]
ggpairs(subset)
```



We can examine one particular market (Boston) more closely

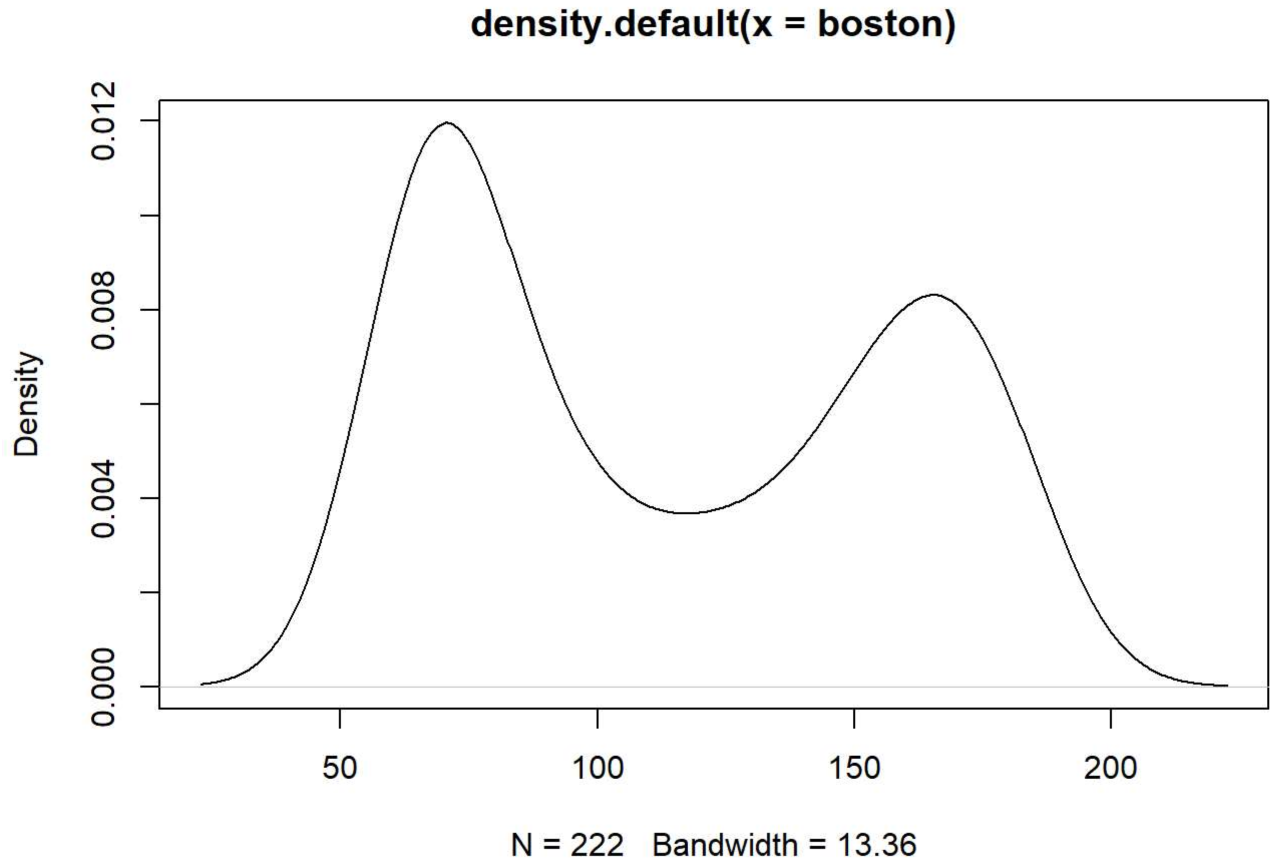
```
boston<-mydata$`MA-Boston`

summary(boston)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   62.94   70.10  102.29  114.18  158.67  182.45
```

```
#standard deviation
sd(boston)
```

```
## [1] 43.72929
```

```
plot(density(boston))
```



Let's examine the relationship between San Francisco Los Angeles more closely.

One way to concisely capture relationship between two random variables is to look at the correlations. It is equivalent to find a simple linear function (or model) like $y_t = \alpha + \beta * x_t + \epsilon_t$ where y_t is the price of SF market at time t, x_t is the LA market.

α is called intercept and β is called slope. In particular, the slope tells us how much SF housing price moves with LA housing prices.

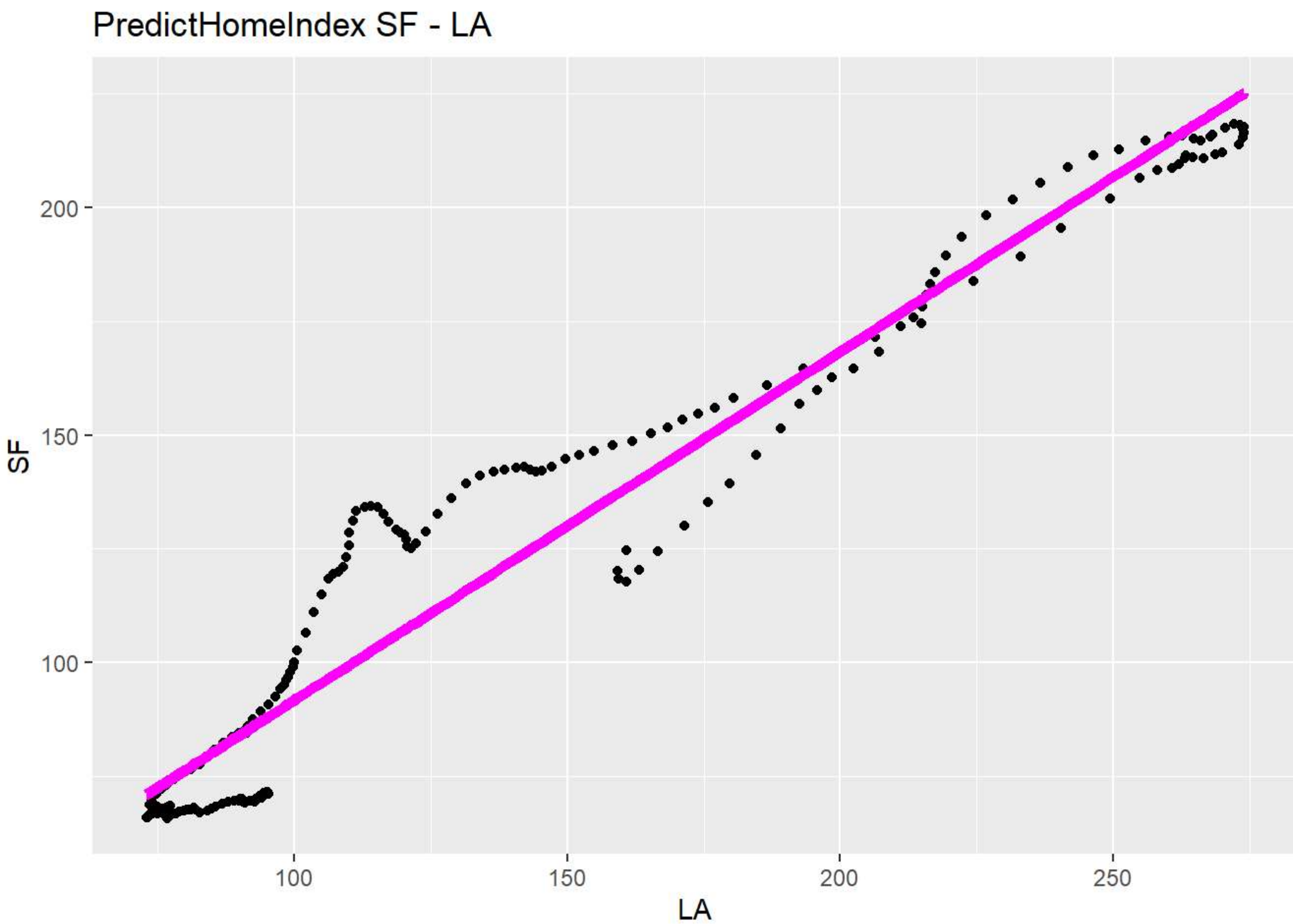
```
CA<-mydata%>%
  select(contains("CA-"))

head(CA)
```

```
## # A tibble: 6 x 3
##   `CA-LosAngeles` `CA-SanDiego` `CA-SanFrancisco`
##             <dbl>         <dbl>             <dbl>
## 1            95.3          83.1              71.2
## 2            94.1          81.9              70.3
## 3            92.8          80.9              69.6
## 4            92.8          80.7              69.5
## 5            93.4          81.4              70.1
## 6            94.2          82.2              70.8
```

```
colnames(CA)<-c("LA","SD","SF")

mymodel<-lm(SF~LA,data=CA)
summary(mymodel)
```

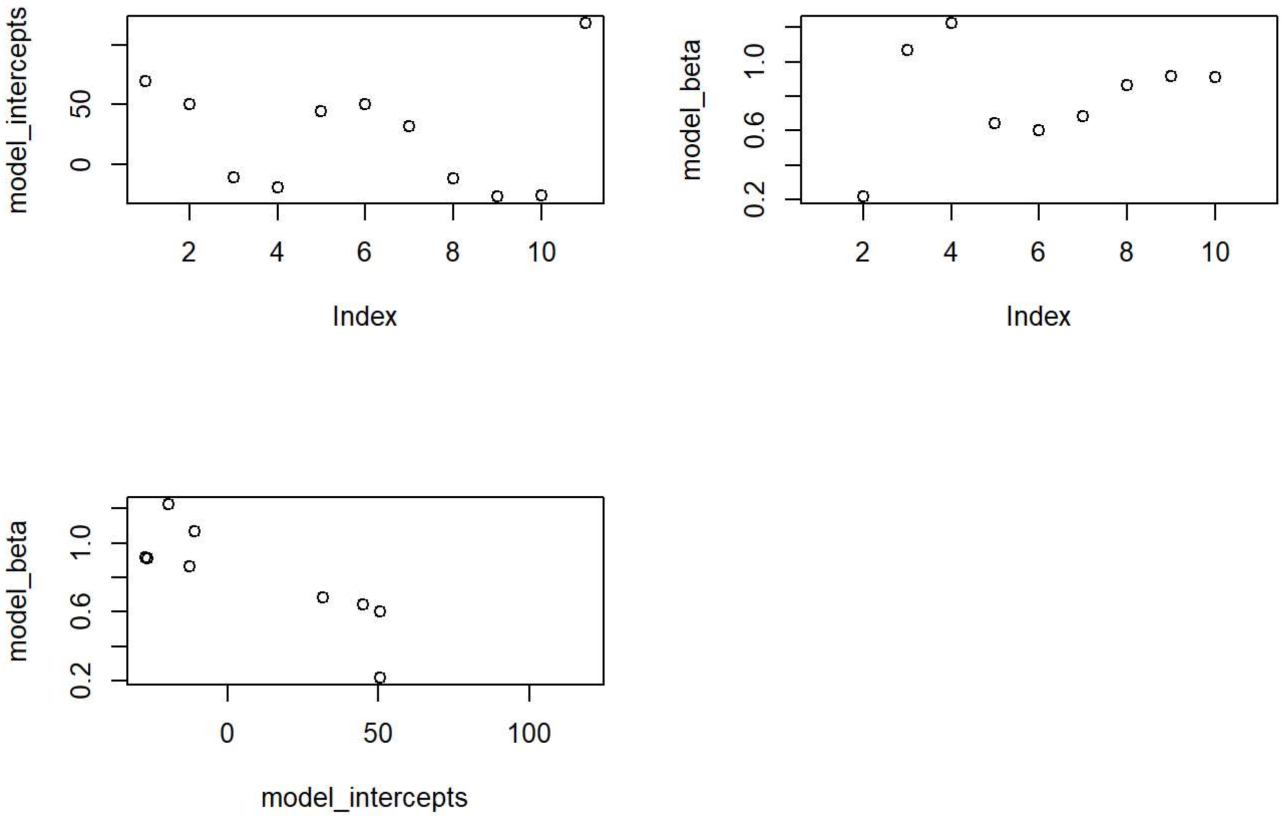
```
##
## Call:
## lm(formula = SF ~ LA, data = CA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.662  -7.739  -3.570   6.133  32.898
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 14.95019    1.88959   7.912 1.23e-13 ***
## LA           0.76735    0.01251  61.358  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.33 on 220 degrees of freedom
## Multiple R-squared:  0.9448, Adjusted R-squared:  0.9445
## F-statistic:  3765 on 1 and 220 DF,  p-value: < 2.2e-16
```

```
CA$pred_sf = predict(mymodel,data=CA)

ggplot(data=CA, aes(x = LA)) +
geom_point(aes(y = SF)) +
geom_line(aes(y = pred_sf), color='Magenta', size=2) +
ggtitle("PredictHomeIndex SF - LA")
```



Final example, we want to see if the relationship between SF and LA changes over time. Although not applicable, but this is the same concept as in pair trade in stock. If you have two stocks A and B and you believe their price relationship in the long-term should be stable. If you then a significant deviation of one stock's price, you could buy or sell, in anticipation of the relationship going back to normal in the near future.

```
mydata<-mydata%>%
  select(`CA-SanFrancisco`,`CA-LosAngeles`,date_field)%>%
  rename(SF=`CA-SanFrancisco`,LA=`CA-LosAngeles`)


model_intercepts<-numeric(11)
model_beta<-numeric(11)
for (i in 1:11){
  temp<-mydata[(i-1)*20+1:i*20,]
  mymodel<-lm(SF~LA,data=temp)
  model_intercepts[i]<-mymodel$coefficients[1]
  model_beta[i]<-mymodel$coefficients[2]
}


par(mfrow=c(2,2))
plot(model_intercepts)
plot(model_beta)
plot(model_intercepts,model_beta)
```


model_intercepts
0
50
2
4
6
8
10
Index
model_beta
0.2
0.6
1.0
2
4
6
8
10
Index
model_beta
0.2
0.6
1.0
0
50
100
model_intercepts