House Property Sales

Oscar Monroy

6/16/2022

Bringing in the Data

```r
hps <- read.csv("ma_lga_12345.csv")
head(hps, 10)
```

```
##      saledate     MA  type bedrooms
## 1  30/09/2007 441854 house        2
## 2  31/12/2007 441854 house        2
## 3  31/03/2008 441854 house        2
## 4  30/06/2008 441854 house        2
## 5  30/09/2008 451583 house        2
## 6  31/12/2008 440256 house        2
## 7  31/03/2009 442566 house        2
## 8  30/06/2009 446113 house        2
## 9  30/09/2009 440123 house        2
## 10 31/12/2009 442131 house        2
```

```r
summary(hps) # Everything seems to be in order
```

```
##        saledate          MA              type        bedrooms
##  30/06/2008:  7   Min.   : 316751   house:200   Min.   :1.000
##  30/06/2009:  7   1st Qu.: 427740   unit :147   1st Qu.:2.000
##  30/06/2010:  7   Median : 507744               Median :3.000
##  30/06/2011:  7   Mean   : 548132               Mean   :2.867
##  30/06/2012:  7   3rd Qu.: 627516               3rd Qu.:4.000
##  30/06/2013:  7   Max.   :1017752               Max.   :5.000
##  (Other)   :305
```

```r
sum(duplicated(hps)) # No duplicate data
```

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

```r
hps$saledate <- as.Date(hps$saledate, format = "%d/%m/%Y")
hps$bedrooms <- as.factor(hps$bedrooms)
head(hps, 10)
```

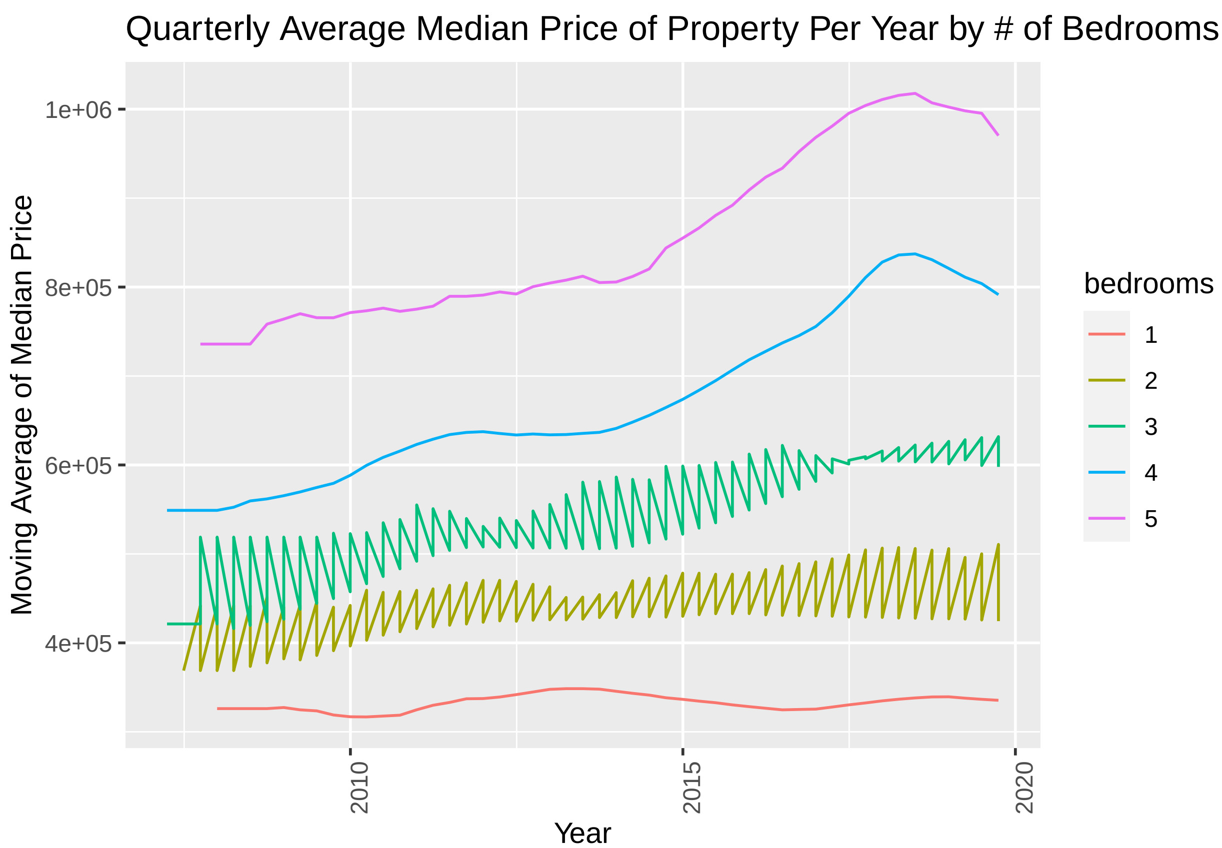
```
##      saledate     MA  type bedrooms
## 1  2007-09-30 441854 house        2
## 2  2007-12-31 441854 house        2
## 3  2008-03-31 441854 house        2
## 4  2008-06-30 441854 house        2
## 5  2008-09-30 451583 house        2
## 6  2008-12-31 440256 house        2
## 7  2009-03-31 442566 house        2
## 8  2009-06-30 446113 house        2
## 9  2009-09-30 440123 house        2
## 10 2009-12-31 442131 house        2
```

Plotting the Current Quarterly Values

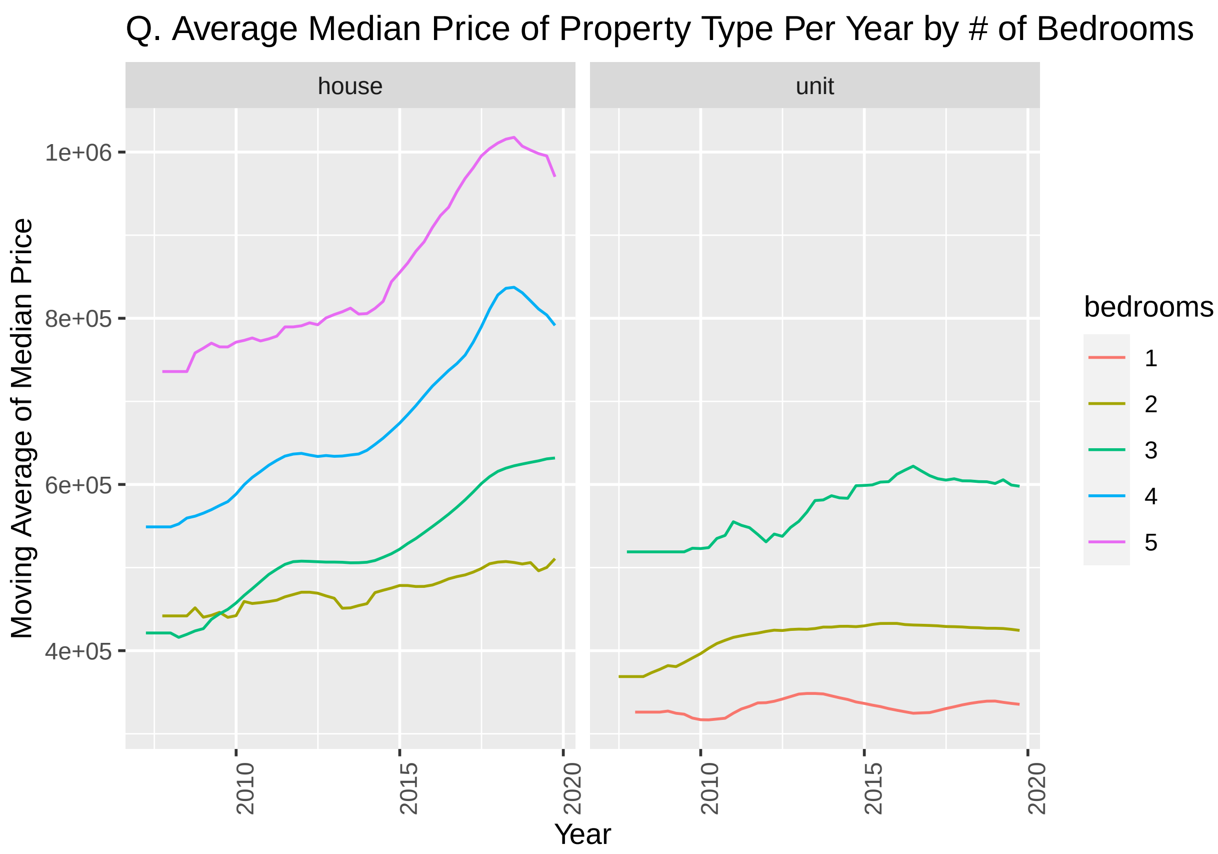
```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

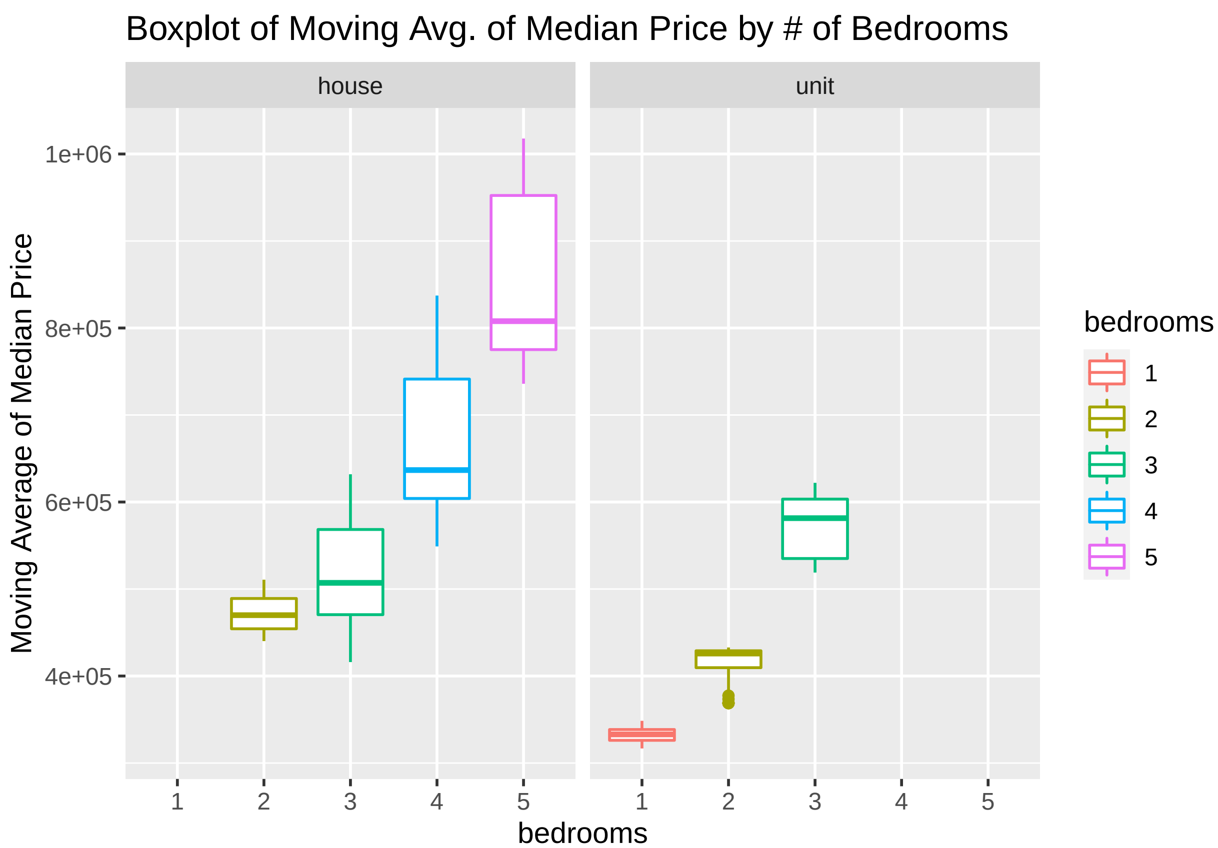
```r
ggplot(data = hps, aes(x = saledate, y = MA, group = bedrooms, color = bedrooms)) +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ylab("Moving Average of Median Price") +
  xlab("Year") +
  ggtitle("Quarterly Average Median Price of Property Per Year by # of Bedrooms")
```




```r
ggplot(data = hps, aes(x = saledate, y = MA, group = bedrooms, color = bedrooms)) +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  facet_grid(~ type) +
  ylab("Moving Average of Median Price") +
  xlab("Year") +
  ggtitle("Q. Average Median Price of Property Type Per Year by # of Bedrooms")
```




```r
ggplot(data = hps, aes(x = bedrooms, y = MA, color = bedrooms)) +
  geom_boxplot() +
  facet_grid(~ type) +
  ylab("Moving Average of Median Price") +
  ggtitle("Boxplot of Moving Avg. of Median Price by # of Bedrooms")
```



Implementing the Time Series Function for Property Types

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.3

##
## 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
unit_1br <- hps %>% filter(bedrooms == 1, type == "unit")
unit_1br <- ts(unit_1br[,2], start = c(2007, 4), frequency = 4)

unit_2br <- hps %>% filter(bedrooms == 2, type == "unit")
```

```r
unit_2br <- ts(unit_2br[,2], start = c(2007, 2), frequency = 4)

unit_3br <- hps %>% filter(bedrooms == 3, type == "unit")
unit_3br <- ts(unit_3br[,2], start = c(2007, 3), frequency = 4)

house_2br <- hps %>% filter(bedrooms == 2, type == "house")
house_2br <- ts(house_2br[,2], start = c(2007, 3), frequency = 4)

house_3br <- hps %>% filter(bedrooms == 3, type == "house")
house_3br <- ts(house_3br[,2], start = c(2007, 1), frequency = 4)

house_4br <- hps %>% filter(bedrooms == 4, type == "house")
house_4br <- ts(house_4br[,2], start = c(2007, 1), frequency = 4)

house_5br <- hps %>% filter(bedrooms == 5, type == "house")
house_5br <- ts(house_5br[,2], start = c(2007, 3), frequency = 4)
```

Forecasting Models

```r
# install.packages("forecast")
# install.packages("quantmod")
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.6.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
arima_u1 <- auto.arima(unit_1br)
summary(arima_u1)
```

```
## Series: unit_1br
## ARIMA(1,1,0)
##
## Coefficients:
##          ar1
##       0.7429
## s.e.  0.0925
##
## sigma^2 estimated as 2238012:  log likelihood=-410.16
## AIC=824.32   AICc=824.59   BIC=828.02
##
## Training set error measures:
##                    ME   RMSE      MAE        MPE      MAPE      MASE
## Training set 39.93949 1464.5 1005.183 0.01412482 0.3036909 0.1434905
##                     ACF1
## Training set -0.07697338
```

```r
u1_f <- forecast(arima_u1, 13)
u1_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       334630.4 332713.2 336547.6 331698.2 337562.5
## 2020 Q1       334022.2 330169.8 337874.6 328130.5 339913.9
## 2020 Q2       333570.4 327722.7 339418.2 324627.0 342513.8
## 2020 Q3       333234.8 325419.1 341050.5 321281.7 345187.9
## 2020 Q4       332985.5 323271.0 342699.9 318128.5 347842.4
## 2021 Q1       332800.3 321274.2 344326.3 315172.7 350427.8
## 2021 Q2       332662.7 319417.7 345907.6 312406.2 352919.1
## 2021 Q3       332560.5 317688.0 347432.9 309815.0 355305.9
## 2021 Q4       332484.5 316071.4 348897.6 307382.9 357586.2
## 2022 Q1       332428.1 314555.2 350301.0 305093.9 359762.3
## 2022 Q2       332386.2 313127.8 351644.7 302932.9 361839.5
## 2022 Q3       332355.1 311778.7 352931.5 300886.2 363824.0
## 2022 Q4       332332.0 310499.1 354164.9 298941.4 365722.5
```

```r
arima_u2 <- auto.arima(unit_2br)
summary(arima_u2)
```

```
## Series: unit_2br
## ARIMA(0,2,1)
##
## Coefficients:
##           ma1
##       -0.4607
## s.e.   0.1415
##
## sigma^2 estimated as 2243076:  log likelihood=-418.62
## AIC=841.24   AICc=841.51   BIC=844.99
##
## Training set error measures:
##                     ME     RMSE      MAE          MPE      MAPE      MASE
## Training set -43.95493 1452.065 963.0706 -0.006027837 0.2377975 0.1577489
##                    ACF1
## Training set 0.01884361
```

```r
u2_f <- forecast(arima_u2, 12)
u2_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       423432.6 421513.3 425352.0 420497.2 426368.1
## 2020 Q1       422453.3 418930.1 425976.5 417065.0 427841.5
## 2020 Q2       421473.9 416151.4 426796.5 413333.8 429614.0
## 2020 Q3       420494.6 413175.0 427814.2 409300.2 431688.9
## 2020 Q4       419515.2 410012.7 429017.7 404982.4 434048.0
## 2021 Q1       418535.8 406676.9 430394.8 400399.2 436672.5
## 2021 Q2       417556.5 403178.4 431934.6 395567.1 439545.9
## 2021 Q3       416577.1 399526.4 433627.9 390500.3 442654.0
## 2021 Q4       415597.8 395728.9 435466.6 385211.0 445984.5
## 2022 Q1       414618.4 391792.9 437443.8 379709.9 449526.9
```

```
## 2022 Q2       413639.0 387724.5 439553.5 374006.2 453271.8
## 2022 Q3       412659.7 383529.0 441790.3 368108.2 457211.2
```

```r
arima_u3 <- auto.arima(unit_3br)
summary(arima_u3)
```

```
## Series: unit_3br
## ARIMA(2,1,0)(2,0,0)[4]
##
## Coefficients:
##          ar1     ar2     sar1     sar2
##       0.3018  0.4231  -0.5448  -0.3209
## s.e.  0.1326  0.1338   0.1428   0.1352
##
## sigma^2 estimated as 26434054:  log likelihood=-477.03
## AIC=964.07   AICc=965.5   BIC=973.42
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE      MAPE      MASE
## Training set 793.4379 4872.033 3464.103 0.1435795 0.6040255 0.2840096
##                     ACF1
## Training set -0.03468343
```

```r
u3_f <- forecast(arima_u3, 13)
u3_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       596417.9 589828.9 603006.8 586340.9 606494.8
## 2020 Q1       592558.5 581742.2 603374.7 576016.5 609100.5
## 2020 Q2       594385.1 578255.5 610514.7 569717.0 619053.2
## 2020 Q3       594040.6 572794.0 615287.1 561546.7 626534.4
## 2020 Q4       594364.2 569846.2 618882.2 556867.1 631861.3
## 2021 Q1       594182.4 566642.1 621722.8 552063.1 636301.8
## 2021 Q2       594463.6 564437.5 624489.7 548542.7 640384.5
## 2021 Q3       594533.8 562229.4 626838.3 545128.5 643939.2
## 2021 Q4       594336.6 560046.8 628626.4 541894.9 646778.3
## 2022 Q1       595278.5 559158.1 631398.8 540037.1 650519.8
## 2022 Q2       594212.0 556415.1 632008.8 536406.7 652017.3
## 2022 Q3       594018.2 554652.1 633384.2 533813.0 654223.4
## 2022 Q4       593803.0 552616.4 634989.6 530813.6 656792.5
```

```r
arima_h2 <- auto.arima(house_2br)
summary(arima_h2)
```

```
## Series: house_2br
## ARIMA(0,1,0)(0,0,1)[4] with drift
##
## Coefficients:
##          sma1      drift
##       -0.3693  1440.4390
## s.e.   0.1853   464.6175
##
```

```
## sigma^2 estimated as 24513602:  log likelihood=-475.73
## AIC=957.46   AICc=958.01   BIC=963.07
##
## Training set error measures:
##                     ME     RMSE      MAE         MPE      MAPE      MASE
## Training set -5.730028 4797.163 3141.751 -0.01310796 0.6725938 0.3246236
##                     ACF1
## Training set -0.06523639
```

```r
h2_f <- forecast(arima_h2, 13)
h2_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       511943.7 505598.6 518288.9 502239.7 521647.8
## 2020 Q1       517522.0 508548.6 526495.3 503798.4 531245.5
## 2020 Q2       518183.2 507193.2 529173.3 501375.4 534991.1
## 2020 Q3       516466.9 503776.6 529157.1 497058.8 535874.9
## 2020 Q4       517907.3 504601.0 531213.6 497557.0 538257.6
## 2021 Q1       519347.7 505452.6 533242.9 498097.0 540598.5
## 2021 Q2       520788.2 506328.2 535248.1 498673.6 542902.8
## 2021 Q3       522228.6 507225.1 537232.2 499282.7 545174.6
## 2021 Q4       523669.0 508140.9 539197.2 499920.8 547417.3
## 2022 Q1       525109.5 509073.9 541145.0 500585.2 549633.8
## 2022 Q2       526549.9 510022.5 543077.3 501273.4 551826.4
## 2022 Q3       527990.4 510985.3 544995.4 501983.4 553997.3
## 2022 Q4       529430.8 511961.2 546900.4 502713.3 556148.3
```

```r
arima_h3 <- auto.arima(house_3br)
summary(arima_h3)
```

```
## Series: house_3br
## ARIMA(1,1,0)(2,0,1)[4] with drift
##
## Coefficients:
##          ar1    sar1     sar2     sma1      drift
##       0.8558  0.2403  -0.3121  -0.8628  4285.7969
## s.e.  0.0891  0.2145   0.2077   0.2314   370.4374
##
## sigma^2 estimated as 3398320:  log likelihood=-447.6
## AIC=907.19   AICc=909.14   BIC=918.66
##
## Training set error measures:
##                     ME     RMSE      MAE         MPE      MAPE       MASE
## Training set -58.09083 1731.623 1236.409 -0.01711588 0.2560346 0.06786317
##                     ACF1
## Training set -0.09240477
```

```r
h3_f <- forecast(arima_h3, 13)
h3_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       633627.8 631258.1 635997.5 630003.7 637251.9
```

```
## 2020 Q1       635888.2 630895.1 640881.3 628251.9 643524.5
## 2020 Q2       638021.4 630119.5 645923.2 625936.5 650106.2
## 2020 Q3       642055.7 631089.5 653021.9 625284.3 658827.1
## 2020 Q4       646318.1 633085.2 659551.0 626080.2 666556.1
## 2021 Q1       650780.3 635698.3 665862.4 627714.4 673846.3
## 2021 Q2       655047.8 638383.0 671712.6 629561.2 680534.4
## 2021 Q3       660200.6 642139.7 678261.4 632578.9 687822.2
## 2021 Q4       665203.6 646246.2 684161.0 636210.7 694196.5
## 2022 Q1       670106.0 650566.7 689645.3 640223.2 699988.8
## 2022 Q2       675009.7 655094.1 694925.3 644551.4 705468.0
## 2022 Q3       679539.8 659384.8 699694.8 648715.3 710364.3
## 2022 Q4       683968.9 663641.0 704296.7 652880.1 715057.7
```

```
arima_h4 <- auto.arima(house_4br)
summary(arima_h4)
```

```
## Series: house_4br
## ARIMA(2,1,0)
##
## Coefficients:
##          ar1      ar2
##       1.4048  -0.4918
## s.e.  0.1244   0.1247
##
## sigma^2 estimated as 7000906:  log likelihood=-465.31
## AIC=936.62   AICc=937.14   BIC=942.35
##
## Training set error measures:
##                   ME     RMSE     MAE        MPE      MAPE       MASE
## Training set 270.726 2566.922 1880.71 0.05037792 0.2779408 0.07043778
##                      ACF1
## Training set -0.006186785
```

```
h4_f <- forecast(arima_h4, 13)
h4_f
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2019 Q4       777455.1 774064.3 780846.0 772269.2 782641.1
## 2020 Q1       763938.9 755107.6 772770.1 750432.7 777445.1
## 2020 Q2       751832.9 735969.4 767696.5 727571.7 776094.1
## 2020 Q3       741474.7 717562.5 765386.8 704904.2 778045.2
## 2020 Q4       732878.0 700355.1 765400.8 683138.6 782617.3
## 2021 Q1       725896.1 684532.8 767259.5 662636.4 789155.9
## 2021 Q2       720316.5 670112.9 770520.1 643536.7 797096.3
## 2021 Q3       715912.4 657020.0 774804.7 625844.3 805980.4
## 2021 Q4       712469.8 645133.6 779806.1 609487.9 815451.8
## 2022 Q1       709800.0 634317.0 785283.1 594358.6 825241.4
## 2022 Q2       707742.7 624434.2 791051.3 580333.3 835152.2
## 2022 Q3       706165.9 615358.8 796972.9 567288.4 845043.3
## 2022 Q4       704962.5 606977.6 802947.5 555107.5 854817.5
```

```r
arima_h5 <- auto.arima(house_5br)
summary(arima_h5)
```

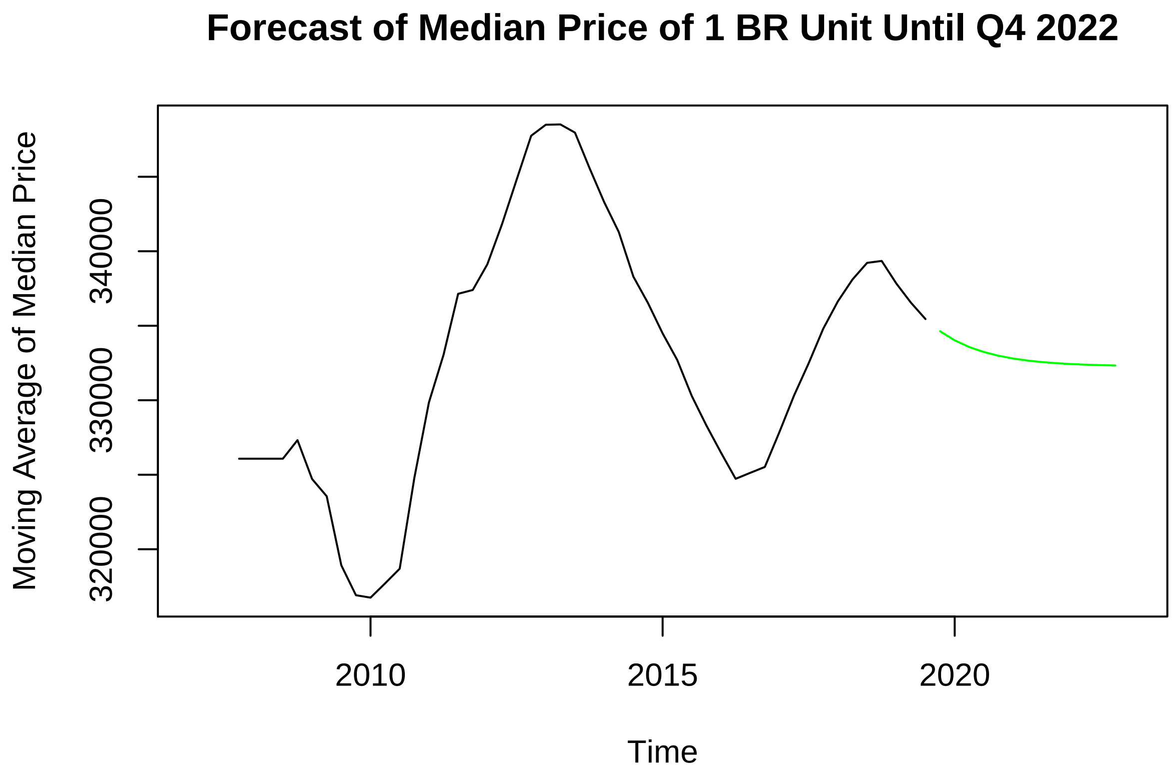
```
## Series: house_5br
## ARIMA(1,1,1)
##
## Coefficients:
##          ar1      ma1
##       0.9000  -0.4386
## s.e.  0.0788   0.1666
##
## sigma^2 estimated as 51615008:  log likelihood=-493.74
## AIC=993.48   AICc=994.03   BIC=999.1
##
## Training set error measures:
##                    ME     RMSE      MAE        MPE      MAPE      MASE
## Training set 398.9056 6960.956 4467.352 0.06313621 0.5264191 0.1685397
##                     ACF1
## Training set -0.01247393
```

```r
h5_f <- forecast(arima_h5, 13)
h5_f
```

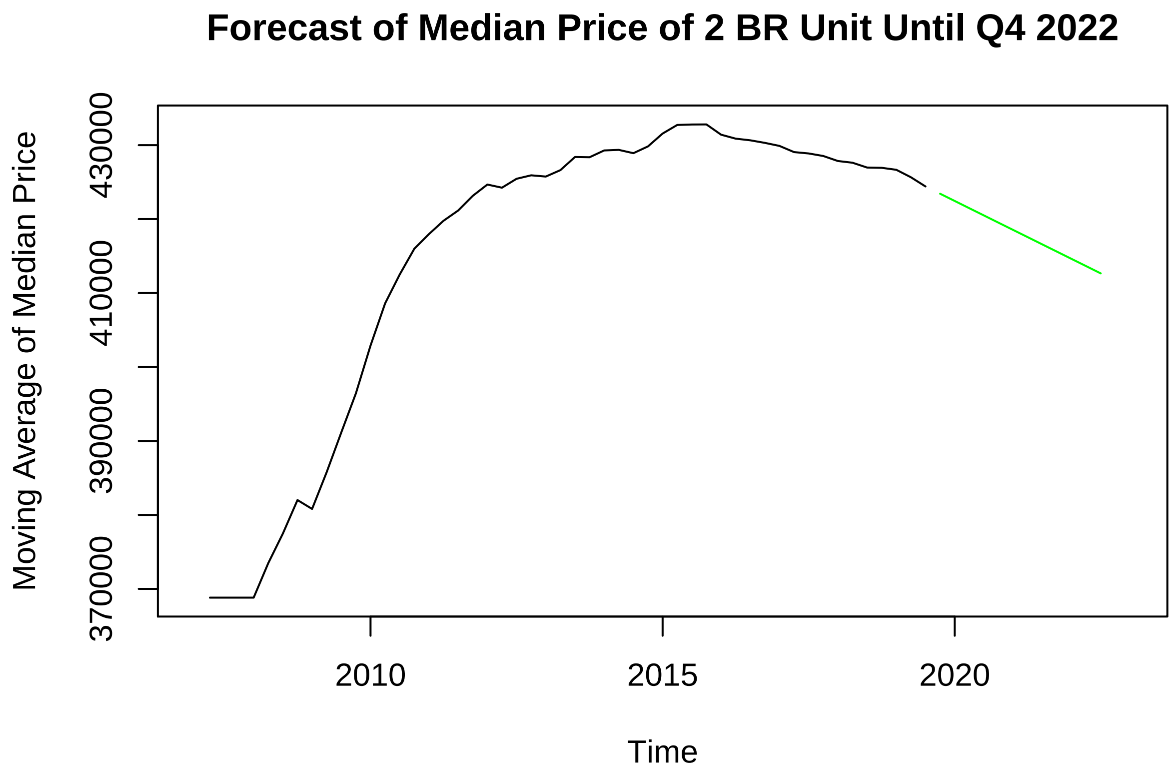
```
##         Point Forecast    Lo 80    Hi 80    Lo 95     Hi 95
## 2019 Q4       957442.4 948235.3 966649.5 943361.3  971523.5
## 2020 Q1       945899.5 929595.6 962203.4 920964.8  970834.2
## 2020 Q2       935510.9 911754.4 959267.4 899178.5  971843.4
## 2020 Q3       926161.3 894638.7 957684.0 877951.6  974371.1
## 2020 Q4       917746.7 878238.4 957255.1 857324.0  978169.5
## 2021 Q1       910173.7 862543.1 957804.2 837329.1  983018.3
## 2021 Q2       903358.0 847535.2 959180.8 817984.4  988731.6
## 2021 Q3       897223.9 833190.7 961257.1 799293.6  995154.2
## 2021 Q4       891703.3 819482.0 963924.5 781250.4 1002156.1
## 2022 Q1       886734.8 806378.9 967090.6 763841.0 1009628.5
## 2022 Q2       882263.1 793849.9 970676.3 747046.8 1017479.5
## 2022 Q3       878238.7 781863.6 974613.8 730845.7 1025631.6
## 2022 Q4       874616.7 770388.8 978844.6 715213.9 1034019.5
```

Graphing out the Forecast Models

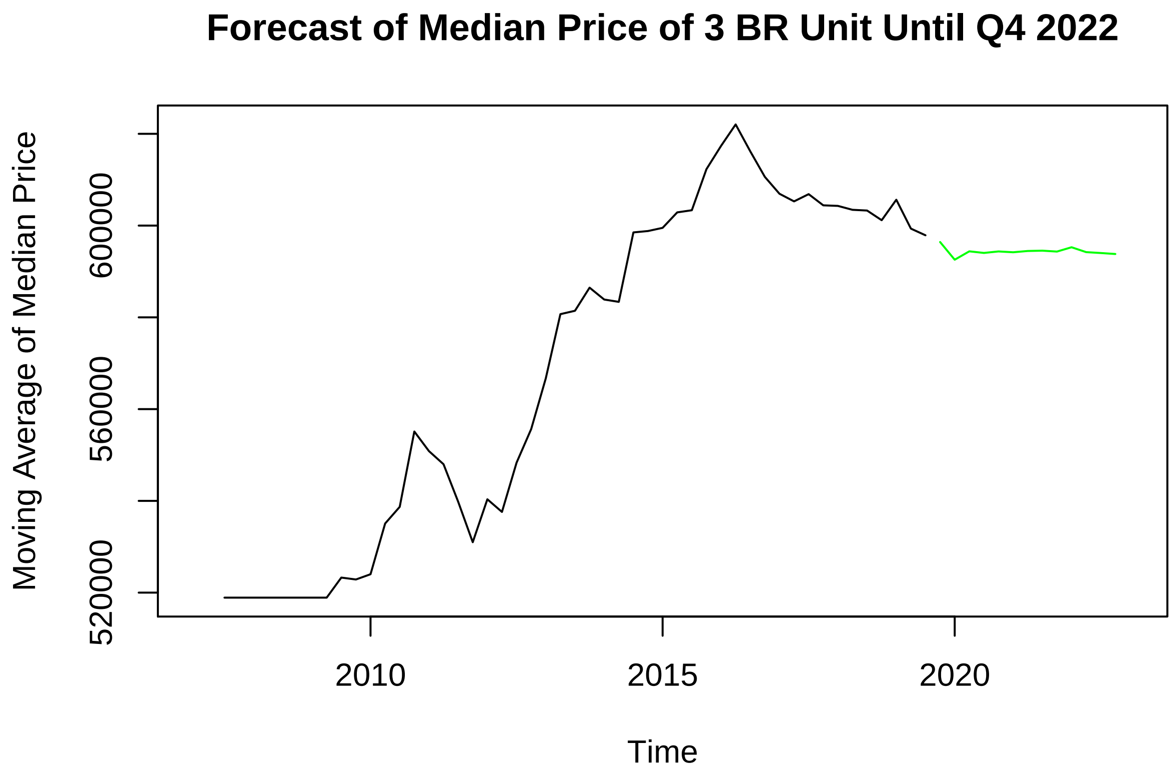
```r
plot(unit_1br, xlim = c(2007, 2023), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 1 BR Unit Until Q4 2022")
lines(u1_f$mean, col = "green")
```



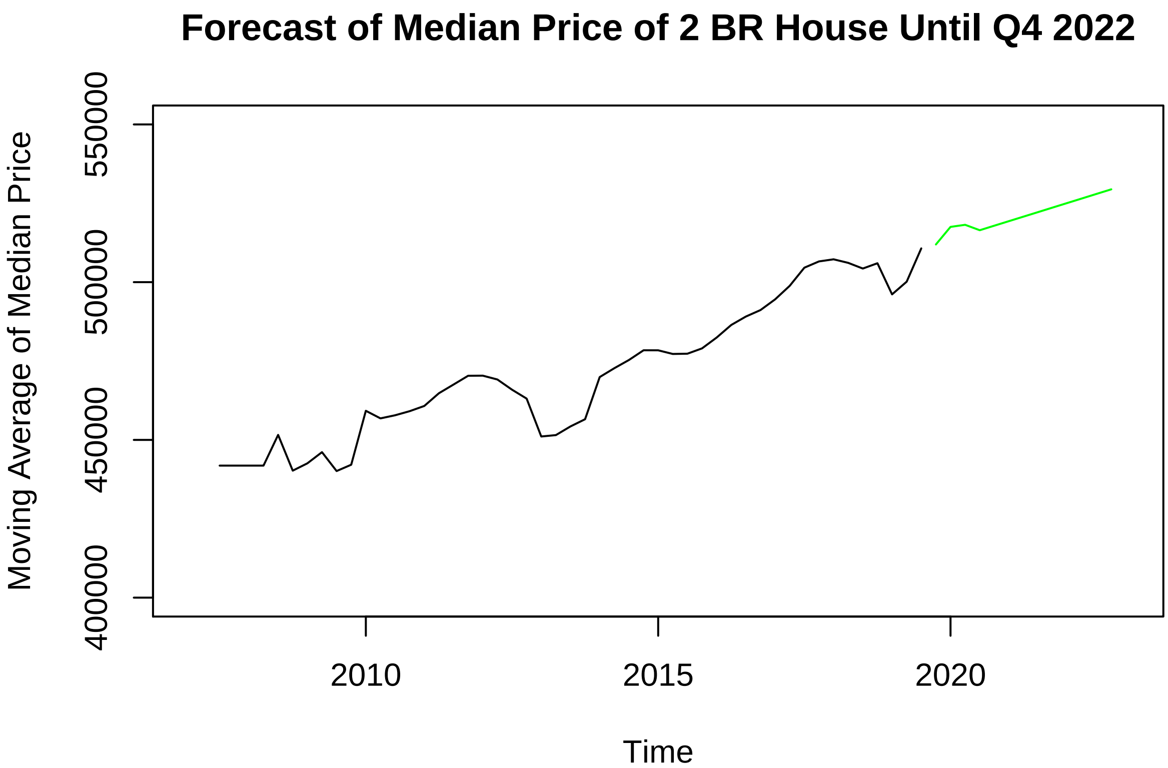

```r
plot(unit_2br, xlim = c(2007, 2023), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 2 BR Unit Until Q4 2022")
lines(u2_f$mean, col = "green")
```



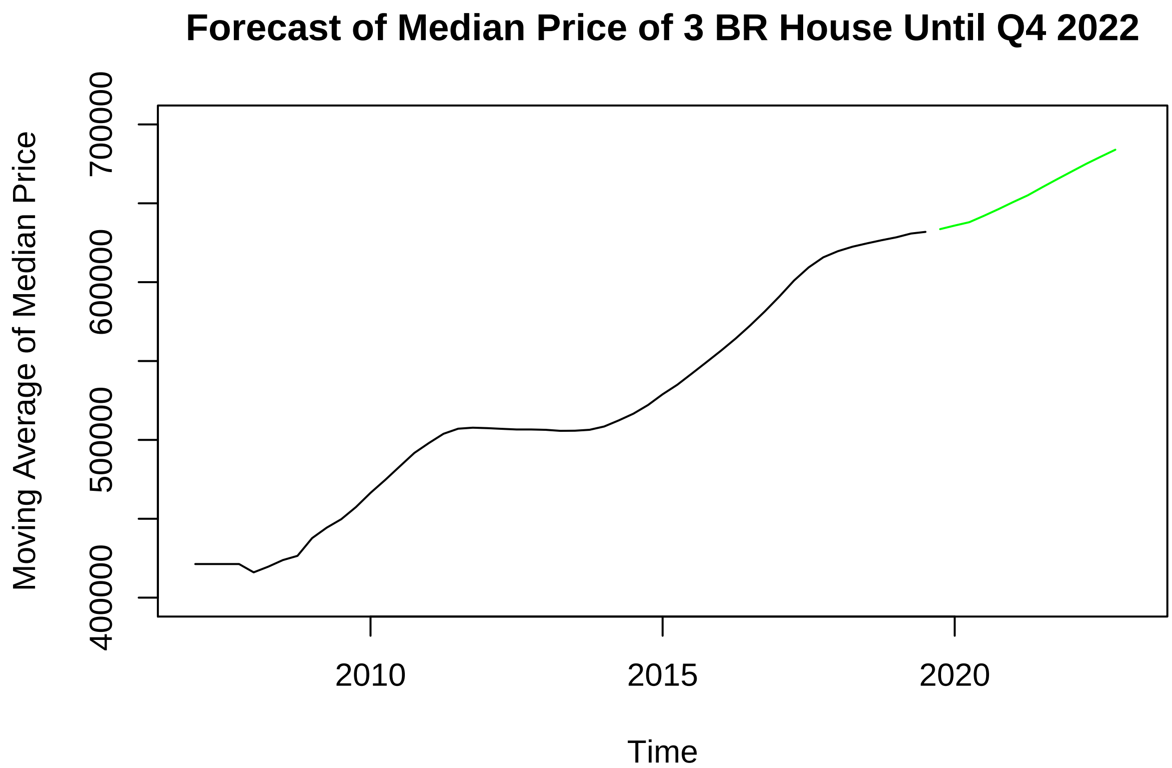

```
plot(unit_3br, xlim = c(2007, 2023), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 3 BR Unit Until Q4 2022")
lines(u3_f$mean, col = "green")
```



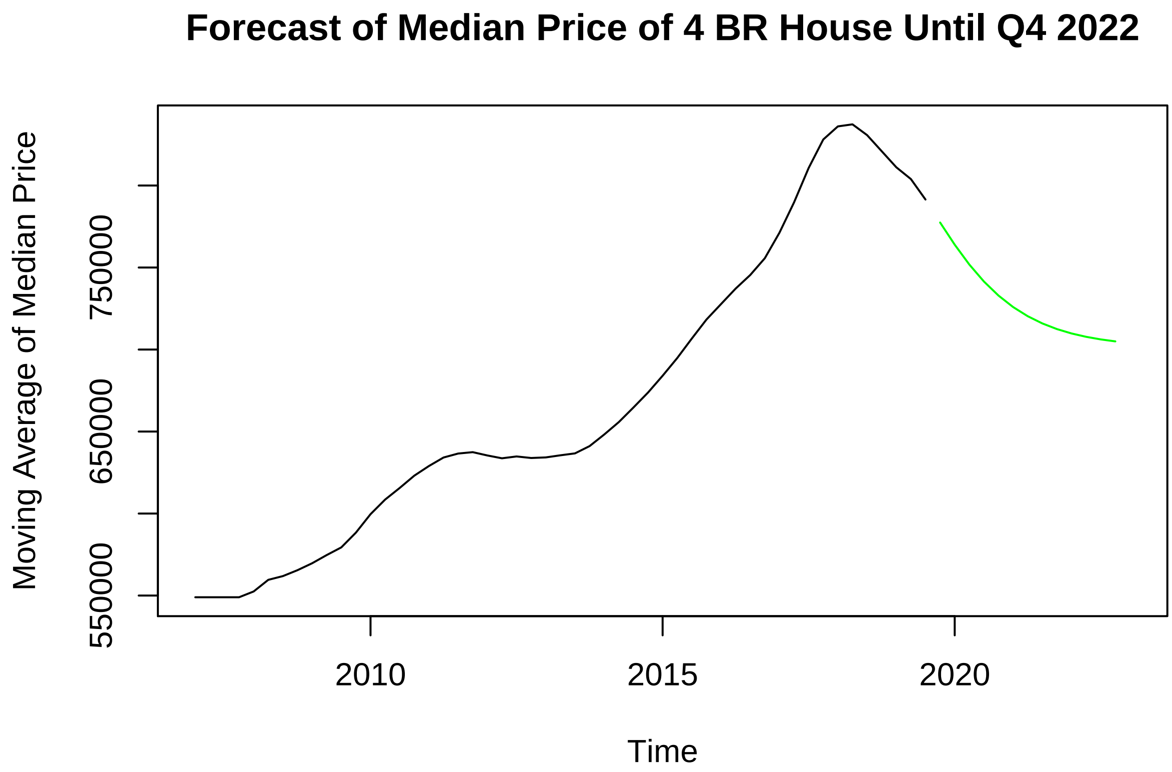

```
plot(house_2br, xlim = c(2007, 2023),
     ylim = c(400000, 550000), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 2 BR House Until Q4 2022")
lines(h2_f$mean, col = "green")
```



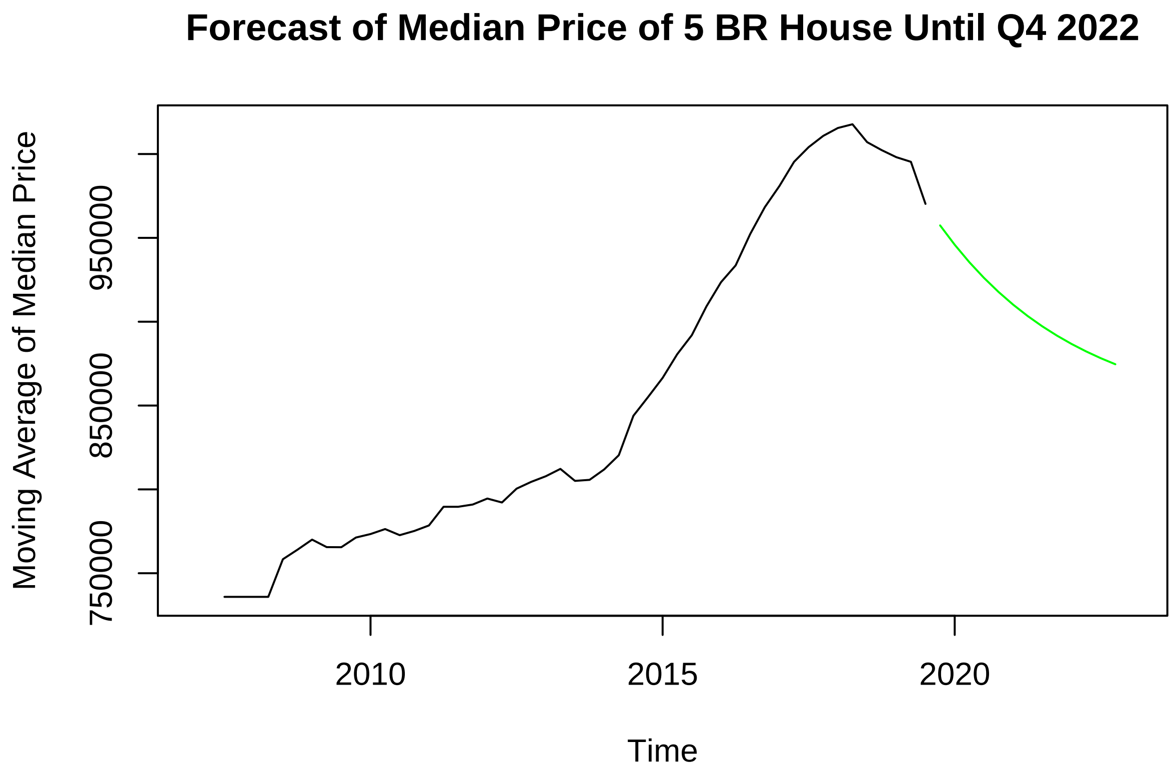

```r
plot(house_3br, xlim = c(2007, 2023),
     ylim = c(400000, 700000), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 3 BR House Until Q4 2022")
lines(h3_f$mean, col = "green")
```




```r
plot(house_4br, xlim = c(2007, 2023), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 4 BR House Until Q4 2022")
lines(h4_f$mean, col = "green")
```




```r
plot(house_5br, xlim = c(2007, 2023), ylab = "Moving Average of Median Price",
     main = "Forecast of Median Price of 5 BR House Until Q4 2022")
lines(h5_f$mean, col = "green")
```


Forecast of Median Price of 5 BR House Until Q4 2022
Moving Average of Median Price
750000
850000
950000
2010
2015
2020
Time