BUAN6357_Project_Zhang

Xiaojia Zhang

11/17/2020

Data Information

1. Property sales data for the 2007 – 2019 period for Australian Capital Territory.
2. Column name, data type, and description:

- Date Sold (Date time): date on which this property was sold.
- Postcode (Integer): 4-digit postcode of the suburb where the property was sold.
- Price (Integer): price for which the property was sold.
- Property Type (String): house or unit
- Bedrooms (Integer): number of bedrooms

Hypothesis Statement

- Null Hypothese(H0): the price of property will not change in the future 2 years.
- Alternative Hypothesis(Ha): the price of property will increase in the future 2 years.

```
library(pacman)
pacman::p_load(fpp2, fpp3, patchwork, purrr, feasts, forecast, ggplot2, tsibble, dplyr, lubridate, ggfo
options(scipen = 999)
options(digits=2)
set.seed(123)
```

Data Preparation

```
# Load raw data and quarterly data
sales <- read.csv("property_sales.csv")
mydata <- read.csv("property_quarterly_sales.csv")
```

- The datatype for 'datesold' column is factor, I will convert it to datetime later.
- Number of bedrooms might be 0. Taking studio of unit into account.

```
# Structure of the data frames
str(sales)
```

```
## 'data.frame':    29580 obs. of  5 variables:
##  $ datesold    : Factor w/ 3582 levels "1/1/16 0:00",..: 1385 1319 1695 1714 1545 1953 1878 1940 217
##  $ postcode    : int  2607 2906 2905 2905 2906 2905 2607 2606 2902 2906 ...
##  $ price       : int  525000 290000 328000 380000 310000 465000 399000 1530000 359000 320000 ...
##  $ propertyType: Factor w/ 2 levels "house","unit": 1 1 1 1 1 1 1 1 1 1 ...
##  $ bedrooms    : int  4 3 3 4 3 4 3 4 3 3 ...
```

```
summary(sales)
```

```
##           datesold        postcode        price          propertyType
##  10/28/17 0:00:   50   Min.   :2600   Min.   :  56500   house:24552
##  11/18/17 0:00:   39   1st Qu.:2607   1st Qu.: 440000   unit : 5028
##  3/24/18 0:00 :   38   Median :2615   Median : 550000
##  11/11/17 0:00:   37   Mean   :2730   Mean   : 609736
##  4/8/17 0:00  :   37   3rd Qu.:2905   3rd Qu.: 705000
##  2/24/18 0:00 :   35   Max.   :2914   Max.   :8000000
##  (Other)      :29344
##     bedrooms
##  Min.   :0.0
##  1st Qu.:3.0
##  Median :3.0
##  Mean   :3.3
##  3rd Qu.:4.0
##  Max.   :5.0
##
```

```
str(mydata)
```

```
## 'data.frame':    347 obs. of  4 variables:
##  $ saledate: Factor w/ 51 levels "30/06/2007","30/06/2008",..: 14 40 28 2 15 41 29 3 16 42 ...
##  $ MA      : int  441854 441854 441854 441854 451583 440256 442566 446113 440123 442131 ...
##  $ type    : Factor w/ 2 levels "house","unit": 1 1 1 1 1 1 1 1 1 1 ...
##  $ bedrooms: int  2 2 2 2 2 2 2 2 2 2 ...
```

```
summary(mydata)
```

```
##        saledate          MA             type        bedrooms
##  30/06/2008:  7   Min.   : 316751   house:200   Min.   :1.0
##  30/06/2009:  7   1st Qu.: 427740   unit :147   1st Qu.:2.0
##  30/06/2010:  7   Median : 507744               Median :3.0
##  30/06/2011:  7   Mean   : 548132               Mean   :2.9
##  30/06/2012:  7   3rd Qu.: 627516               3rd Qu.:4.0
##  30/06/2013:  7   Max.   :1017752               Max.   :5.0
##  (Other)   :305
```

```
# Check missing values -> no missing value
miss_value <- length(which(is.na(sales)))
miss_value
```

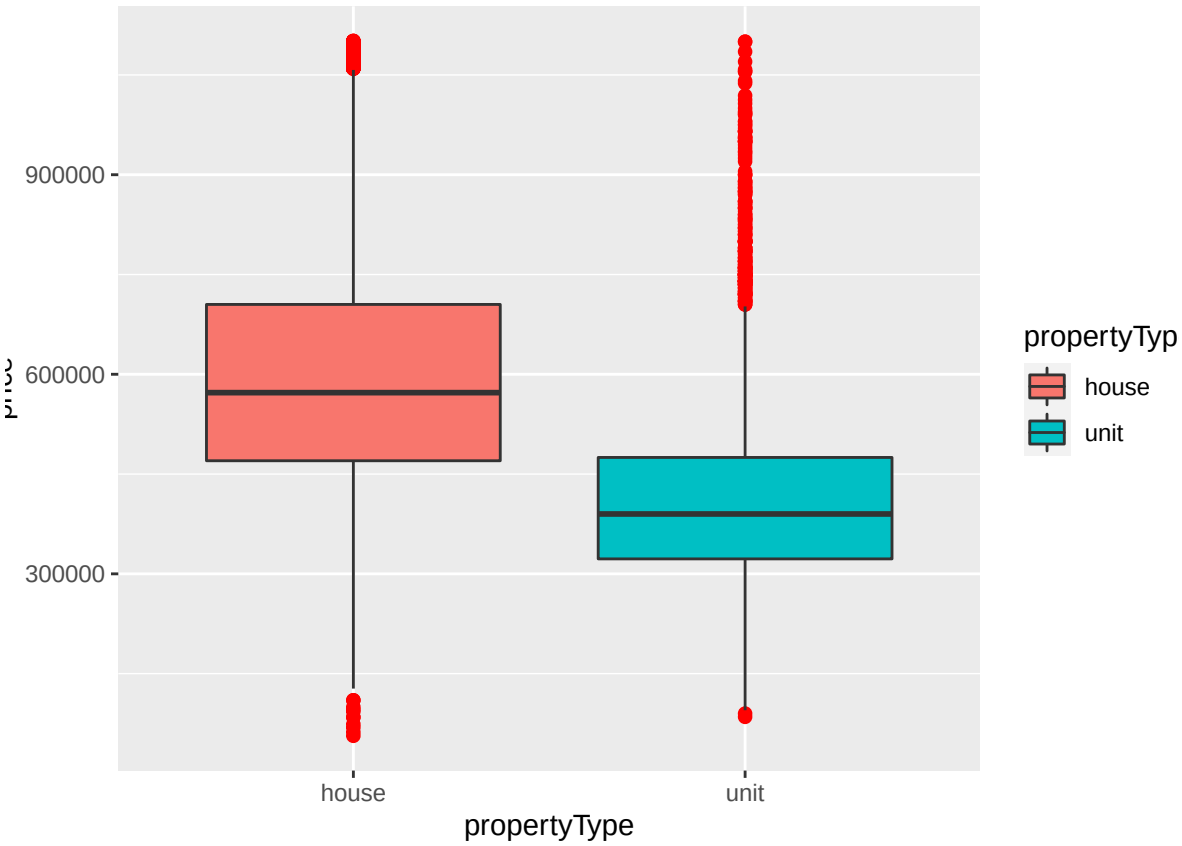
```
## [1] 0
```

```
# Detect outliers and remove them
outliers <- boxplot.stats(sales$price)$out
sales_df <- sales[-c(which(sales$price %in% outliers)),]
```

Explortory Data Analysis (EDA)

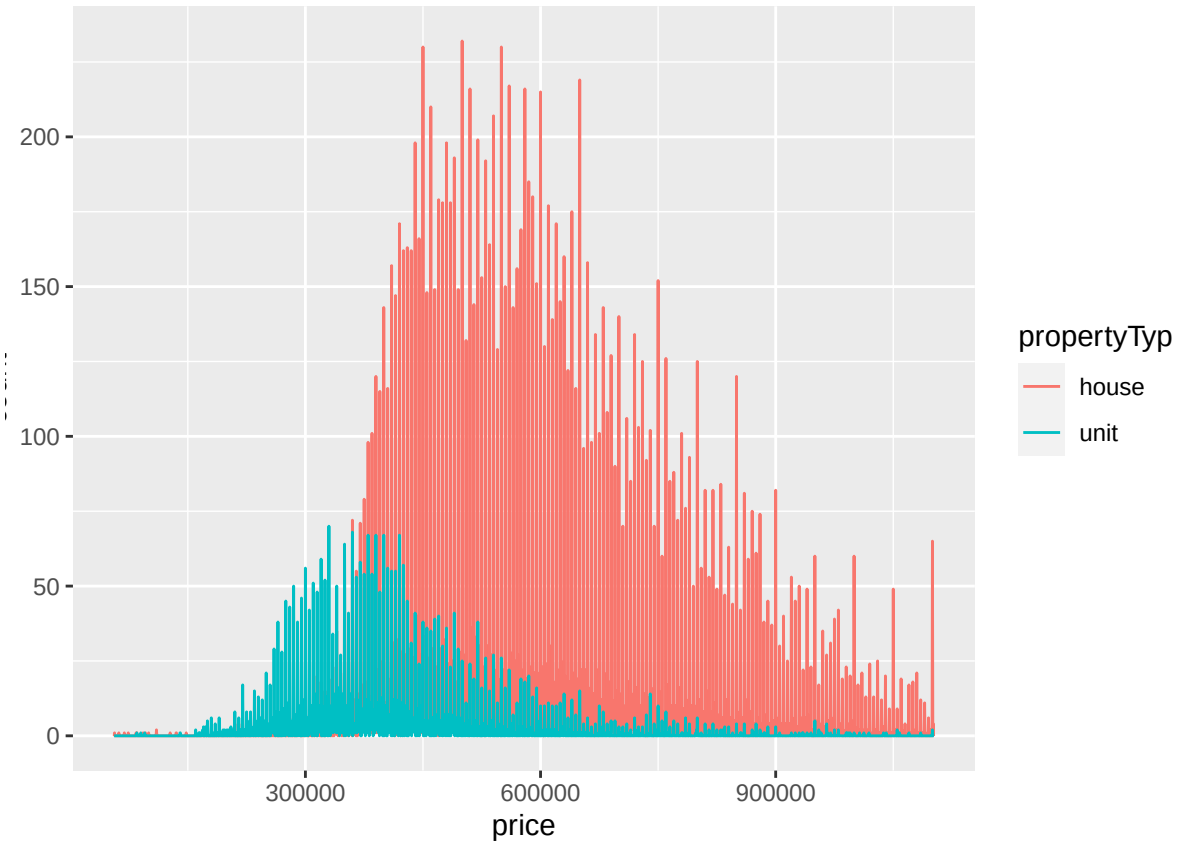
- The mean value and range of House are higher than Unit.

```
# Boxplot of Price and each PropertyType (House & Unit)
ggplot(sales_df, aes(propertyType, price)) +
  geom_boxplot(aes(fill = propertyType), outlier.color = 'red',
               outlier.shape = 20, outlier.size = 3) +
  theme(legend.position = 'right')
```



- It looks like more people would like to buy house rather than buying unit, even though the price of house is usually much higher than the price of unit.

```
# Property Sold Distribution for each PropertyType (House & Unit)
ggplot(data = sales_df, mapping = aes(x = price, colour = propertyType)) +
  geom_freqpoly(binwidth = 500)
```

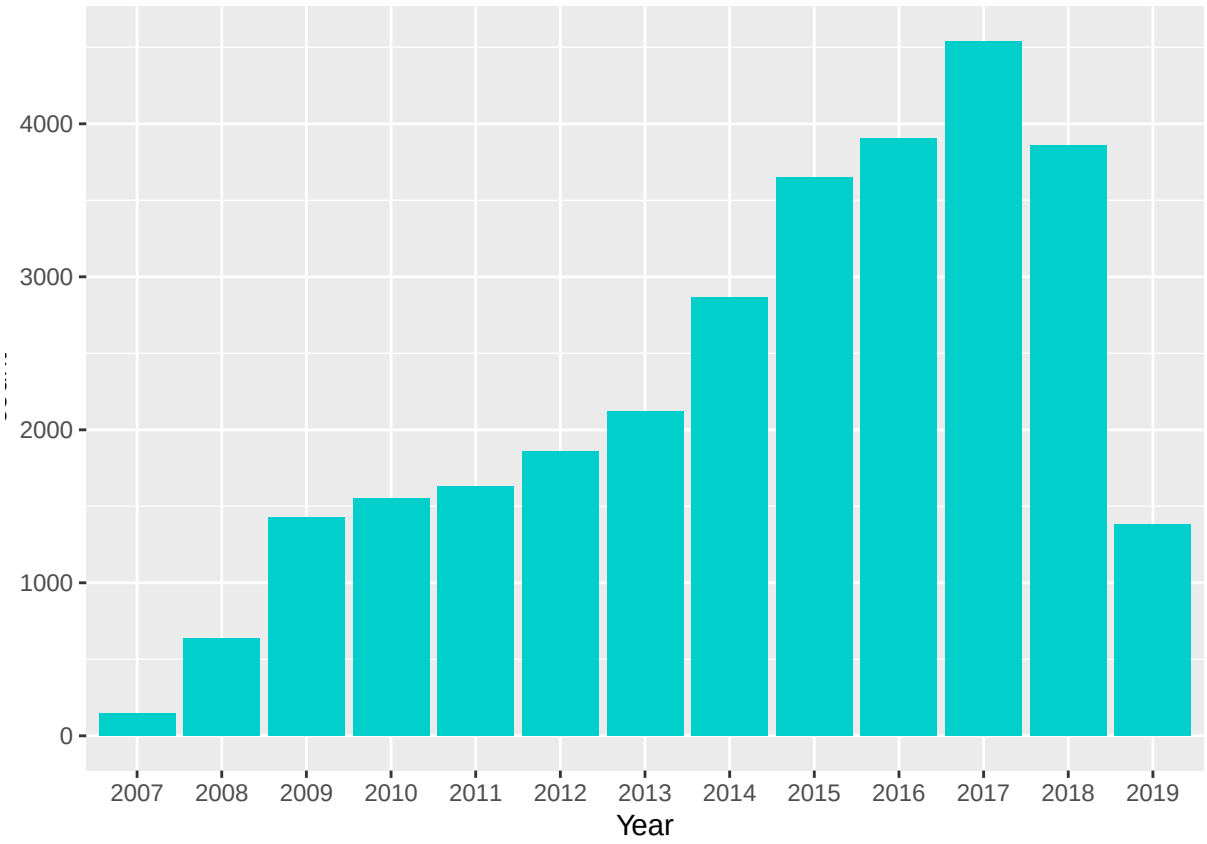



```
# Convert datesold(factor) to date format
sales$datesold <- as.character(sales$datesold)
sales$datesold <- mdy_hm(as.character(sales$datesold))
str(sales$datesold)
```

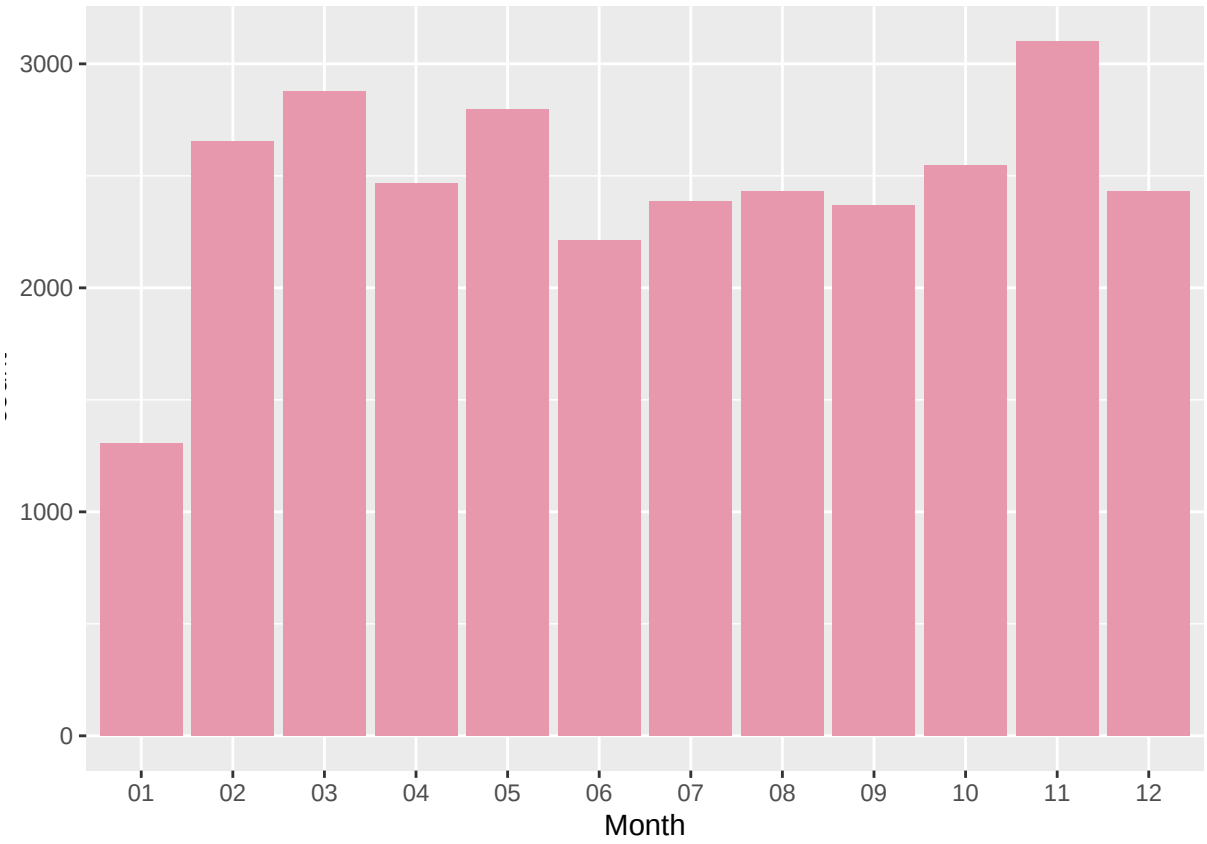
```
##  POSIXct[1:29580], format: "2007-02-07" "2007-02-27" "2007-03-07" "2007-03-09" "2007-03-21" ...
```

- January doesn't have many property sales relates to other 11 months. November has the best sales which is over 3000.
- From 2007 to 2017, number of property sales increases slowly. In 2018, it starts to decrease. The most interesting part is that number of sales decrease rapidly in 2019.

```
# Property sales distribution for years
ggplot(sales) + aes(x = format(datesold, "%Y")) +
  geom_bar(fill = '#00cfcc') +
  xlab("Year")
```

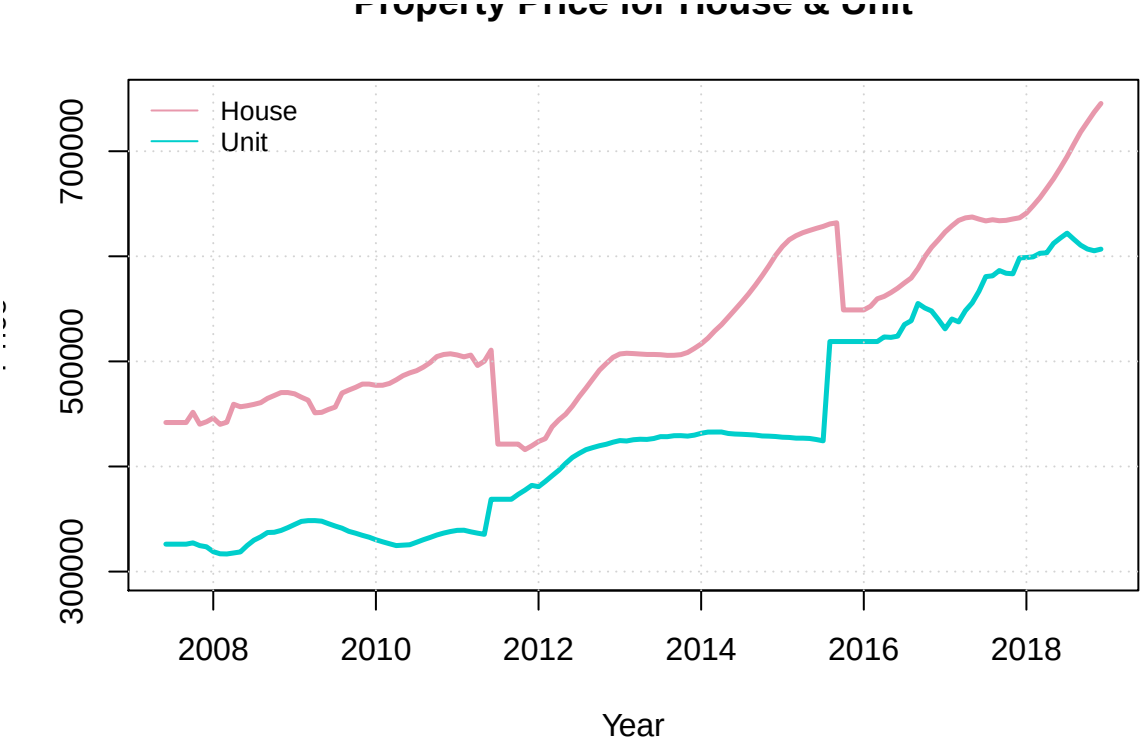



```
# Property sales distribution for months
ggplot(sales) + aes(x = format(datesold, "%m")) +
  geom_bar(fill = '#e898ac') +
  xlab("Month")
```



- We can infer from the graph itself that the data points follows an overall upward trend with some outliers in terms of sudden lower values.

```
# Monthly property price trend for House & Unit
myhouse <- mydata[which(mydata$type == "house"),]
plot(ts(myhouse[,"MA"], start = c(2007, 6), end = c(2018, 12),
        frequency = 12), main = "Property Price for House & Unit",
        xlab = 'Year', ylab = 'Price', col = '#e898ac',
        ylim = c(300000,750000), lwd = 2.5)
myunit <- mydata[which(mydata$type == "unit"),]
lines(ts(myunit[,"MA"], start = c(2007, 6), end = c(2018, 12),
        frequency = 12), col = '#00cfcc', , lwd = 2.5)
legend("topleft", c("House","Unit"),col=c("#e898ac","#00cfcc"), lty=1:1, cex=0.8, box.lty=0)
grid()
```



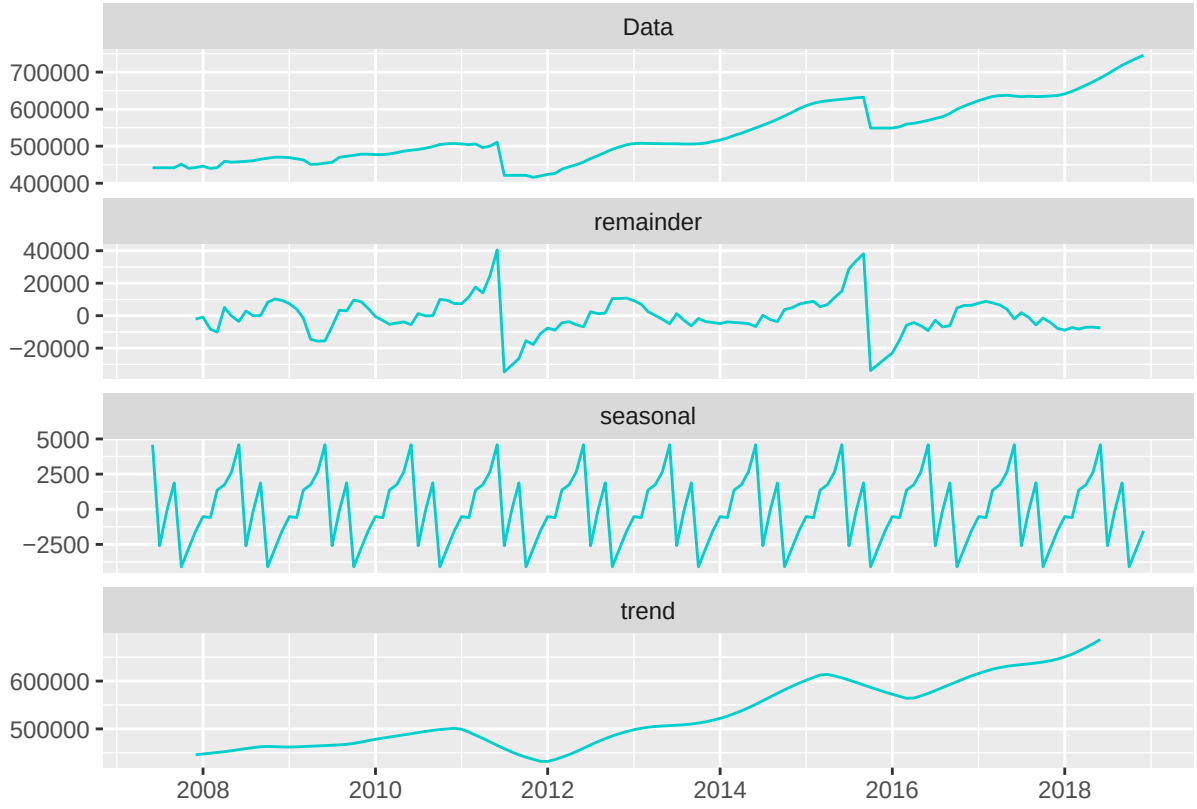
Decomposition

```
myhouse <- ts(myhouse[,"MA"], start = c(2007, 6), end = c(2018, 12),
   frequency = 12)

myhouse %>%
  decompose %>%
  autoplot(ts.colour = '#00cfcc')
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

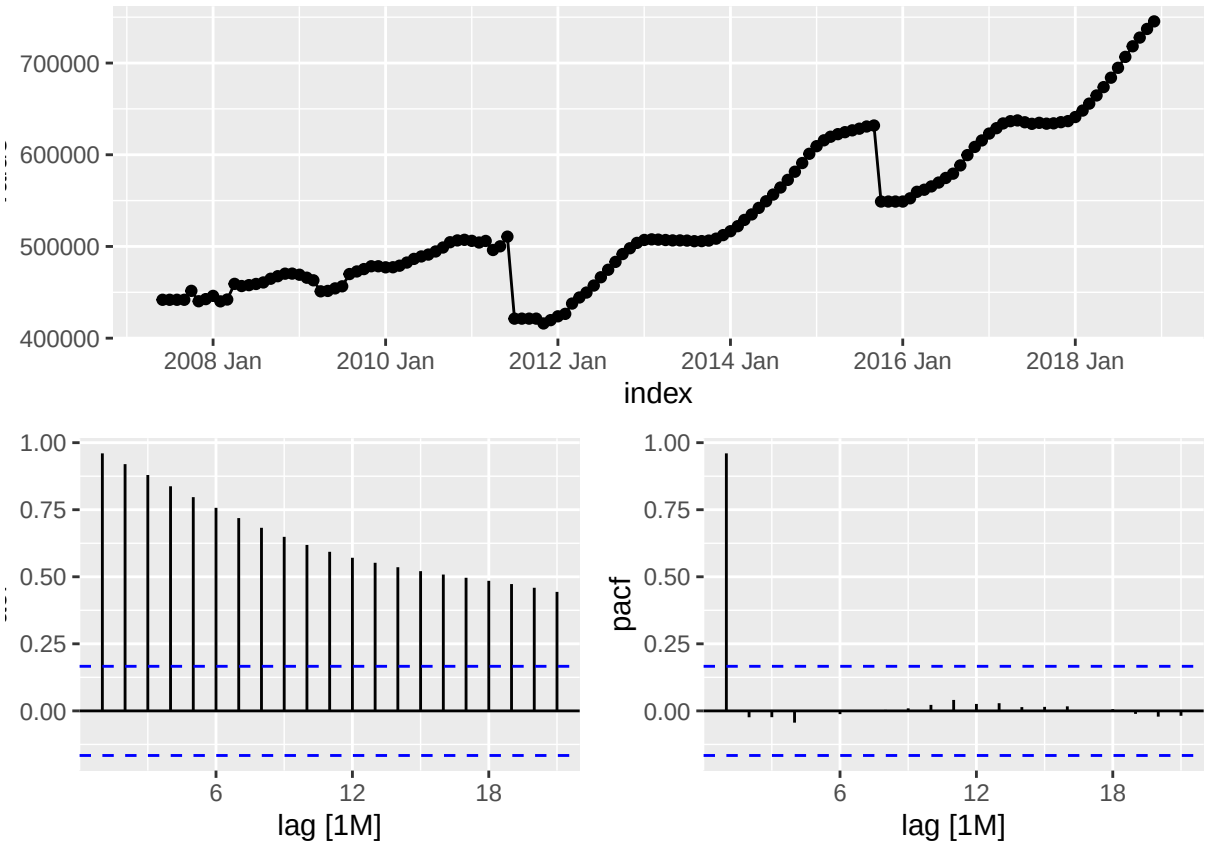
```
## Warning: Removed 24 row(s) containing missing values (geom_path).
```



ACF & PACF without 1st Differencing

- Without the 1st diff, the ACF decreases slowly which means the data is not stationary.

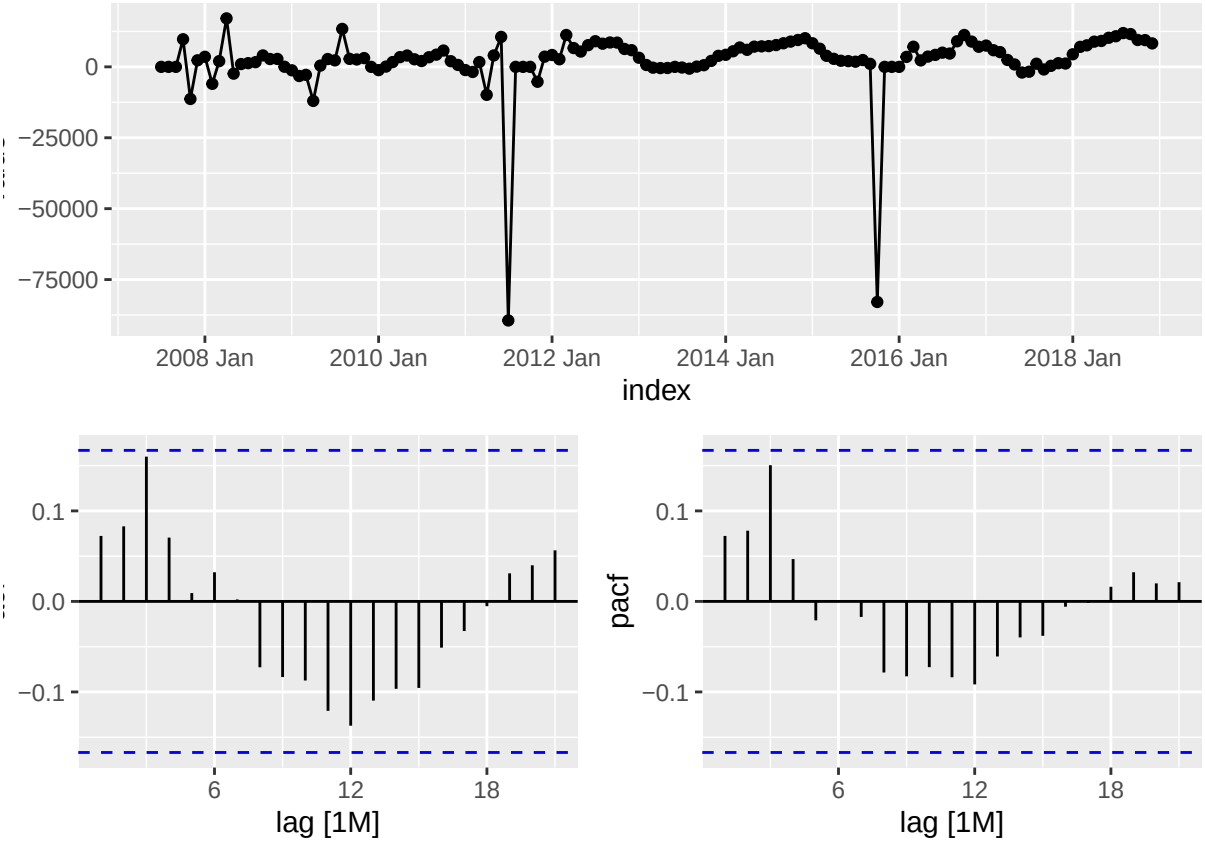
```
myhouse %>%
  as_tsibble() %>%
  gg_tsdisplay(value, plot_type = "partial")
```



ACF & PACF with 1st Differencing

- With 1st diff, there is no lag across the dash line in the ACF and PACF plots which is a good sign. From these plots, I can say that $\quad = 1$ and $\quad = 1$.

```
myhouse %>%
  diff() %>%
  as_tsibble() %>%
  gg_tsdisplay(value, plot_type = "partial")
```



Stationary Check - Unit Root Test

Null Hypothesis in KPSS Test: the series is stationary.

- P-value = 0.1 > 5% (assume the significant level is 5%) which we accept the null hypothesis, thus we can conclude that the series is stationary.
- Number of diffs = 1, which means 1st diff is required for the data to be stationary.

```
myhouse %>%
  as_tsibble() %>%
  mutate(diff_value = difference(value)) %>%
  features(diff_value, unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1     0.210         0.1
```

```
myhouse %>%
  as_tsibble() %>%
  features(value, unitroot_ndiffs)
```

```
## # A tibble: 1 x 1
##   ndiffs
##    <int>
## 1      1
```

ARIMA Modeling

I build the ARIMA(0, 1, 0) model and apply it to myhouse. The AICc is 2979. I also force the model to run all the combinations, but still get ARIMA(0, 1, 0) and same AICc. Force the model runs all the combination, but still get ARIMA(0,1,0).

```
#force run all combinations
myhouse %>%
  as_tsibble() %>%
  model(ARIMA(value ~ pdq(d=1), stepwise = F, approximation = F)) %>%
  report()
```

```
## Series: value
## Model: ARIMA(0,1,0)(1,0,1)[12] w/ drift
##
## Coefficients:
##         sar1  sma1  constant
##       -0.88  0.78      4008
## s.e.   0.22  0.27      1736
##
## sigma^2 estimated as 132983970:  log likelihood=-1486
## AIC=2979   AICc=2979   BIC=2991
```

```
fit <- myhouse %>%
  as_tsibble() %>%
  model(arima = ARIMA(value ~ pdq(0, 1, 0))) %>%
  report()
```

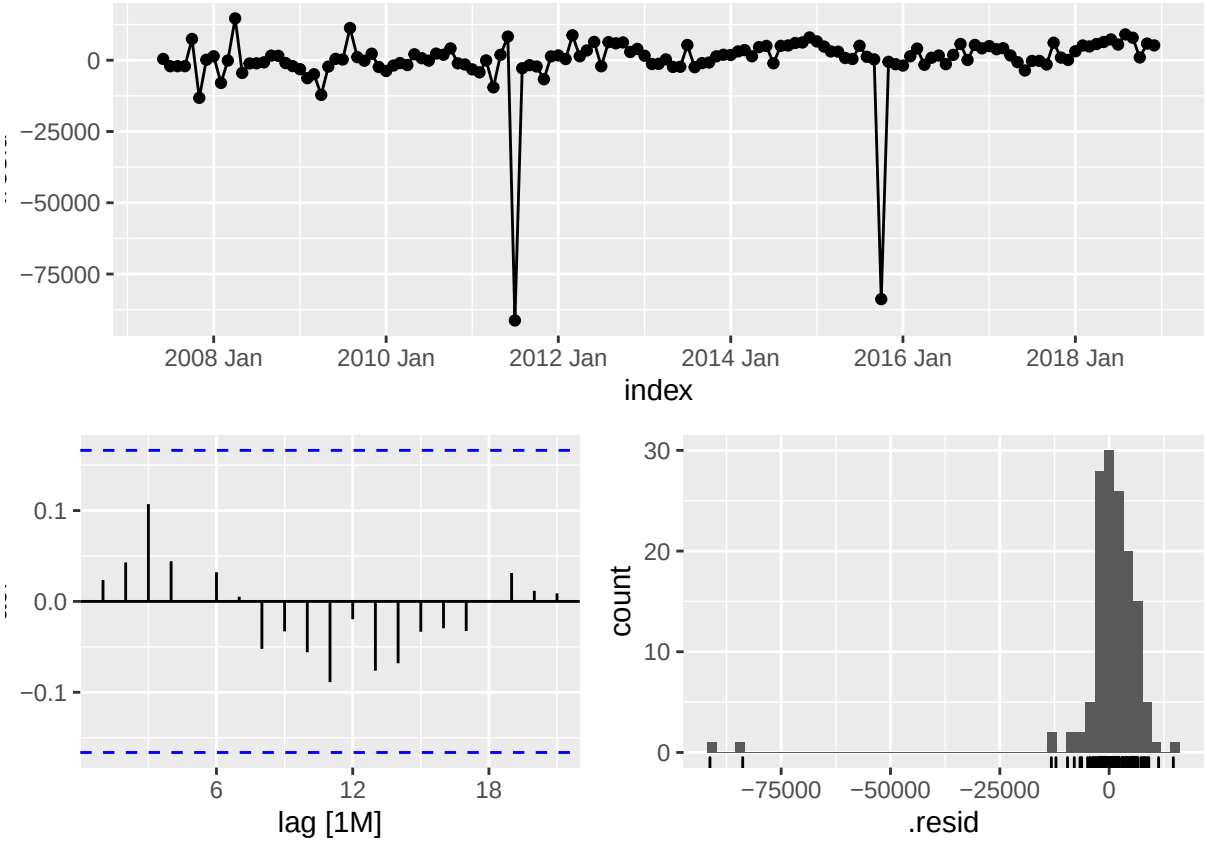
```
## Series: value
## Model: ARIMA(0,1,0)(1,0,1)[12] w/ drift
##
## Coefficients:
##         sar1  sma1  constant
##       -0.88  0.78      4008
## s.e.   0.22  0.27      1736
##
## sigma^2 estimated as 132983970:  log likelihood=-1486
## AIC=2979   AICc=2979   BIC=2991
```

Diagnostic Measures - Ljung-Box Test

Null Hypothesis in Ljung-Box Test: No serial correlation for future data.

- P-value = 0.944 > 5% which is too large to reject null hypothesis (No serial correlation for future data), so there is no pattern in the residuals. In addition, the plots support the result:
  - There is no lag across the dash line in ACF
  - The residuals are normally distributed

```
gg_tsresiduals(fit)
```




```
#check for autocorrelation: Ljung-box Test
#Null Hypothesis: No serial correlation upto 8 lags
#lag = sqrt(length(data)), dof = (p+q)
augment(fit) %>%
  features(.resid, ljung_box, lag = 8, dof = 0)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 arima     2.84     0.944
```

Forecasting

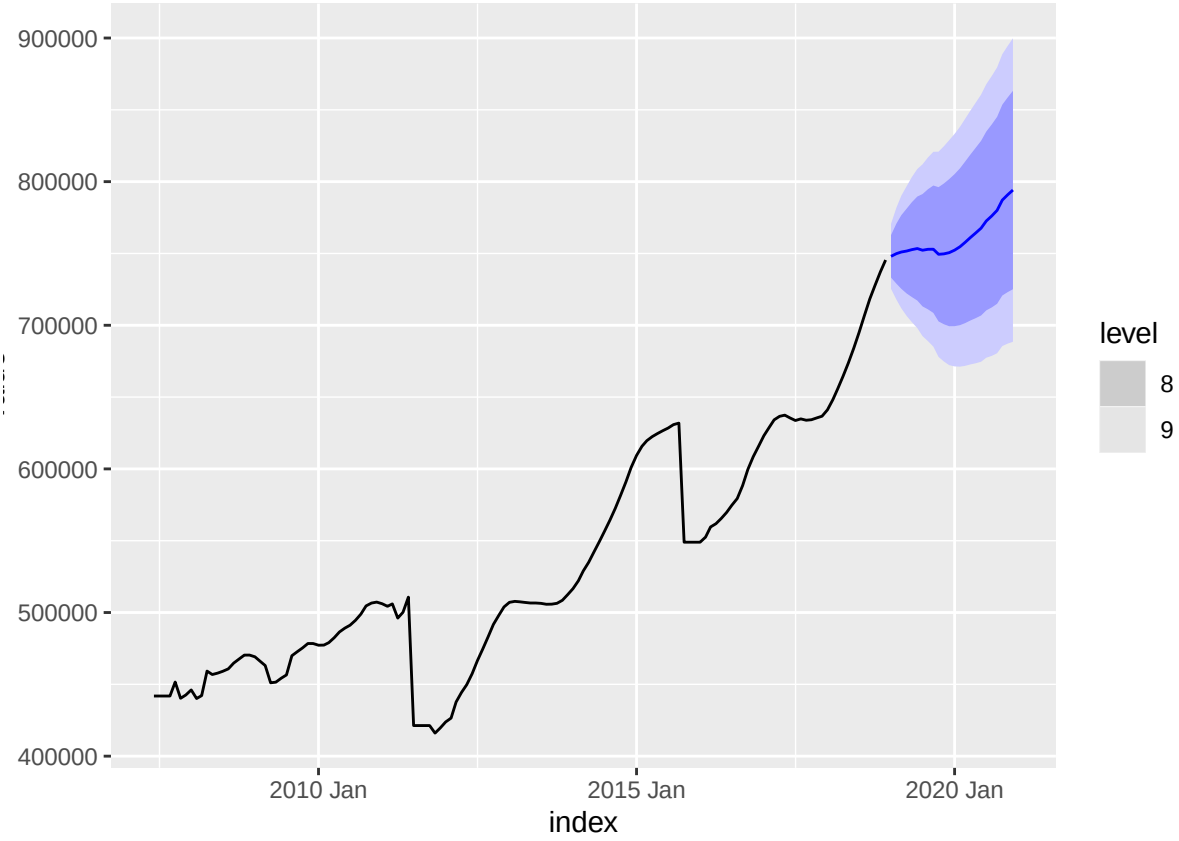
This plot shows the 80% and 95% confidence level of the prediction. If I focus on the lower bound of the confidence level, the trend of house price would be decrease until January 2020 and increase after January 2020. If I focus on upper bound, it will have upward trend since 2019.

```
fit %>%
  forecast(h = 24) %>%
  print()
```

```
## # A fable: 24 x 4 [1M]
## # Key:     .model [1]
##    .model    index                value   .mean
##    <chr>     <mth>               <dist>   <dbl>
##  1 arima  2019 Jan  N(747994, 133005374) 747994.
##  2 arima  2019 Feb  N(749858, 266010748) 749858.
```

```
##  3 arima  2019 Mar  N(750996, 399016122) 750996.
##  4 arima  2019 Apr  N(751673, 532021495) 751673.
##  5 arima  2019 May  N(752675, 665026869) 752675.
##  6 arima  2019 Jun  N(753405, 798032243) 753405.
##  7 arima  2019 Jul  N(752239, 931029330) 752239.
##  8 arima  2019 Aug N(752889, 1064026418) 752889.
##  9 arima  2019 Sep N(752946, 1197023505) 752946.
## 10 arima  2019 Oct N(749406, 1330020592) 749406.
## # ... with 14 more rows
```

```
fit %>%
  forecast(h = 24) %>%
  autoplot(myhouse)
```



Conclusions

From the plots in Exploratory Data Analysis, the property sales increase from 2007 to 2017. In the meanwhile, the price of property increases. However, the price starts to decrease may be due to the rapid decreasing in property sales. After applying the ARIMA model to forecast the house prices in future 2 years, I got the prediction plot in 24 periods (2 years). With the result of forecasting, I'm be able to provide insightful business/individual decisions. For homebuyers, I would suggest that buying houses before January. 2020, and selling houses after January 2020. Because homebuyers can buy the houses at a relatively lower prices, and sell their houses at a relatively higher prices. For real estate companies or agents, they could use the upper bound and lower bound as a reference for price range. However, it should still take the house market into account.