Forecasting Median Home Sale Prices in NJ, USA

Ajay Vishnu Addala

2023-08-01

```
library(fpp)
```

```
## Loading required package: forecast
```

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

```
## Loading required package: fma
```

```
## Loading required package: expsmooth
```

```
## Loading required package: lmtest
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: tseries
```

```
library(fpp2)
```

```
## ── Attaching packages ─────────────────────────────────────────────── fpp2 2.5 ──
```

```
## ✔ ggplot2 3.4.2
```

```
##
```

```
##
## Attaching package: 'fpp2'
```

```
## The following objects are masked from 'package:fpp':
##
##     ausair, ausbeer, austa, austourists, debitcards, departures,
##     elecequip, euretail, guinearice, oil, sunspotarea, usmelec
```

```
library(TTR)
library(ggplot2)
library(readr)
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
```

```
NJ_MedianListingPrice_AllHomes <- read_csv(file = 'Data_NJ_MedianListingPrice_AllHomes.c
sv')
```

```
## Rows: 257 Columns: 2
```

```
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): YYYY-MM
## dbl (1): Value
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
NJ_Home_Raw <- NJ_MedianListingPrice_AllHomes$Value
NJ_Home_TS_actual <- ts(NJ_Home_Raw,frequency = 12, start = c(1996,4))
```

About the Data

About

- Zillow see's listing nationwide. Taking advantage of the vast amount of listing data, Zillow has been able to produce monthly index for various data point of interest. For this mid-term, we will look at the median home prices for House Listing in New Jersey.

Data Source

- Link: https://www.zillow.com/research/data/#median-home-value (https://www.zillow.com/research/data/#median-home-value)

Data Dictionary

- YYYY-MM: Year and Month during with the data was recorded
- Value: Median Listing Price of properties in New Jersey, USA

Question and Hypothesis

Question

- What will be the best method to forecast the given time series data?

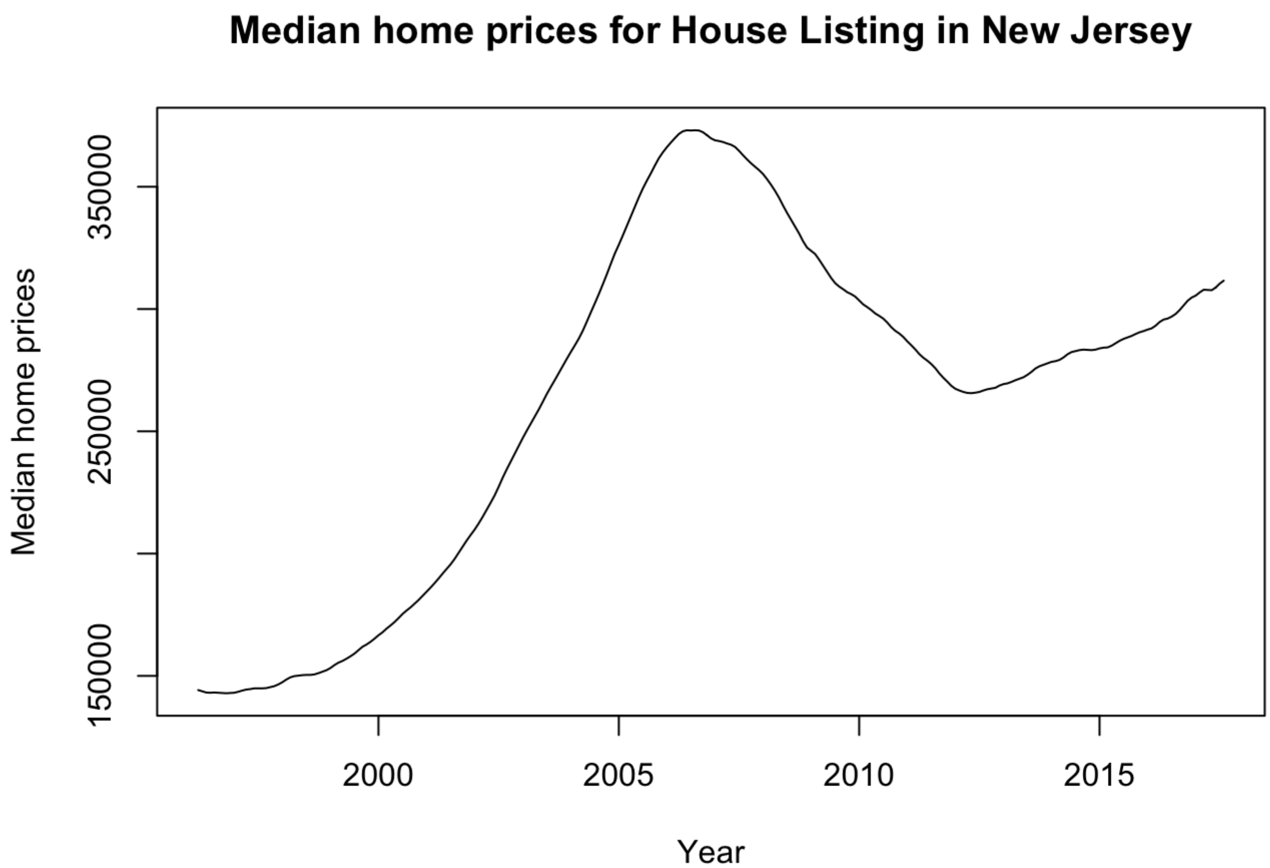
Hypothesis

- As we know, House pricing increases with time. It can be a time series with an increasing trend. HoltWinters can be the best method to forecast this type of data.
- We can check this hypothesis based on the accuracy of each model that we can check below.

Plot and Inference

Time Series Plot

```
plot(NJ_Home_TS_actual, main = 'Median home prices for House Listing in New Jersey', xla
b = 'Year', ylab = 'Median home prices')
```



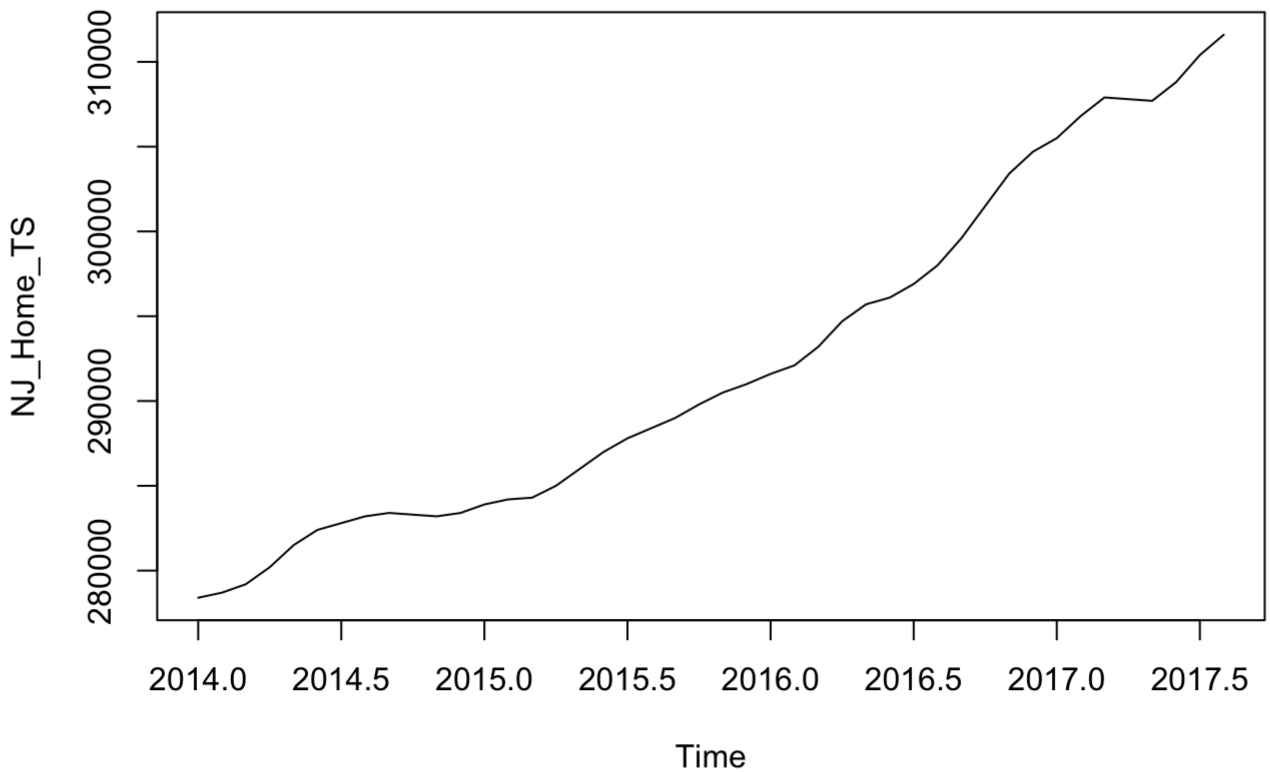
- We start with plotting the time series to visualise and understand the data.

Initial Observations

- The plot shows that there is an increasing trend in the median home prices starting from 1996 till around 2006.
- From 2006 till 2012, there has been a decreasing trend in the home prices.
- From 2012, there has been steady increasing trend till the year 2017.
- The data however doesn't appear to show any seasonal variation.
- If we were to forecast the data, we should be considering the window from 2014.

Considering only a window

```
NJ_Home_TS <- window(NJ_Home_TS_actual, start = 2014)
plot(NJ_Home_TS)
```



- Window function has been used from the year 2014 to forecast the data better.
- If we consider the whole data, that might not give us the exact forecast.
- From 2014 it will be more than 3 years data that we are considering and this data should be good enough to be considered for forecasting.

Central Tendency

Min, max, mean, median, 1st and 3rd Quartile values of the times series

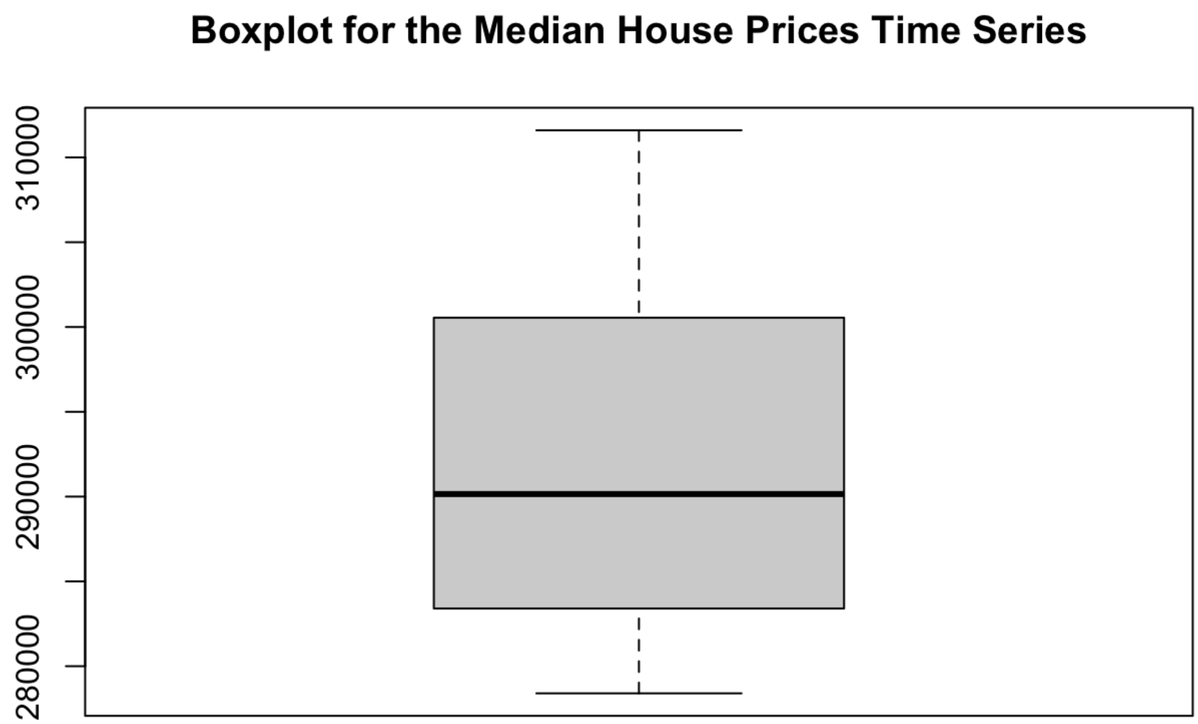
```
summary(NJ_Home_TS)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  278400  283400  290150  292286  300075  311600
```

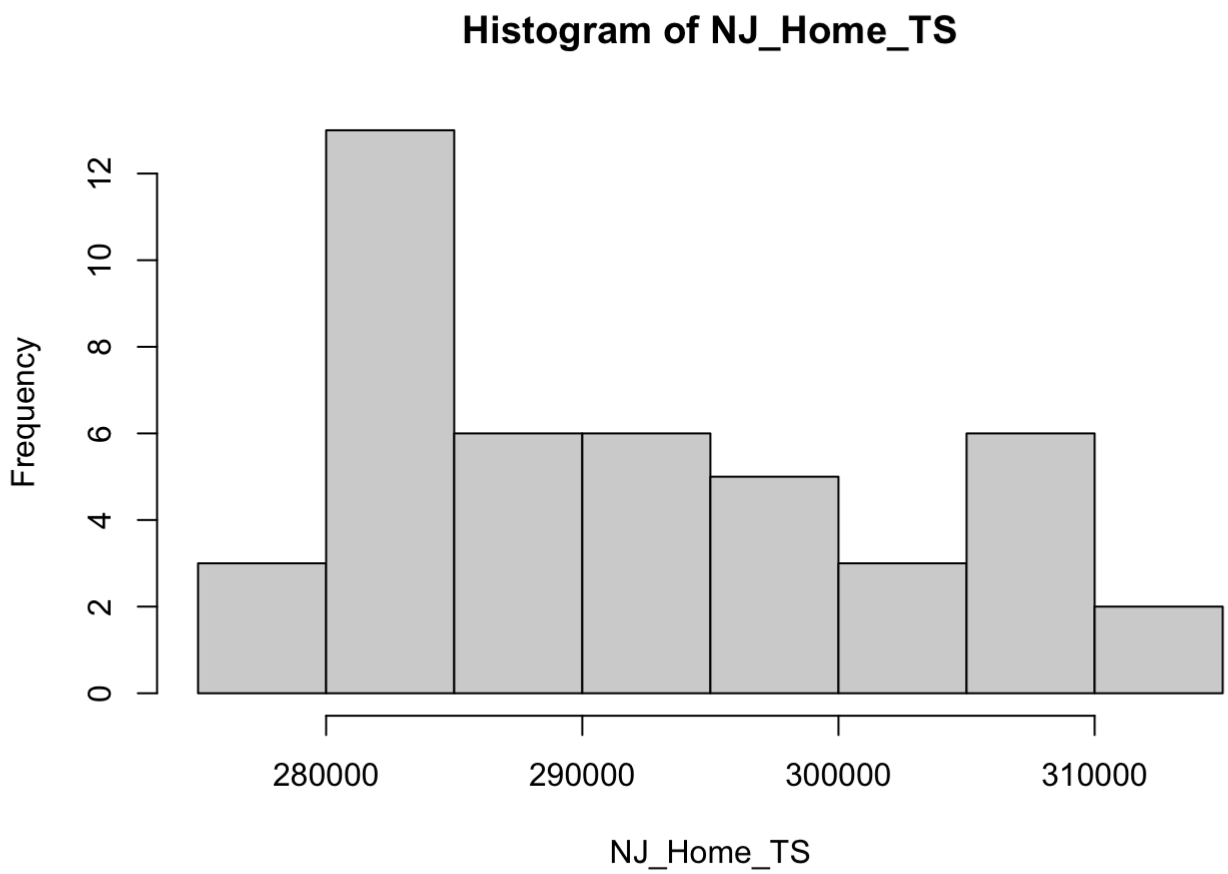
- The summary function above gives the min, max, mean, median, 1st and 3rd Quartile values of the times series.

Box Plot

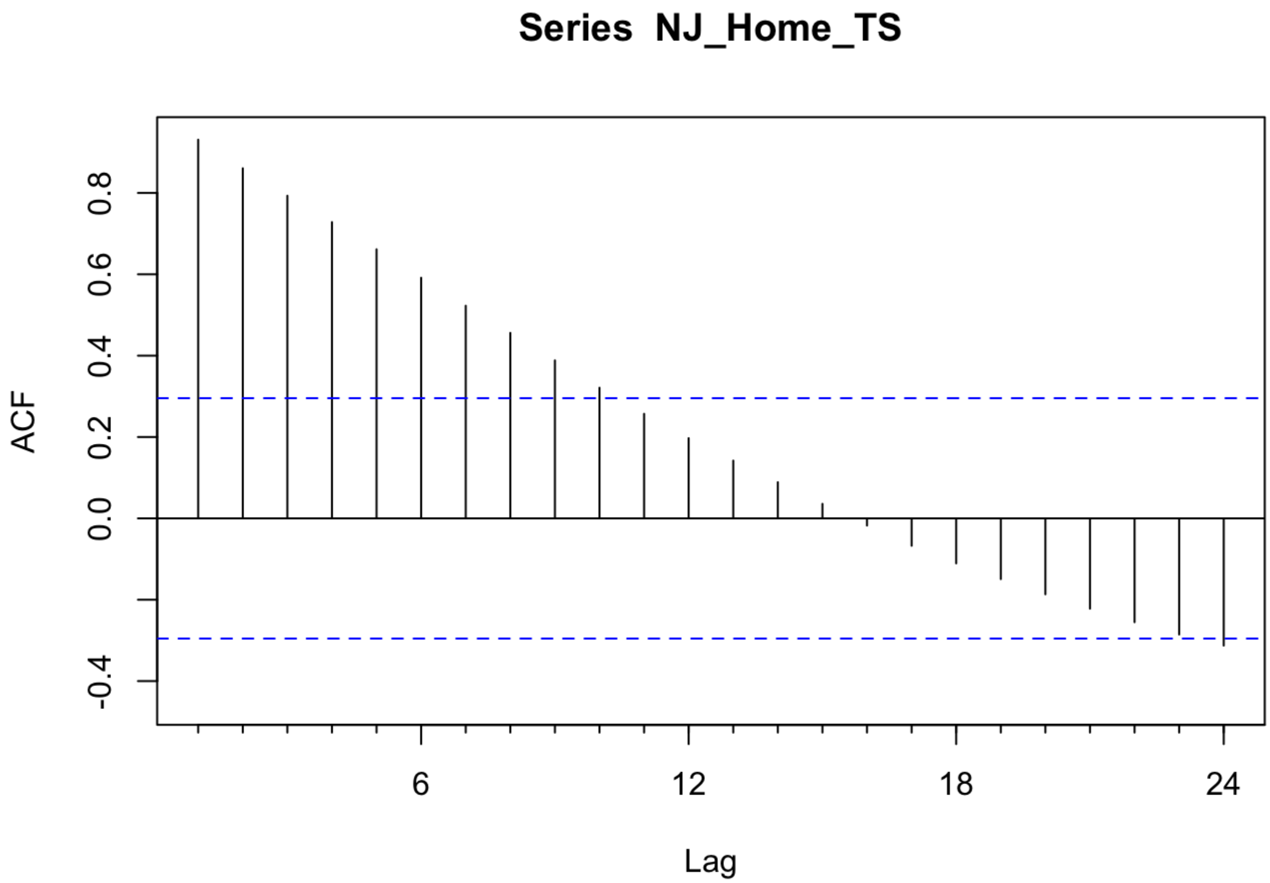
```
boxplot(NJ_Home_TS, main = 'Boxplot for the Median House Prices Time Series')
```




```
hist(NJ_Home_TS)
```




```
Acf(NJ_Home_TS)
```



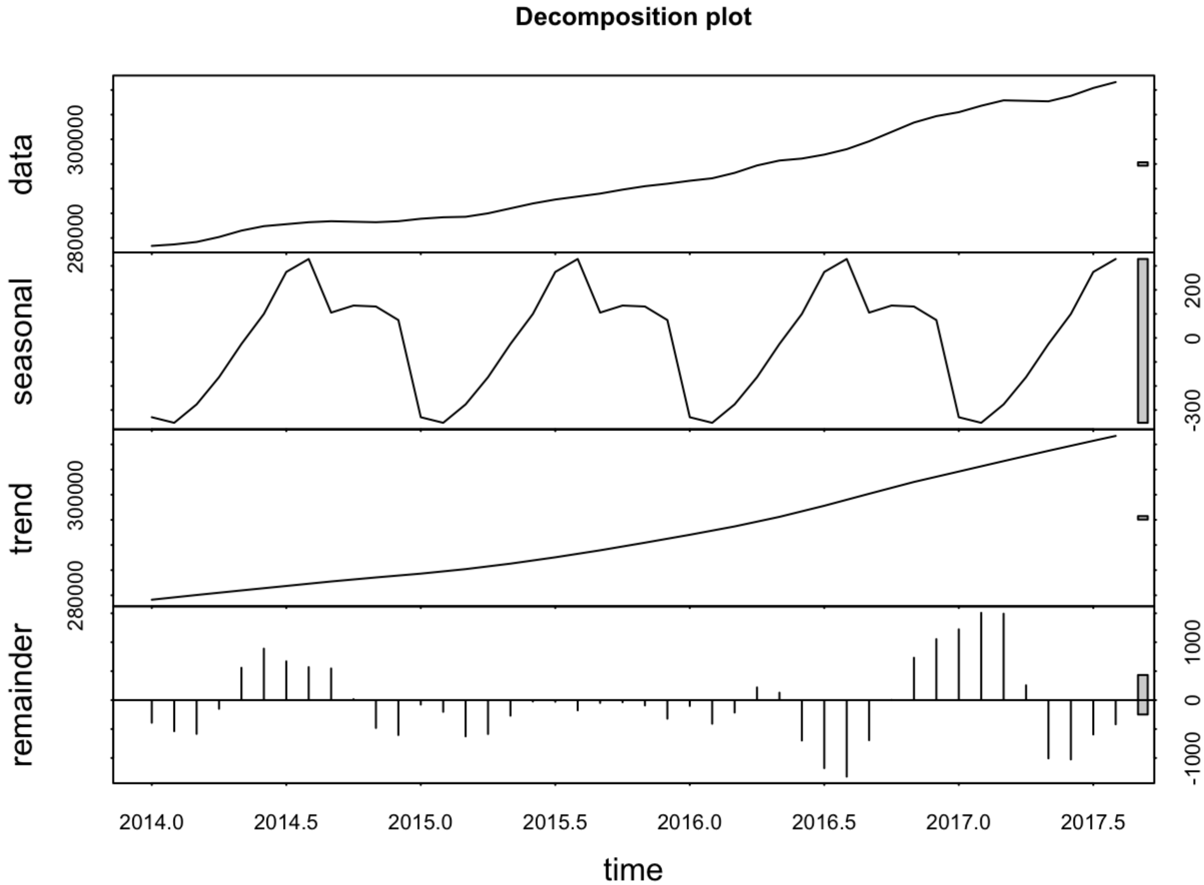
Observations and Inferences

- The boxplot shows that there are no outliers in the data.
- The Median is more towards the first quartile.
- From summary, we can also see that the median value is less than the mean for the time series.
- This means that the data is right skewed. This can be justified seeing the histogram above as well.
- From the ACF plot, we can see that many of the values crossed the confidence intervals, stating there is a trend component in the data.
- Also, we can see that after 15th lag period, the ACF plot is dipping into the negative values stating seasonality also exists in the data.

Decomposition

Decomposition Plot

```
stl_decomp <- stl(NJ_Home_TS,s.window ="periodic")
plot(stl_decomp, main = 'Decomposition plot')
```



Is there a seasonality?

- Yes, the time series is seasonal.
- We can infere this from the decomposition plot above.

Decomposition characteristic

```
decom <- decompose(NJ_Home_TS)
decom$type
```

```
## [1] "additive"
```

- The decomposition seems to be additive.
- Because, with as trend increases, we do not see any increase in the seasonality. The seasonality appears to be the same throughout.

Seasonal monthly indices

```
decom$figure
```

```
##  [1]   82.812500    5.034722 -608.854167 -225.520833  -25.520833 -190.104167
##  [7]  148.090278   81.423611   92.534722  177.256944  263.368056  199.479167
```

Observations and Inferences

- From 2014 to 2017, the values of the time series seem to increase throughout.

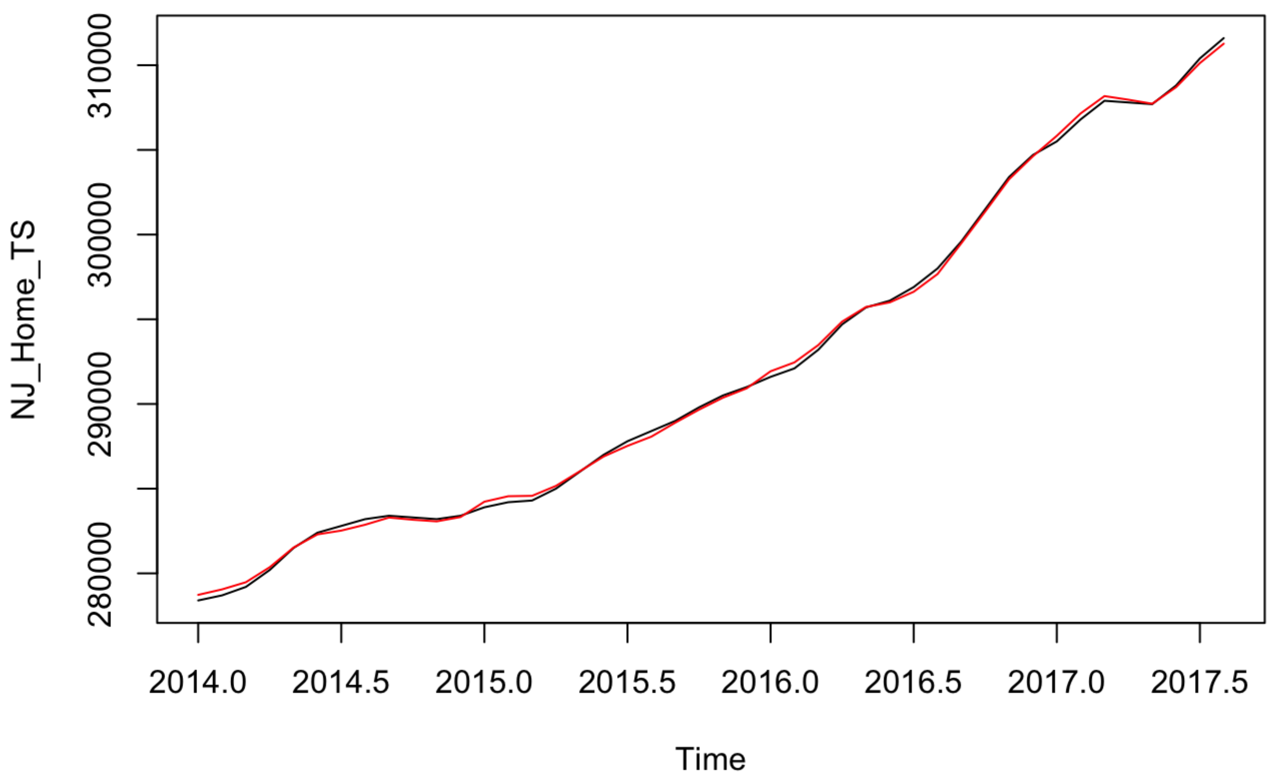
- We can see a peak in Sep 2014 and then a dip and Oct 2014 and then continuous increase.
- Then again a similar case for Mar 2017 and then a dip in Apr 2017 and then a continuous increase.

Plausible reasons

- The plausible reason might be because the influx of international students in September their studies.

Seasonality adjusted plot

```
plot(NJ_Home_TS)
lines(seasadj(stl_decomp), col="Red")
```



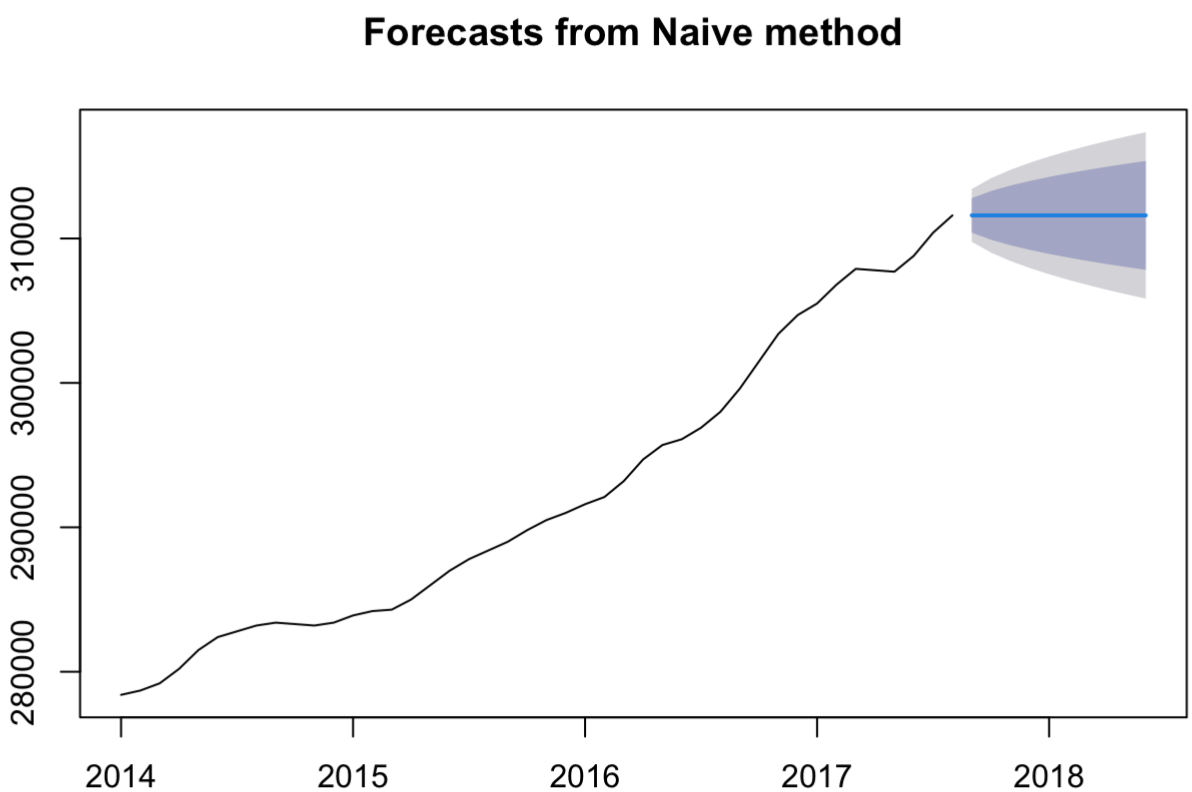
- There are minor fluctuations that can be observed after applying seasonal adjustment.
- With time, these fluctuations will cause deviations and change our forecast. So, it is important to consider the seasonal variation in the data.

Testing various Forecasting methods for the given dataset

Naïve Method

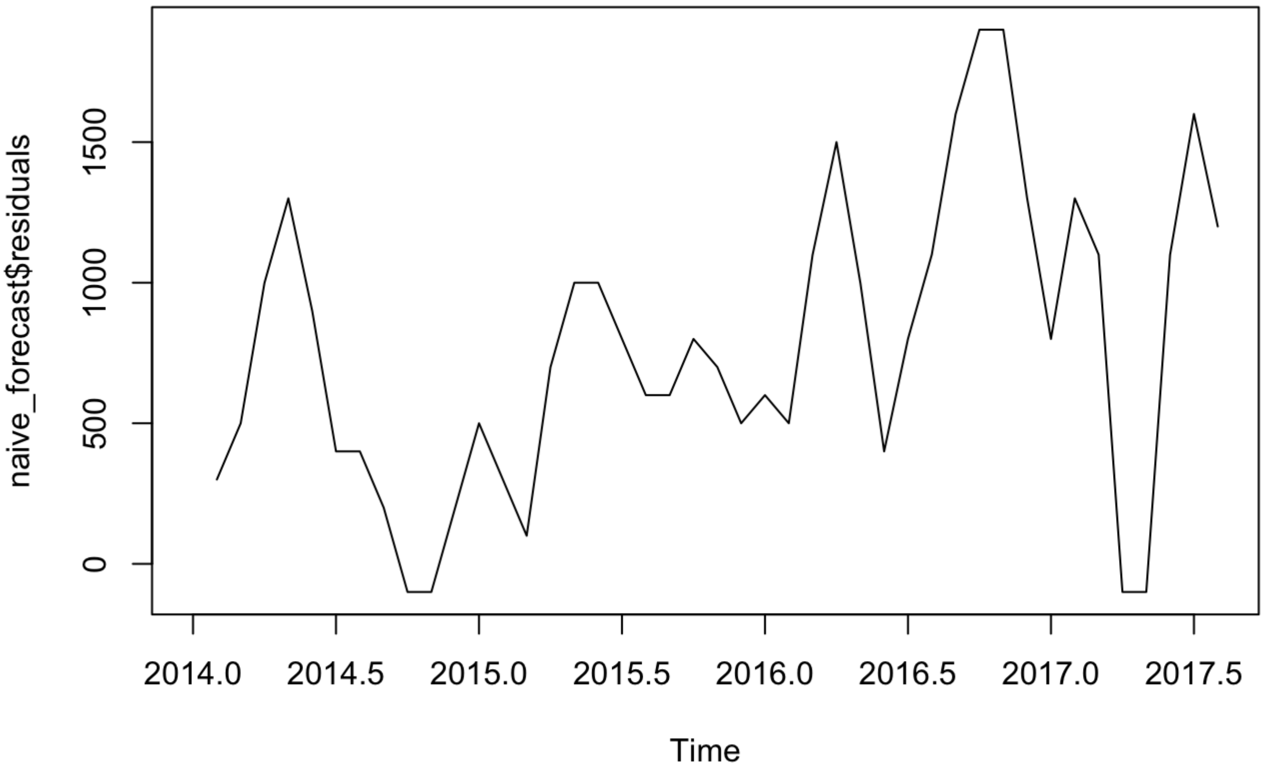
Output

```
naive_forecast = naive(NJ_Home_TS)
plot(naive_forecast)
```



Residual Analysis

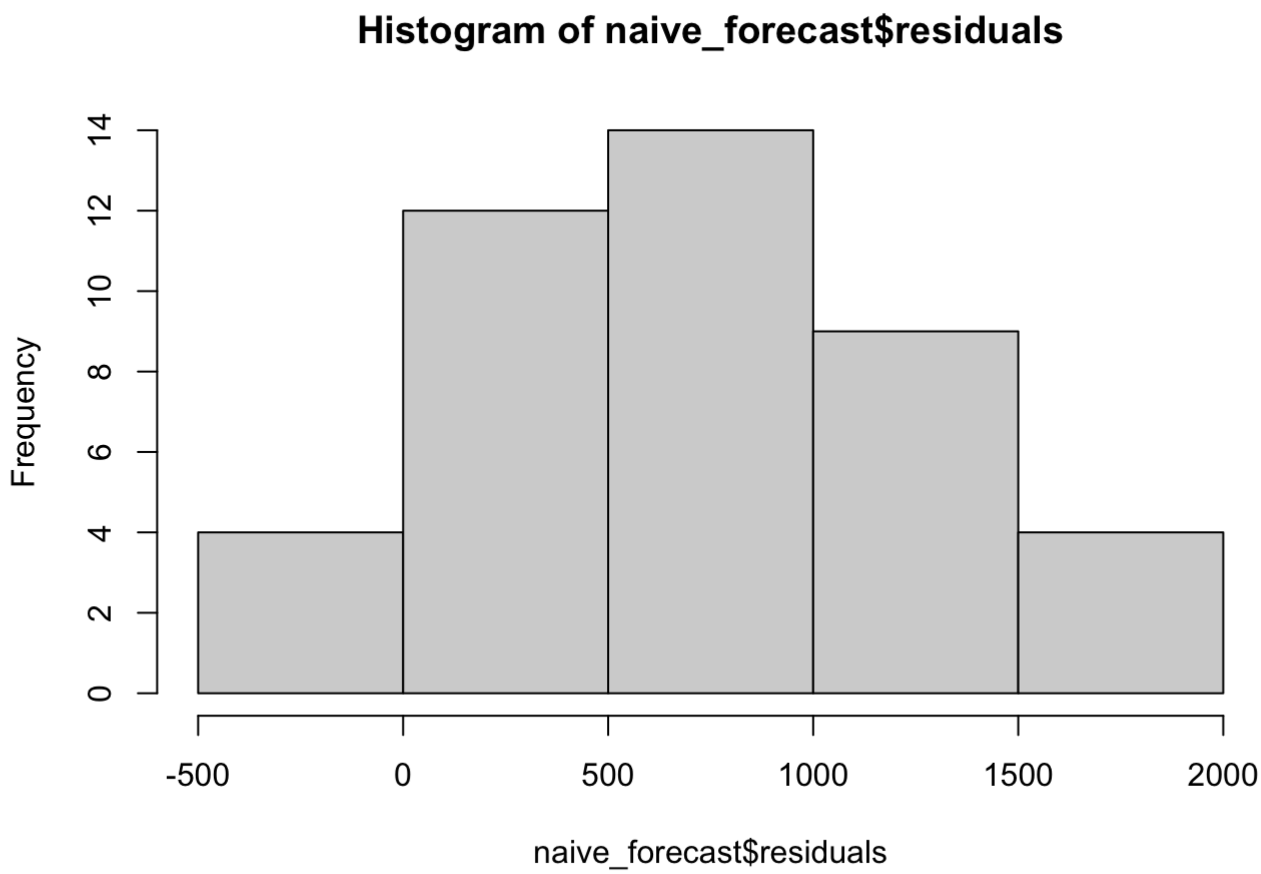
```
plot(naive_forecast$residuals)
```



- The residuals appear to have increasing positive values and then peaked in the third quarter of the year 2016 and then dipped down.
- All the residuals are positive. The residuals do not seem to have a mean at zero.
- We can test this hypothesis in the coming tests.

Residuals Histogram

```
hist(naive_forecast$residuals)
```

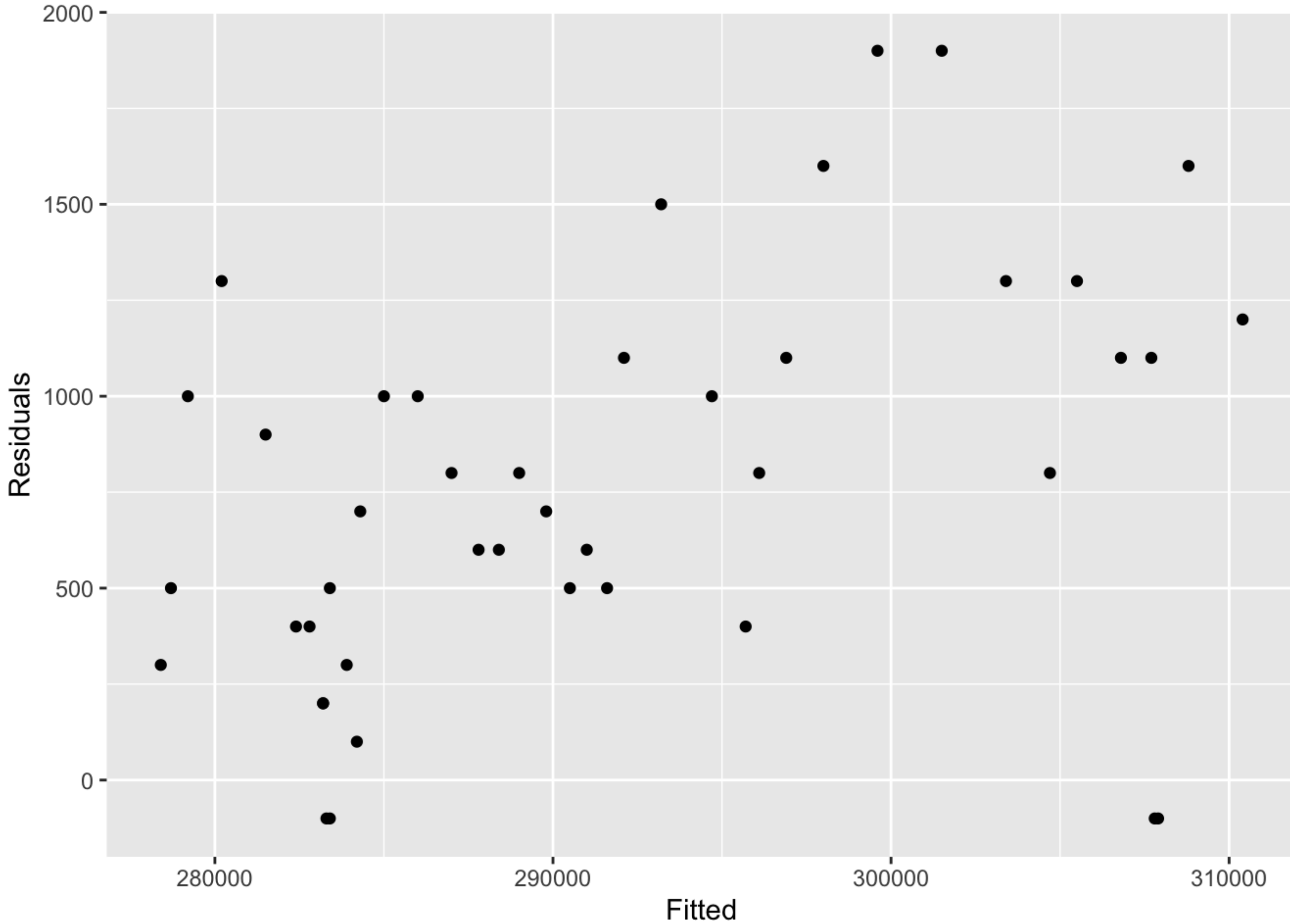


- The histogram appears to be normally distributed.
- But the values do not have a mean zero. The histogram appears to be skewed on one side.
- This means that the data is biased as the mean is not zero.

Fitted vs Residual Values

```
cbind(Fitted = fitted(naive_forecast),
      Residuals=residuals(naive_forecast)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Fitted, y=Residuals)) + geom_point()
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```

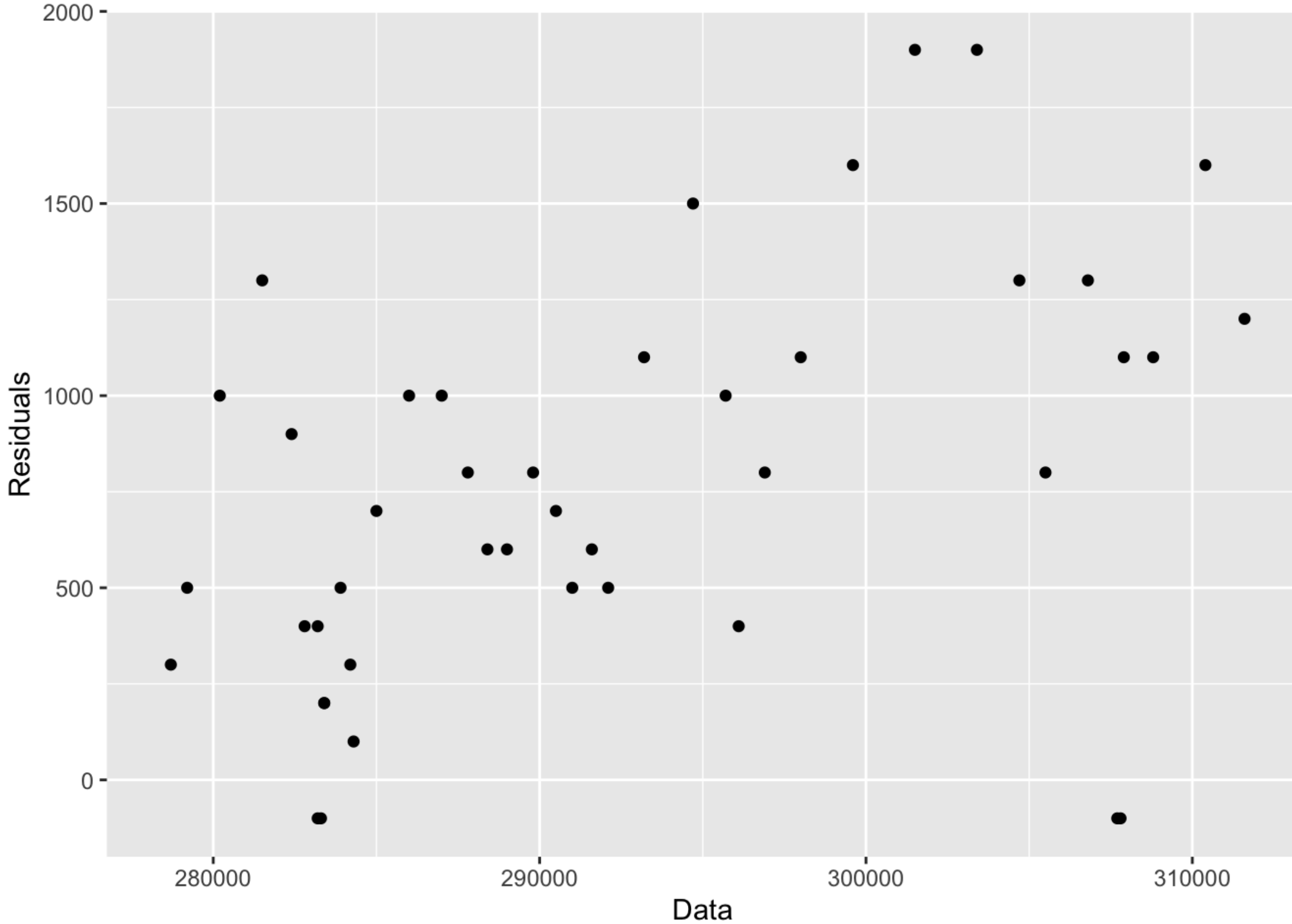


- The Fitted vs Residuals plot appears to have a trend. The plot slightly shows a straight diagonal line pattern.
- This means there is heteroscedasticity in the errors which means that the variance of the residuals may not be constant.

Actual vs Residual values

```
cbind(Data=NJ_Home_TS,
      Residuals=residuals(naive_forecast)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Data, y=Residuals)) + geom_point()
```

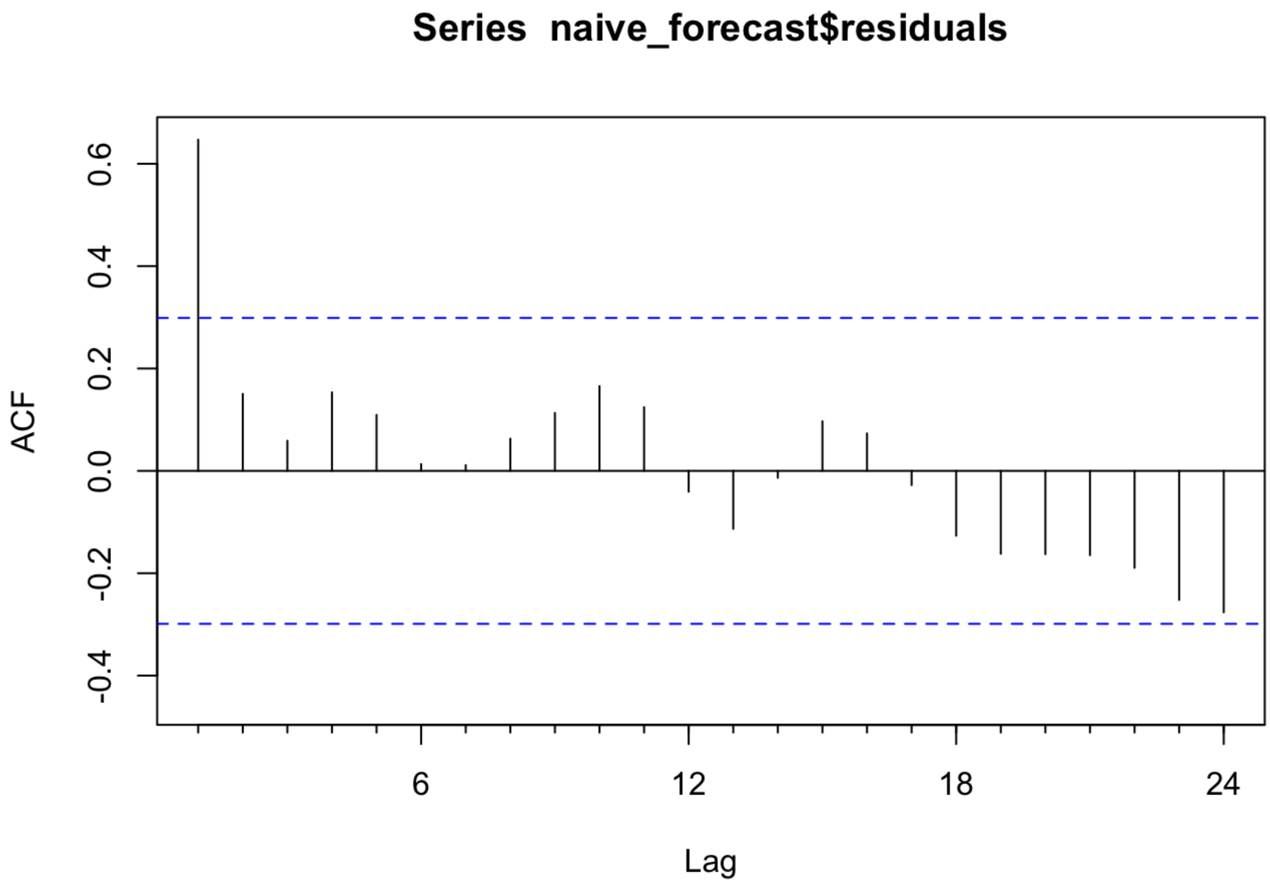
```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



- Similar to the previous plot, The actual vs Residuals plot also appears not to be random.

ACF of residuals

```
Acf(naive_forecast$residuals)
```



- Values of the Acf have crossed the confidence level meaning there is a trend in the residuals and we have missed some variable in our forecast.
- The Acf values also show seasonality in the plot and we missed this variable too.
- Meaning that naive forecast is missing some main variables which we have missed our consideration for the forecast.

Accuracy

```
accuracy(naive_forecast)
```

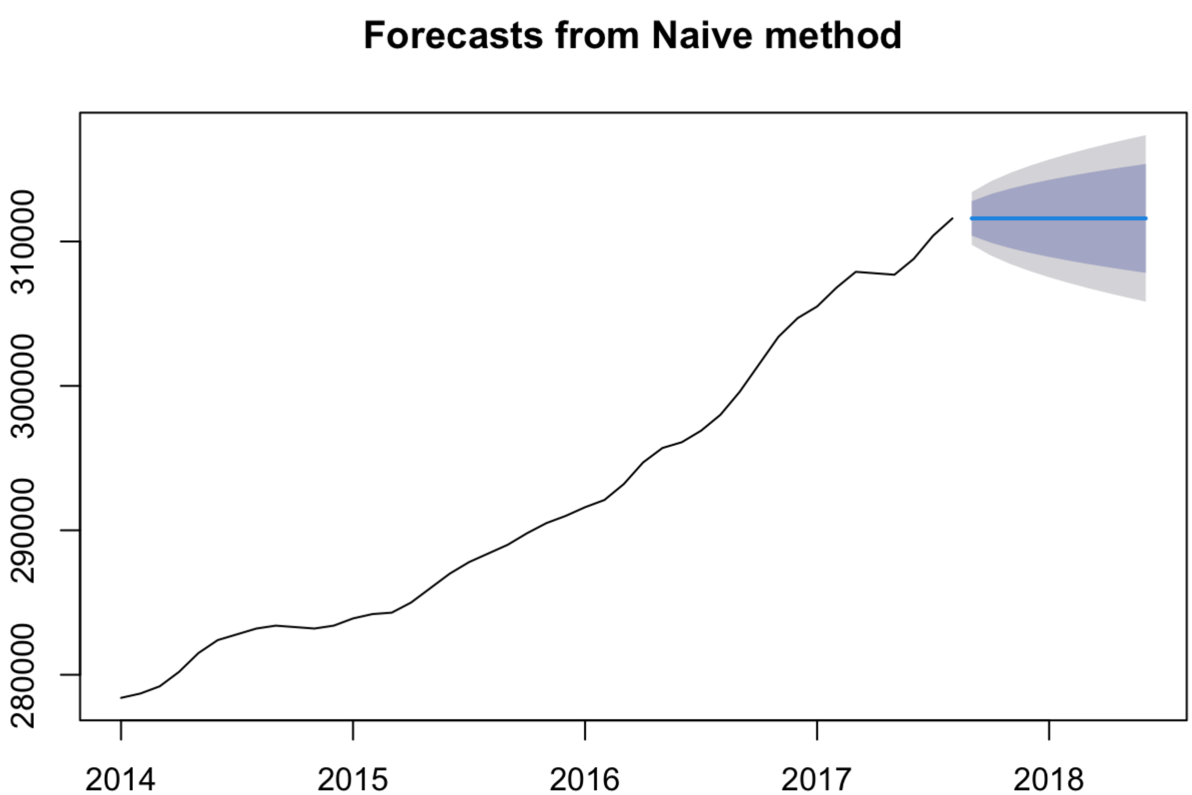
```
##                    ME     RMSE      MAE       MPE      MAPE       MASE      ACF1
## Training set 772.093 929.6161 790.6977 0.2615133 0.2678201 0.08548083 0.6470755
```

Forecast

```
forecast(naive_forecast)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Sep 2017         311600 310408.6 312791.4 309778.0 313422.0
## Oct 2017         311600 309915.2 313284.8 309023.3 314176.7
## Nov 2017         311600 309536.5 313663.5 308444.2 314755.8
## Dec 2017         311600 309217.3 313982.7 307956.0 315244.0
## Jan 2018         311600 308936.1 314263.9 307525.9 315674.1
## Feb 2018         311600 308681.8 314518.2 307137.0 316063.0
## Mar 2018         311600 308448.0 314752.0 306779.4 316420.6
## Apr 2018         311600 308230.4 314969.6 306446.6 316753.4
## May 2018         311600 308025.9 315174.1 306134.0 317066.0
## Jun 2018         311600 307832.6 315367.4 305838.3 317361.7
```

```
plot(forecast(naive_forecast))
```



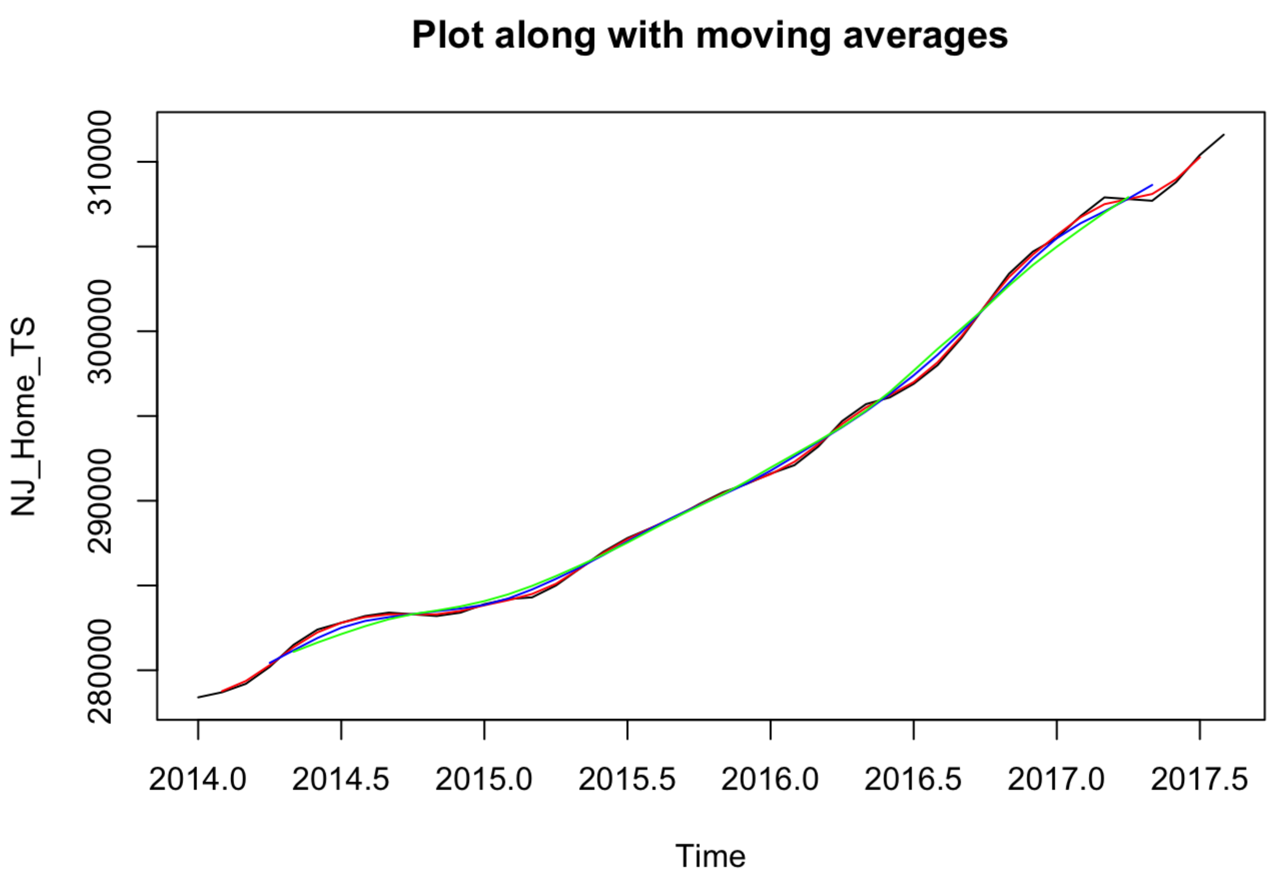
Naive Method Summary

- The ME, RMSE values are very high indicating that this method may not be the right one to go with.
- We can consider more forecasting techniques and check if the error values are less than this one.
- From 2014 to 2017 there is observed to be an increasing trend in the data. So, naive forecast may not be a right way to forecast.
- Rather, we can try naive method with drift component and that may yield us better forecast.

Simple Moving Averages

Simple Moving average of order 3, 6, and 9

```
mavg3_forecast = ma(NJ_Home_TS,order=3)
mavg6_forecast = ma(NJ_Home_TS,order=6)
mavg9_forecast = ma(NJ_Home_TS,order=9)
plot(NJ_Home_TS, main = "Plot along with moving averages")
lines(mavg3_forecast, col="Red")
lines(mavg6_forecast, col="Blue")
lines(mavg9_forecast, col="Green")
```

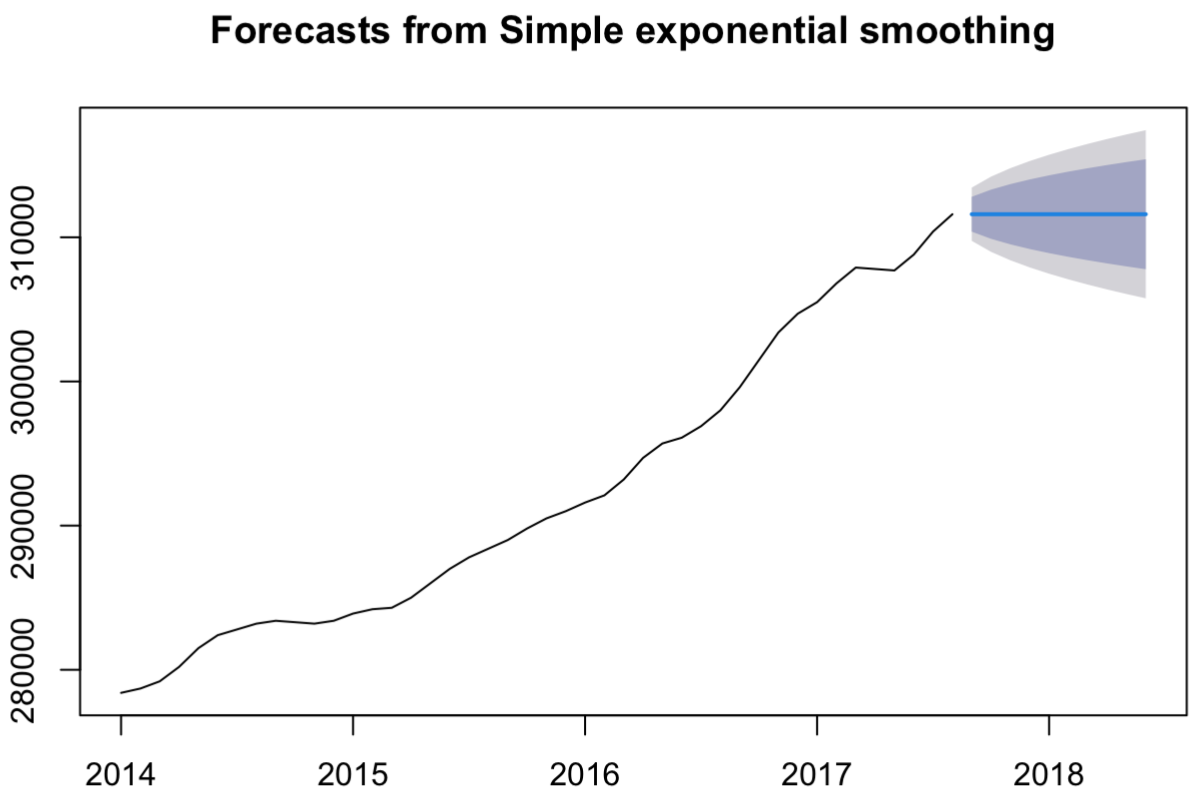


Observations

- From the plots, it is observed that the higher the order we consider, the smoother the moving average curve in the plot.
- It can be seen that the Green line above is the smoothest compared to Blue or Red lines.
- The Red line (order 3) gives the most real data compared to the other two. The higher order averages smoother the plot and do not give the actual values.

Simple Smoothing

```
ses_data <- ses(NJ_Home_TS)
plot(ses_data)
```




```
attributes(ses_data)
```

```
## $names
##  [1] "model"     "mean"      "level"     "x"         "upper"     "lower"
##  [7] "fitted"    "method"    "series"    "residuals"
##
## $class
## [1] "forecast"
```

Observations

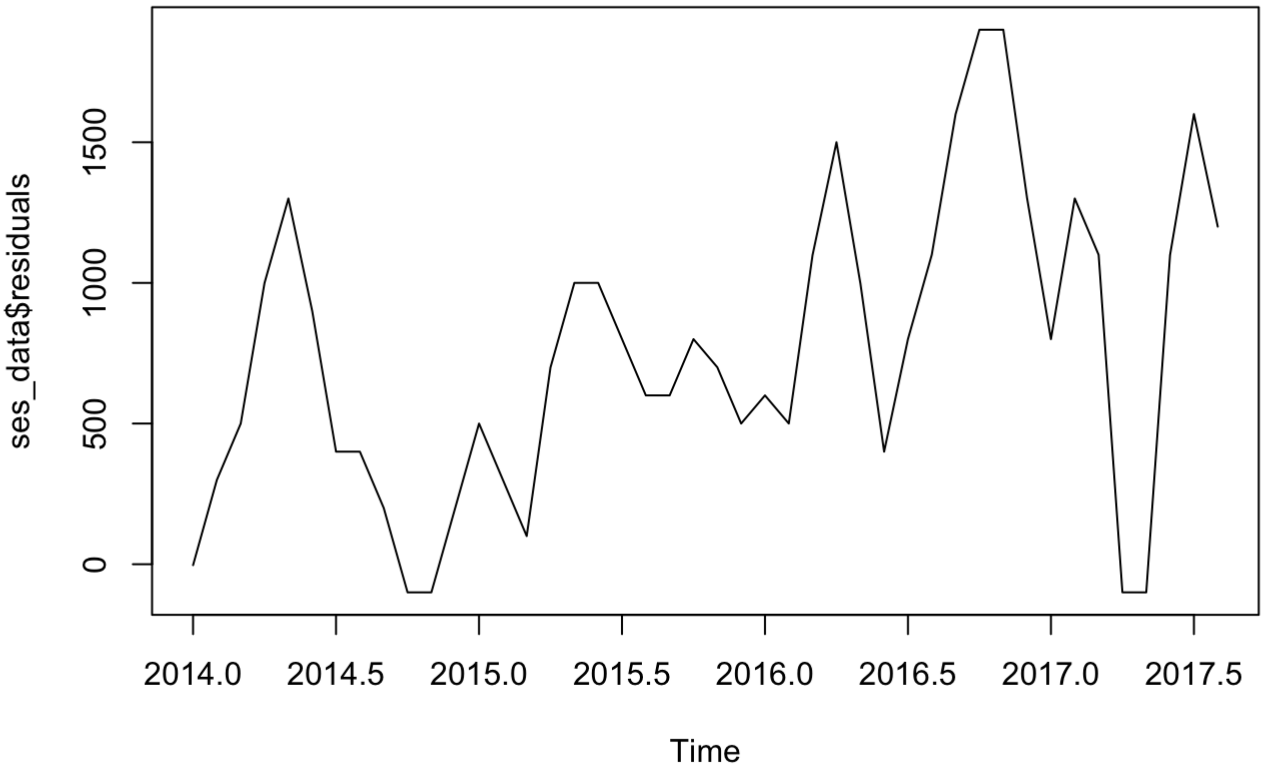
```
summary(ses_data)
```

```
##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
##  ses(y = NJ_Home_TS)
##
##   Smoothing parameters:
##     alpha = 0.9999
##
##   Initial states:
##     l = 278403.3349
##
##   sigma:  940.7005
##
##      AIC     AICc      BIC
## 772.9605 773.5605 778.3130
##
## Error measures:
##                    ME     RMSE     MAE       MPE      MAPE       MASE      ACF1
## Training set 754.5426 919.0724 772.871 0.2555673 0.2617836 0.08355362 0.6452862
##
## Forecasts:
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Sep 2017       311599.9 310394.3 312805.4 309756.1 313443.6
## Oct 2017       311599.9 309895.1 313304.7 308992.6 314207.2
## Nov 2017       311599.9 309511.9 313687.8 308406.6 314793.1
## Dec 2017       311599.9 309188.9 314010.8 307912.7 315287.1
## Jan 2018       311599.9 308904.4 314295.4 307477.5 315722.3
## Feb 2018       311599.9 308647.1 314552.6 307084.0 316115.7
## Mar 2018       311599.9 308410.6 314789.2 306722.2 316477.5
## Apr 2018       311599.9 308190.4 315009.4 306385.5 316814.3
## May 2018       311599.9 307983.5 315216.2 306069.2 317130.6
## Jun 2018       311599.9 307787.9 315411.8 305770.0 317429.8
```

- Alpha = 0.9999
- Alpha specifies the coefficient for the level smoothing. Values near 1.0 mean that the latest value has more weight.
- Initial state: l = 278403.3349
- Sigma: 940.7005. Sigma defines the variance in the forecast predicted.

Residual Analysis

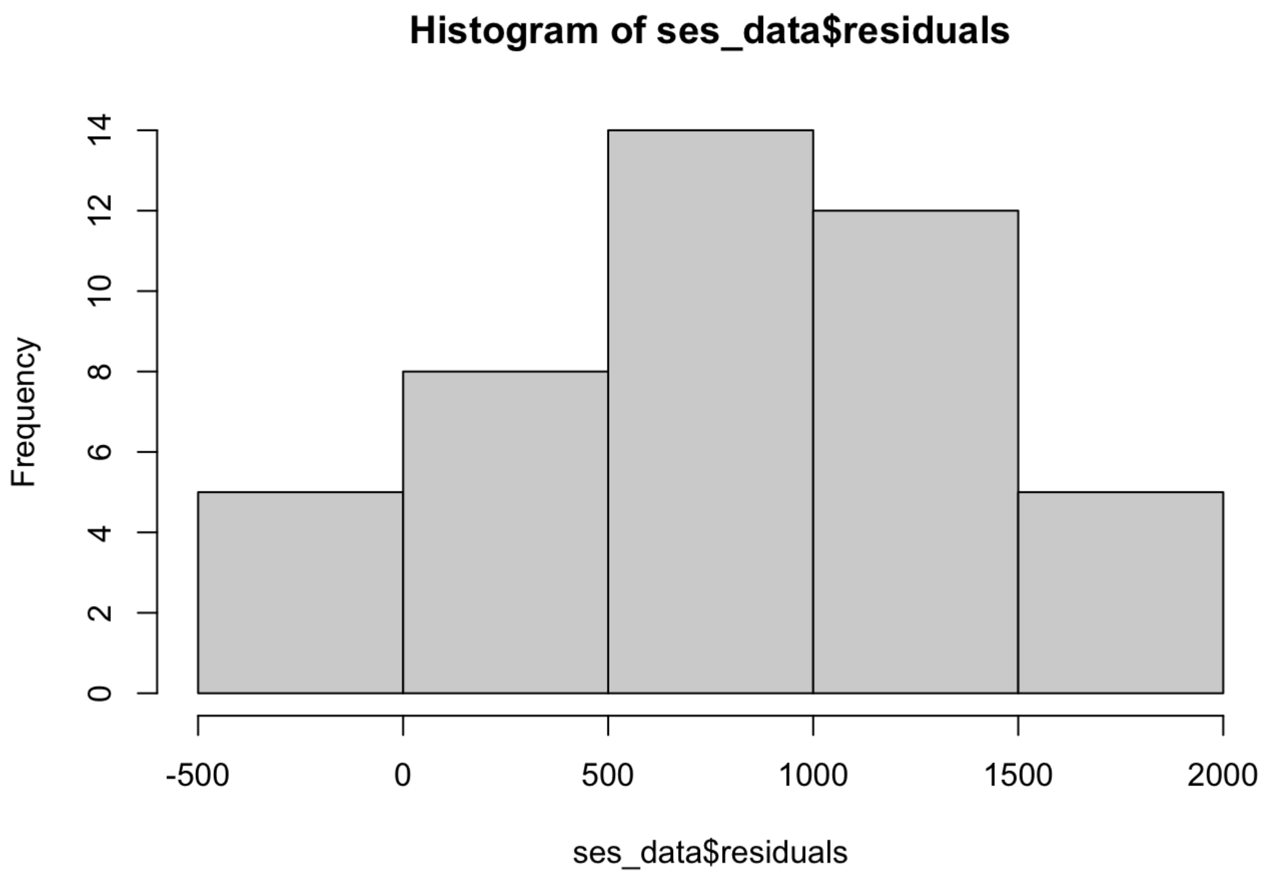
```
plot(ses_data$residuals)
```



- The residuals appear to have increasing positive values and then peaked in the third quarter of the year 2016 and then dipped down.
- Most of the residual values appear to be positive and do not have a mean of zero.

Histogram plot of residuals

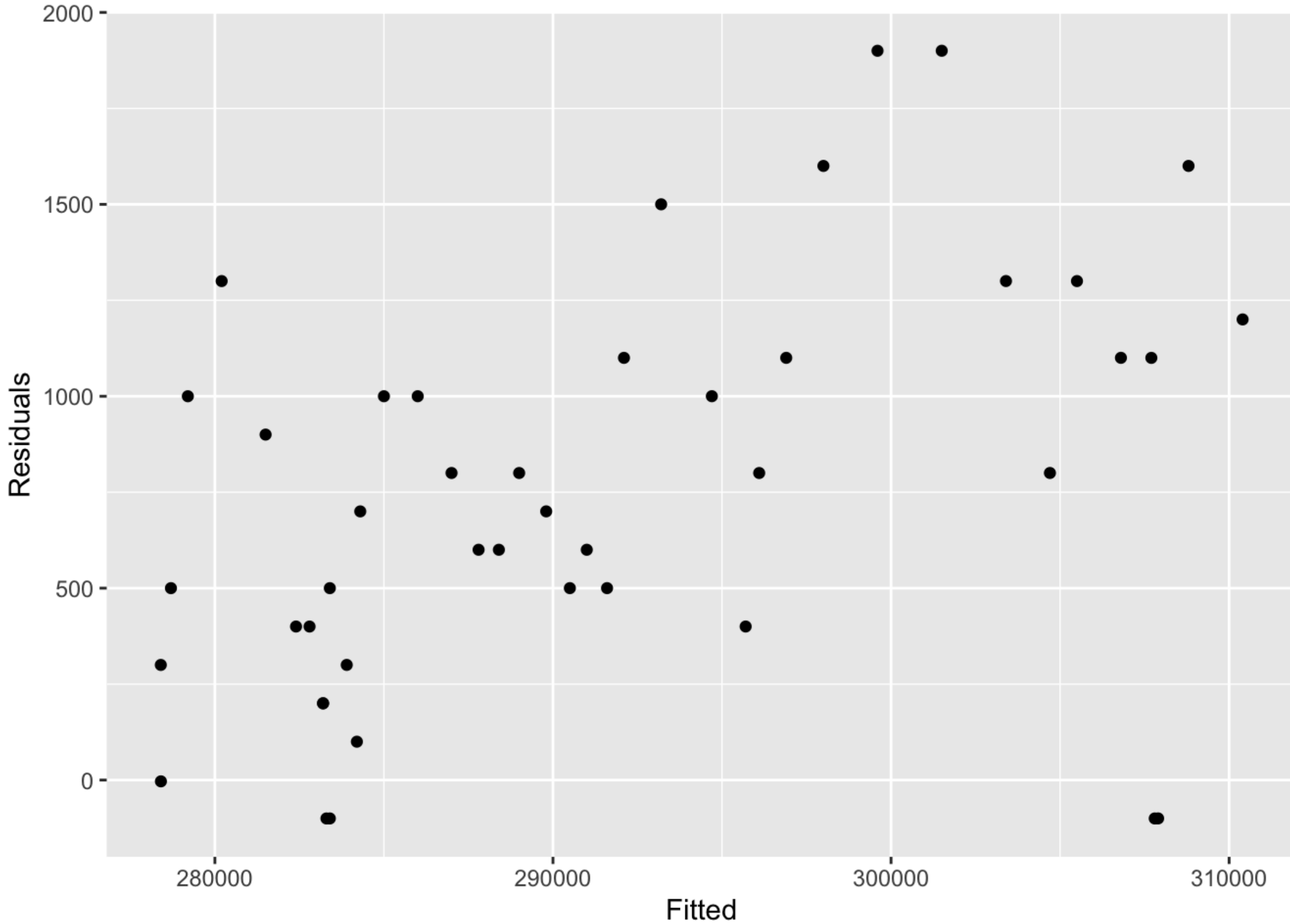
```
hist(ses_data$residuals)
```



- The histogram appears to be normally distributed.
- But the values do not have a mean zero. The histogram appears to be skewed on one side.
- If the residual histogram doesnot have the mean to be zero, it means the data is biased.

Fitted values vs. residuals

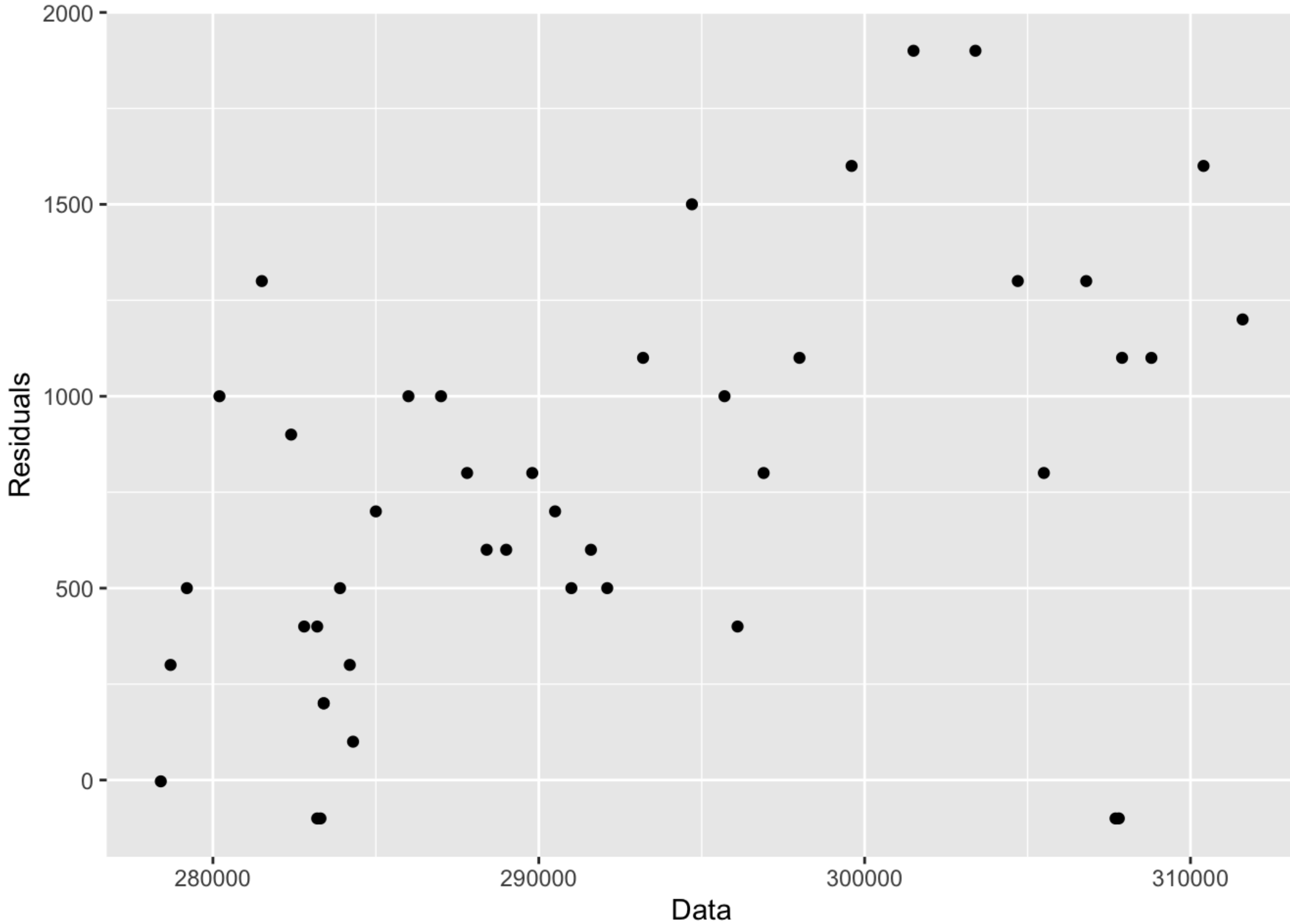
```
cbind(Fitted = fitted(ses_data),
      Residuals=residuals(ses_data)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Fitted, y=Residuals)) + geom_point()
```



- The Fitted vs Residuals plot appears to have a trend. The plot slightly shows a straight diagonal line pattern.
- This means there is heteroscedasticity in the errors which means that the variance of the residuals may not be constant.

Actual values vs. residuals

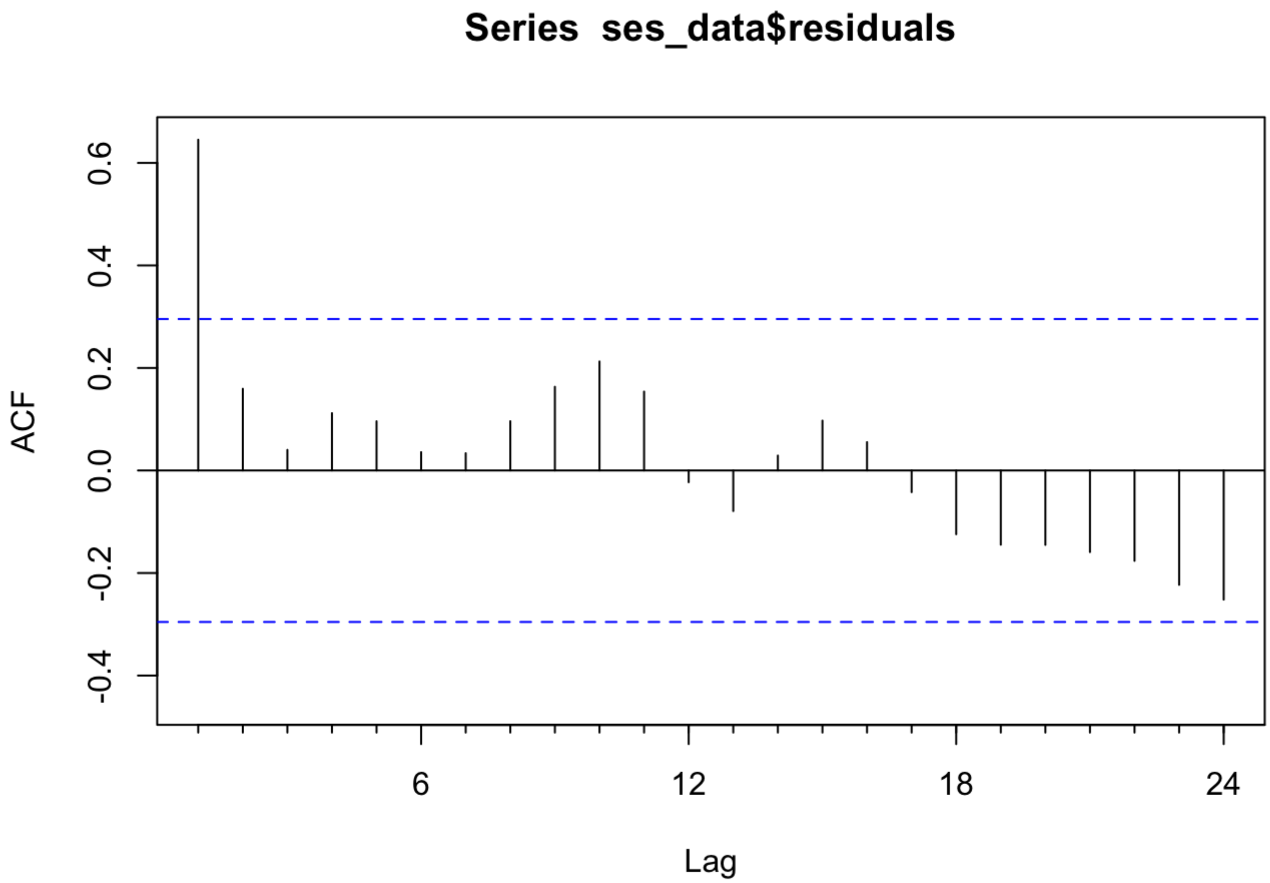
```
cbind(Data = NJ_Home_TS,
      Residuals=residuals(ses_data)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Data, y=Residuals)) + geom_point()
```



- Similar to the previous plot, the Actual vs. Residuals plot appears to have some trend in the data.

ACF plot of the residuals

```
Acf(ses_data$residuals)
```



- Values of the Acf have crossed the confidence level meaning there is a trend in the residuals and we have missed some variable in our forecast.
- The Acf values also show seasonality in the plot and we missed this variable too.
- Meaning that simple smoothing is missing some main variables which we have missed our consideration for the forecast.

Accuracy

```
accuracy(ses_data)
```

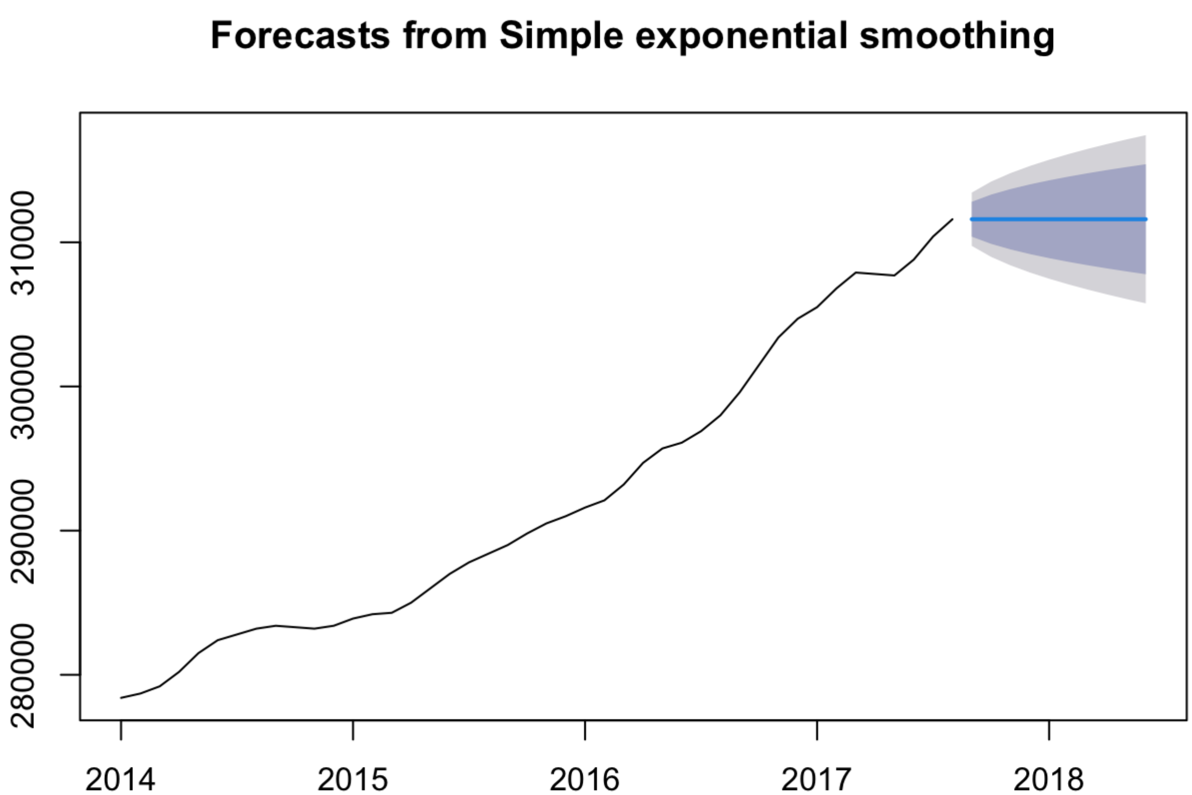
```
##                    ME     RMSE     MAE       MPE      MAPE       MASE      ACF1
## Training set 754.5426 919.0724 772.871 0.2555673 0.2617836 0.08355362 0.6452862
```

Forecast

```
ses_data
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Sep 2017       311599.9 310394.3 312805.4 309756.1 313443.6
## Oct 2017       311599.9 309895.1 313304.7 308992.6 314207.2
## Nov 2017       311599.9 309511.9 313687.8 308406.6 314793.1
## Dec 2017       311599.9 309188.9 314010.8 307912.7 315287.1
## Jan 2018       311599.9 308904.4 314295.4 307477.5 315722.3
## Feb 2018       311599.9 308647.1 314552.6 307084.0 316115.7
## Mar 2018       311599.9 308410.6 314789.2 306722.2 316477.5
## Apr 2018       311599.9 308190.4 315009.4 306385.5 316814.3
## May 2018       311599.9 307983.5 315216.2 306069.2 317130.6
## Jun 2018       311599.9 307787.9 315411.8 305770.0 317429.8
```

```
plot(ses_data)
```

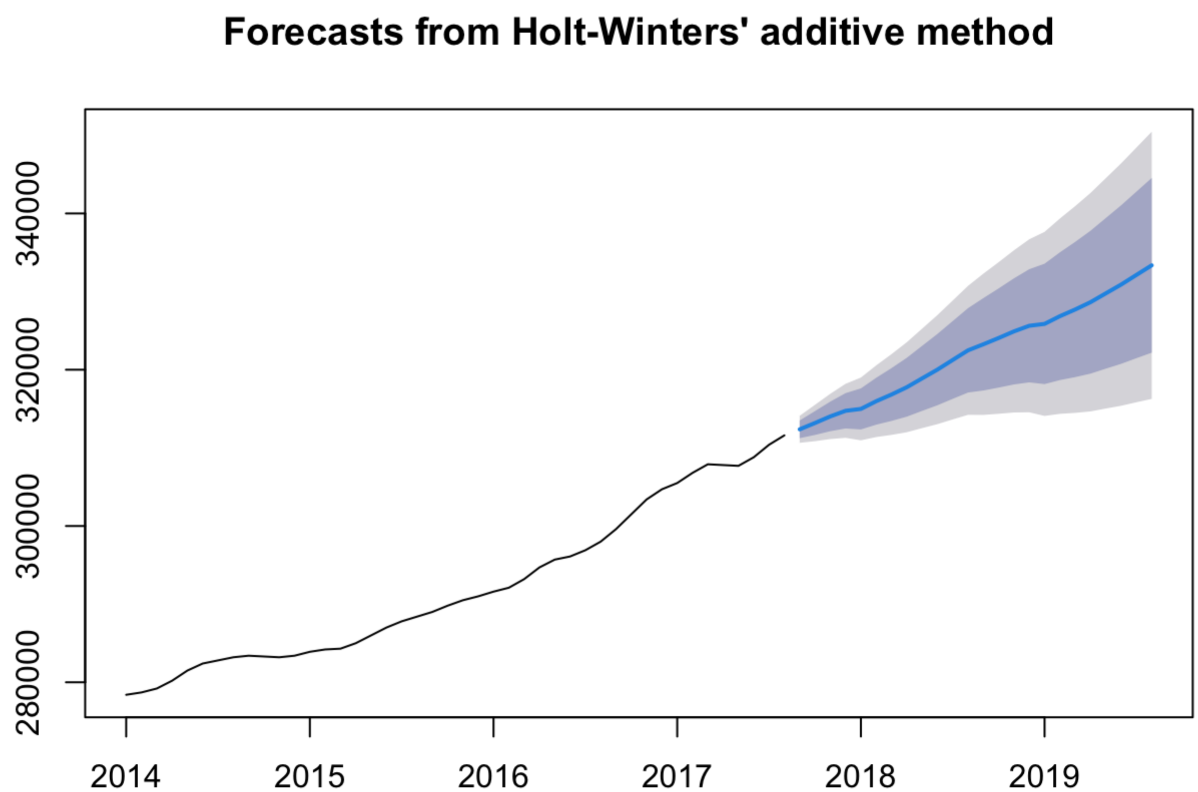


Simple Smoothing Summary

- The ME, RMSE values are very high indicating that this method may not be the right one to go with.
- We can consider more forecasting techniques and check if the error values are less than this one.
- From 2014 to 2017 there is observed to be an increasing trend in the data. So, this forecast may not be a right way to forecast.
- We can try Holtwinters approach as it suits for trend+seasonal time series.

Holt-Winters

```
HW_forecast <- hw(NJ_Home_TS, seasonal = "additive")
plot(forecast(HW_forecast))
```




```
attributes(HW_forecast)
```

```
## $names
##  [1] "model"     "mean"      "level"     "x"         "upper"     "lower"
##  [7] "fitted"    "method"    "series"    "residuals"
##
## $class
## [1] "forecast"
```

```
hw_add <- forecast(HW_forecast)
```

- Here, additive Holtwinters method is considered.
- This is because the seasonality isn’t increasing with trend. This is an additive time series.

Observations

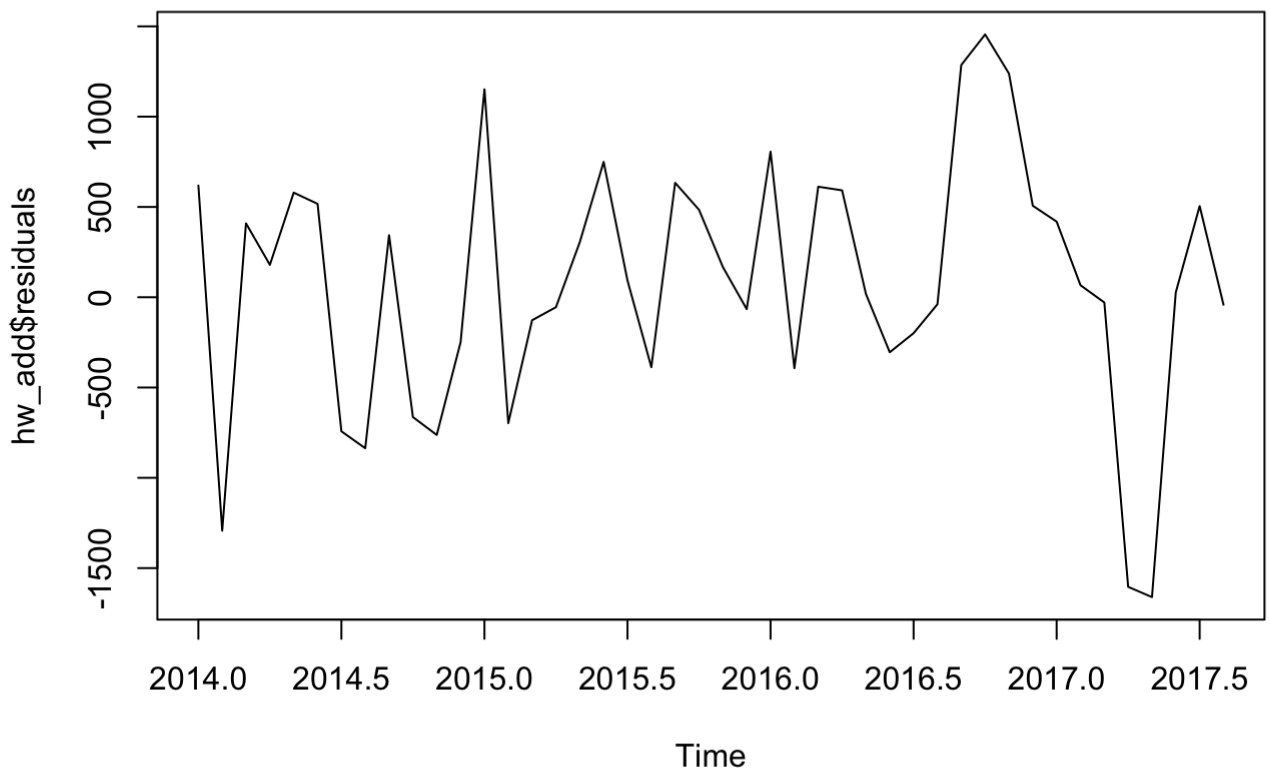
```
hw_add$model
```

```
## Holt-Winters' additive method
##
## Call:
##  hw(y = NJ_Home_TS, seasonal = "additive")
##
##   Smoothing parameters:
##     alpha = 0.8088
##     beta  = 0.0952
##     gamma = 0.1901
##
##   Initial states:
##     l = 278262.0665
##     b = 562.8731
##     s = 155.2537 257.43 199.9368 112.0836 924.0918 418.237
##            -164.0516 -47.7212 -248.0368 -609.1925 45.5616 -1043.592
##
##   sigma:  878.616
##
##      AIC     AICc      BIC
## 777.1116 800.6501 807.4428
```

- Alpha = 0.8088. Alpha specifies the coefficient for the level smoothing in Holtwinters.
- Beta = 00952. Beta specifies the coefficient for the trend smoothing in Holtwinters.
- Gamma = 0.1901. Gamma specifies the coefficient for the seasonal smoothing in Holtwinters.
- Values 1.0 means that the latest value has highest weight.
- Initial states: l = 278262.0665 b = 562.8731 s = 155.2537 257.43 199.9368 112.0836 924.0918 418.237 -164.0516 -47.7212 -248.0368 -609.1925 45.5616 -1043.592
- Sigma = 878.616. Sigma defines the variance of the forecast values.

Residual Analysis

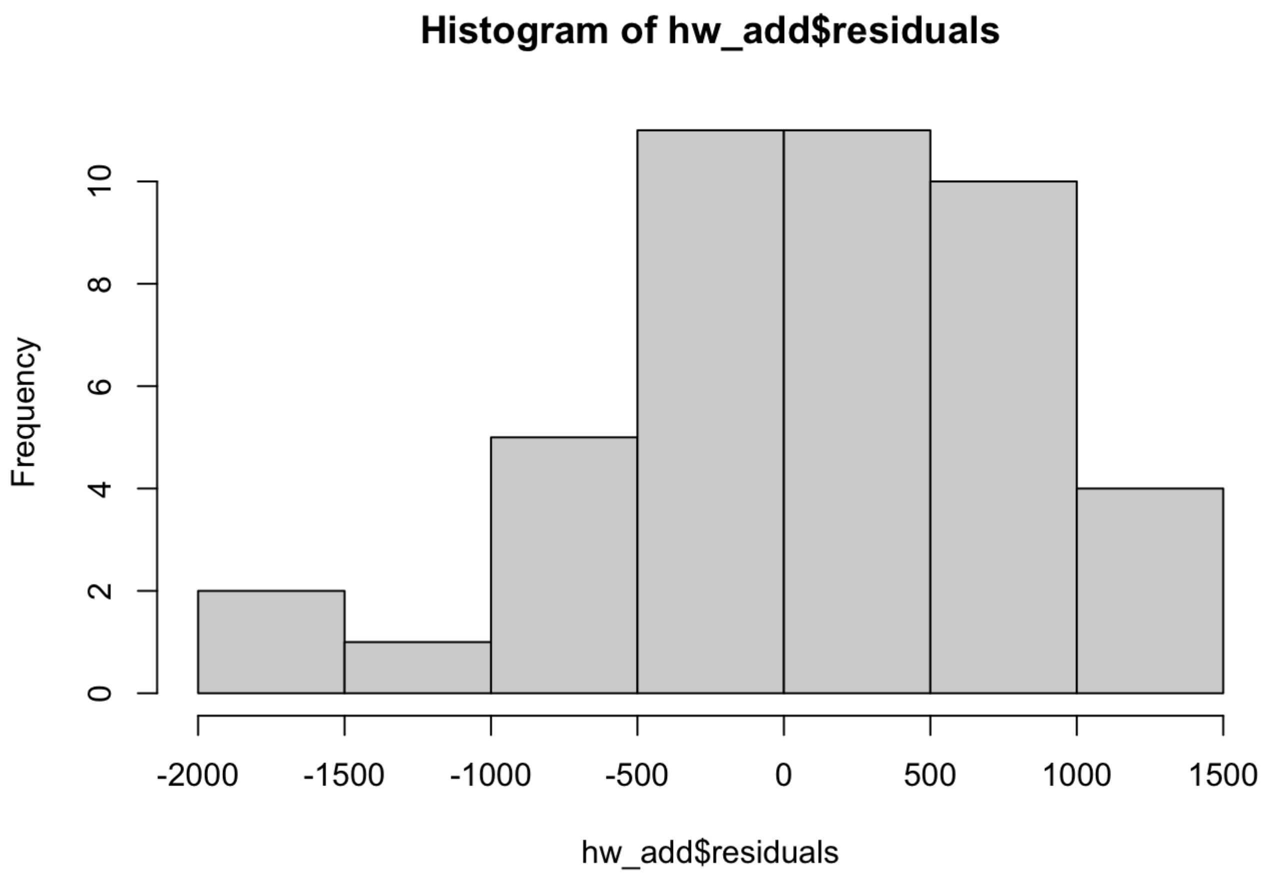
```
plot(hw_add$residuals)
```



- The residuals appear to be random and also the mean looks to be near zero. We can check this with histogram.
- We can observe a couple of up and downs throughout. But even they did not show and growing residual pattern.

Histogram plot of residuals

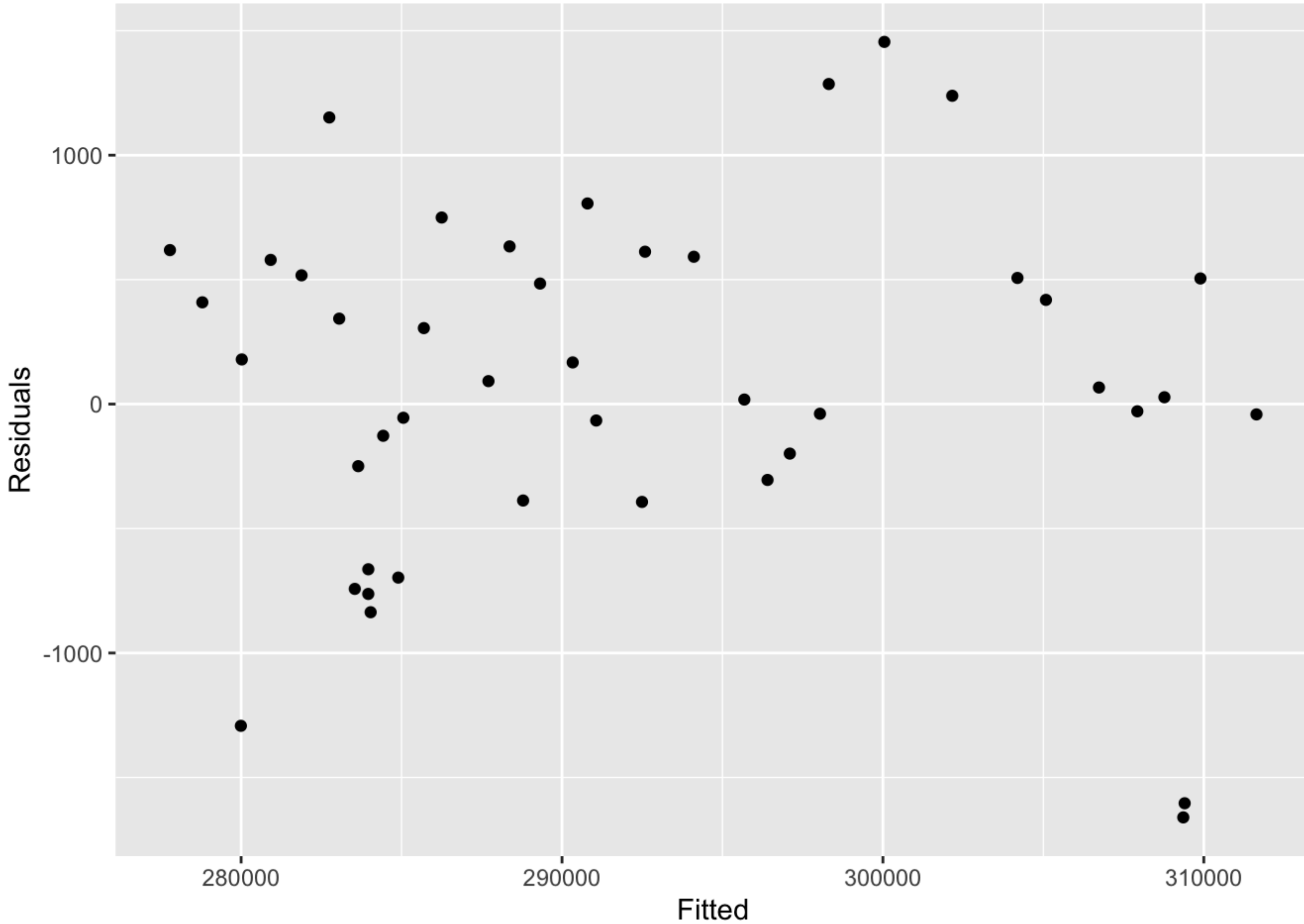
```
hist(hw_add$residuals)
```



- The histogram appears to be normally distributed.
- And the mean is near zero. Indicating the data is not biased.
- Overall, comparing the previous forecasts, this forecast appears to be the best till now.

Fitted values vs. residuals

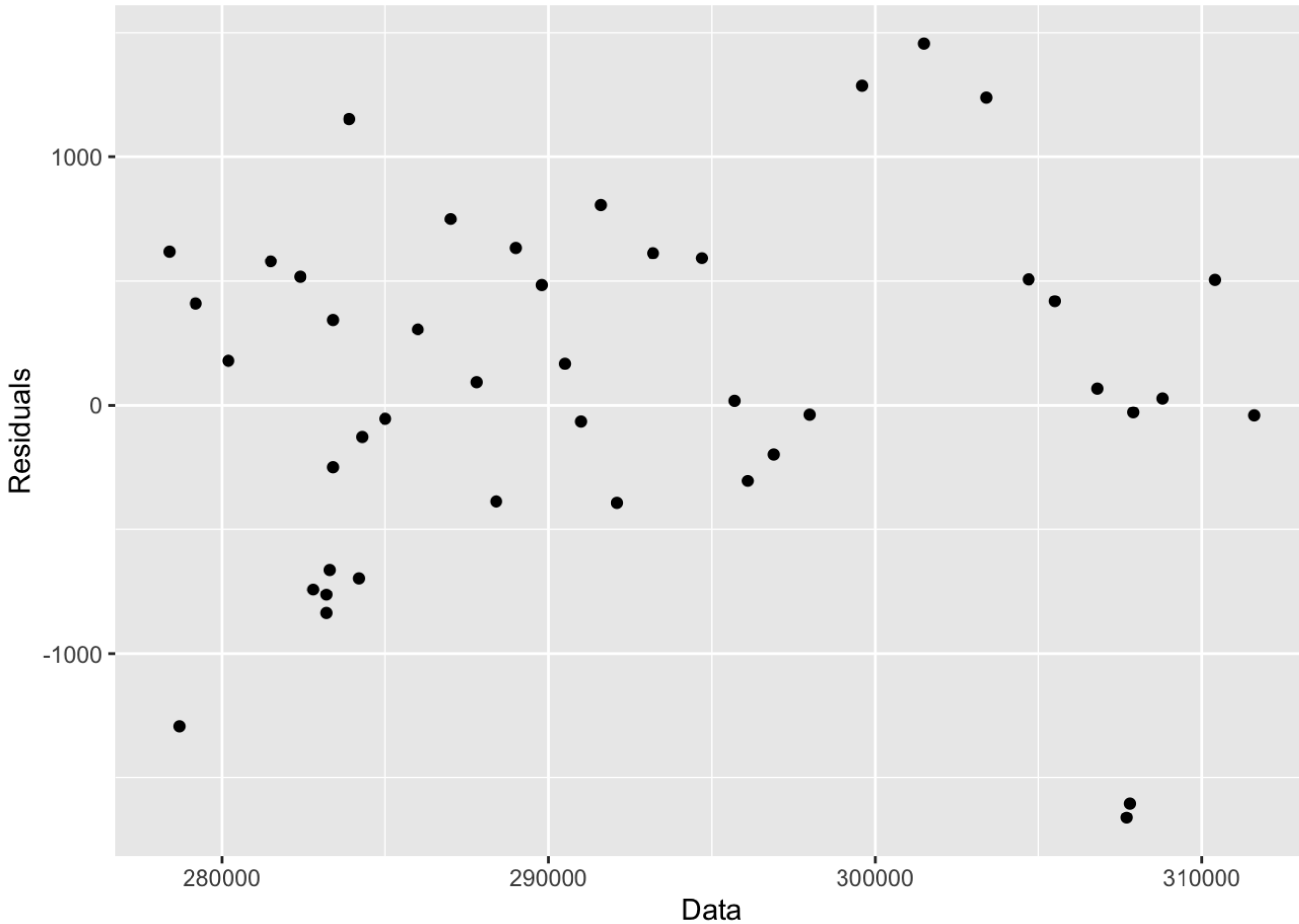
```
cbind(Fitted = fitted(hw_add),
      Residuals=residuals(hw_add)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Fitted, y=Residuals)) + geom_point()
```



- The Fitted vs Residuals plot appears not to have any trend.
- This means there is no heteroscedasticity in the errors which means that the variance of the residuals is constant.

Actual values vs. residuals

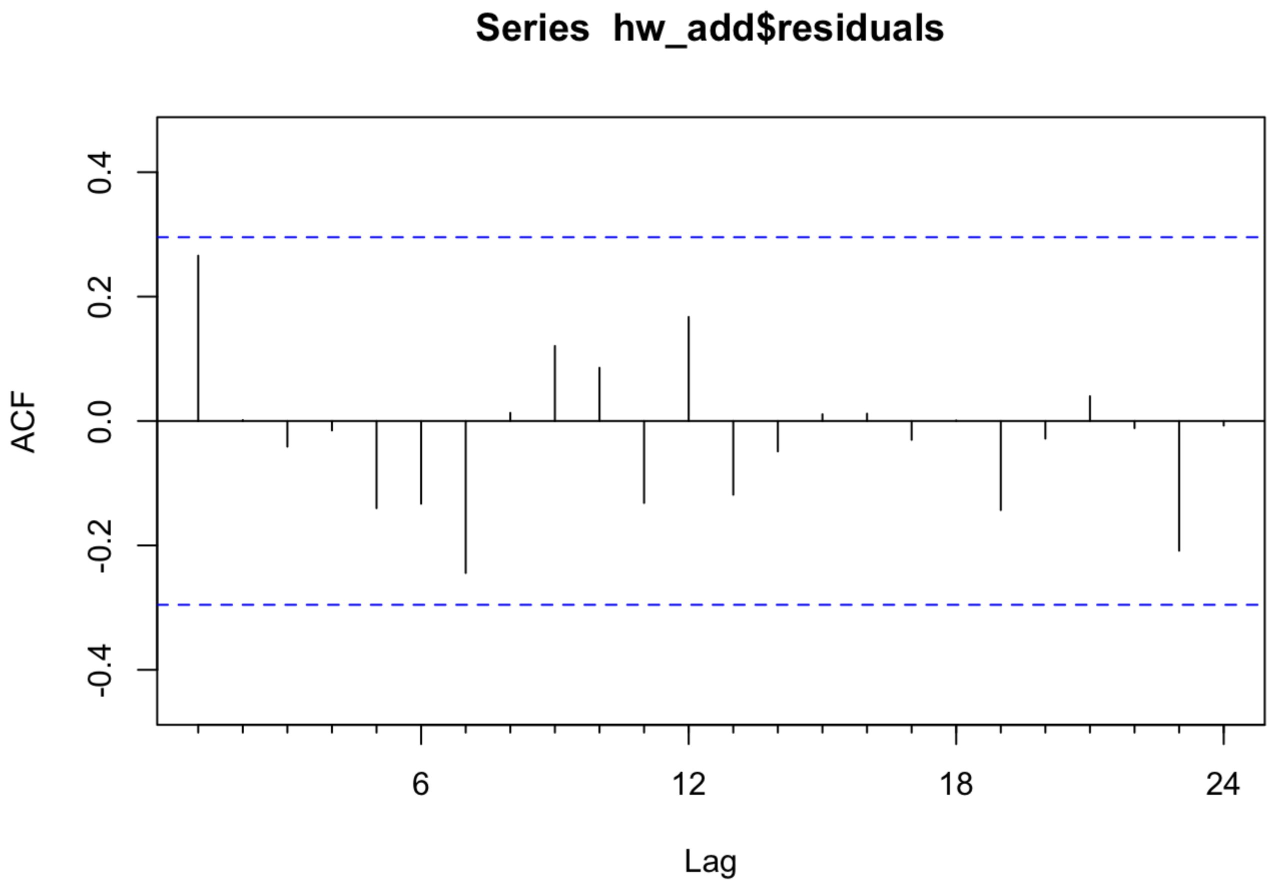
```
cbind(Data = NJ_Home_TS,
      Residuals=residuals(hw_add)) %>%
  as.data.frame() %>%
  ggplot(aes(x=Data, y=Residuals)) + geom_point()
```



- Similar to the previous plot, the Actual vs. Residuals plot appears to be random.

ACF plot of the residuals

```
Acf(hw_add$residuals)
```



- In the Acf plot, none of the values crossed the confidence levels. It appears to be white noice.
- This signifies that the forecast is a good forecast.
- This proves to be the best forecast comparing all the previous ones tested.

Accuracy

```
accuracy(hw_add)
```

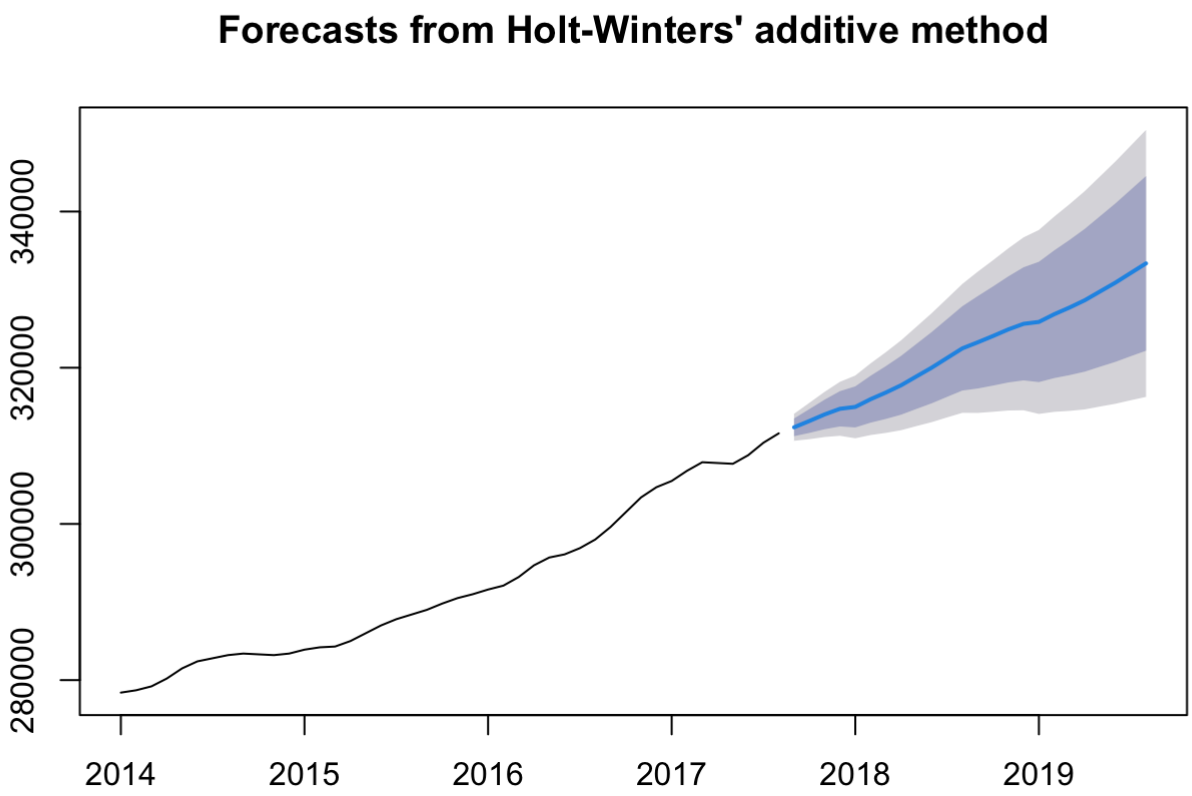
```
##                    ME     RMSE      MAE        MPE      MAPE       MASE
## Training set 82.11107 700.8931 543.4792 0.02774417 0.1859478 0.05875451
##                   ACF1
## Training set 0.2657767
```

Forecast

```
forecast(HW_forecast)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Sep 2017       312372.9 311247.0 313498.9 310650.9 314095.0
## Oct 2017       313179.9 311662.1 314697.7 310858.6 315501.2
## Nov 2017       314023.9 312134.6 315913.2 311134.4 316913.3
## Dec 2017       314742.5 312486.9 316998.1 311292.9 318192.2
## Jan 2018       314983.5 312360.2 317606.8 310971.5 318995.5
## Feb 2018       315969.5 312973.8 318965.2 311388.0 320551.0
## Mar 2018       316826.3 313451.7 320201.0 311665.3 321987.4
## Apr 2018       317761.1 313999.9 321522.3 312008.8 323513.3
## May 2018       318892.8 314736.8 323048.8 312536.7 325248.9
## Jun 2018       320015.4 315455.9 324574.9 313042.3 326988.6
## Jul 2018       321250.8 316279.0 326222.6 313647.1 328854.6
## Aug 2018       322481.0 317087.9 327874.1 314233.0 330729.0
## Sep 2018       323253.9 317346.5 329161.3 314219.3 332288.5
## Oct 2018       324060.9 317720.1 330401.6 314363.5 333758.2
## Nov 2018       324904.8 318121.1 331688.5 314530.1 335279.6
## Dec 2018       325623.5 318387.5 332859.5 314556.9 336690.0
## Jan 2019       325864.4 318166.8 333562.0 314092.0 337636.9
## Feb 2019       326850.5 318682.3 335018.7 314358.3 339342.7
## Mar 2019       327707.3 319059.5 336355.1 314481.7 340932.9
## Apr 2019       328642.0 319505.9 337778.2 314669.6 342614.5
## May 2019       329773.7 320140.7 339406.8 315041.2 344506.2
## Jun 2019       330896.4 320757.9 341034.9 315390.9 346401.9
## Jul 2019       332131.8 321479.5 342784.1 315840.5 348423.1
## Aug 2019       333362.0 322187.6 344536.3 316272.2 350451.7
```

```
plot(forecast(HW_forecast))
```



Holtwinters Summary

- The ME, RMSE values are quite low compared to any of our previous forecasts.
- Holwinters is a better forecast compared to naive and simple smoothing.
- Holtwinters appears to be the best forecast considering all the previous forecast methods.
- However, this forecast can still be improved as we can try forecasting using ARIMA models.

Accuracy Summary

```
accuracy(naive_forecast)
```

```
##                   ME     RMSE      MAE       MPE      MAPE       MASE      ACF1
## Training set 772.093 929.6161 790.6977 0.2615133 0.2678201 0.08548083 0.6470755
```

```
accuracy(ses_data)
```

```
##                    ME     RMSE     MAE       MPE      MAPE       MASE      ACF1
## Training set 754.5426 919.0724 772.871 0.2555673 0.2617836 0.08355362 0.6452862
```

```
accuracy(hw_add)
```

```
##                     ME     RMSE      MAE        MPE      MAPE       MASE
## Training set 82.11107 700.8931 543.4792 0.02774417 0.1859478 0.05875451
##                    ACF1
## Training set 0.2657767
```

Best & Worst Forecasts

- To start with, there is nothing like best or worst forecast.
- Considering the accuracy data above, HoltWinters forecast seems to fit the time series the best as it has the least error values.
- And naive forecast seems to be the worst as it has the largest ME and RMSE values.

Conclusion

- The data seemed to have trend and seasonality initially and we checked the same with Acf and confirmed it.
- Based on the three forecasting methods naive, simple smoothing, and HoltWinters, we can see that HoltWinters forecast provides to be the better forecasting method in this case.
- This is because the forecast fits perfectly and also the error values are quiet low for HoltWinters forecast.
- Additionally residuals in HoltWinters appear to be random and the all the ACF values of residuals are within the confidence interval.
- This shows that our hypothesis is correct based on the accuracy of all the models.
- Based on the analysis and forecast, the time series will increase over the next year and the next 2 years.