TIME SERIES ANALYSIS PROJECT

US HOUSING STARTS

Report By:
AKSHAY G BHAT (me2689)

Table of Contents

Summary

Time series analysis comprises methods for analyzing time series data to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values.

Time Series Analysis accounts for the fact that data points taken over time may have an internal structure (such as autocorrelation, trend, or seasonal variation) that should be accounted for. The main aim of time series modeling is to carefully collect and rigorously study the past observations of a time series to develop an appropriate model which describes the inherent structure of the series. The model is then used to generate future values for the series, i.e., to make forecasts.

Housing starts are the number of new residential construction projects that have begun in a month. US Housing starts is a non-stationary data, and we aim to forecast the housing starts for the future periods using different approaches and find the best model for our time series. To evaluate forecast accuracy as well as to compare among different models fitted to a time series, we have used the five performance measures, viz. MSE, MAD, RMSE, MAPE and Theil's U-statistics.

The following activities were accomplished:

The 8 steps of time series forecasting are applied to the Housing starts data to extract the best forecasting method. The goal is to select the best time series forecasting model to predict the Housing starts by utilizing the data from the past.

Several methods of forecasting like the Naive Forecast, moving average, Simple Exponential smoothing, Advanced exponential smoothing, multiple regression and Arima models are applied on the dataset and it was found that the trailing moving average model did better than the other models in predicting the future housing starts.

The tech stack used in this project involves R, R studio and excel.

Introduction

The goal of this study is to perform statistical analysis and Forecasting on the US Housing starts data which consists of monthly Housing start data from 1959 to 2017. The properties of the data are described, and basic time series techniques are applied to the data. Plots of the series, autocorrelation function and the forecast graphs are some of the graphical tools used to analyze the series. We also aim to fit different models to the data to make credible forecasts from the model. The data was downloaded from the **Forecast Chart website**, (https://www.forecast-chart.com/chart-housing-starts.html), from 1st January 1959 to 1st December 2017.

A year of data is 12 months data which equals to 708 data points overall for 59 years. Here are the steps followed to finalize the best forecasting model for the dataset.

- Define the Goal
- Get Data
- Explore and visualize the series
- Preprocess data
- Data Partition
- Apply Forecasting methods
- Evaluating and comparing model performance
- Implement Forecasts/system

Main Chapter

Motivation: We have chosen housing starts case as it is a leading indicator in the real estate or mortgage market. This forward-looking variable estimates a good gauge for future levels of real estate supply and creates a ripple effect in the overall economy. It is primarily of interest as the inception and collapse of the housing bubble in 2007-08 were the turning points in the subsequent developments that embroiled the American and the Eurozone economies in a deep-seated financial meltdown. Buying new houses also increases the demand of complementary durable goods such as furniture, refrigerators, etc. Thus, new residential construction boosts employment in construction, raw materials, banking, and other manufacturing sectors. Mortgage rates directly affect housing activity as higher interest rates raise the housing expenses. This lowers the number of qualified borrowers, declining home sales, and housing starts.

Data collection: The data is collected from https://www.forecast-chart.com/chart-housing-starts.html which involves monthly data from 1959 to 2017 (59 years). Totally, 708 time series data points are considered here. The dataset consists of 5 columns.

Time – Data point

Date – Date of capture of the housing data which is 1st of every month

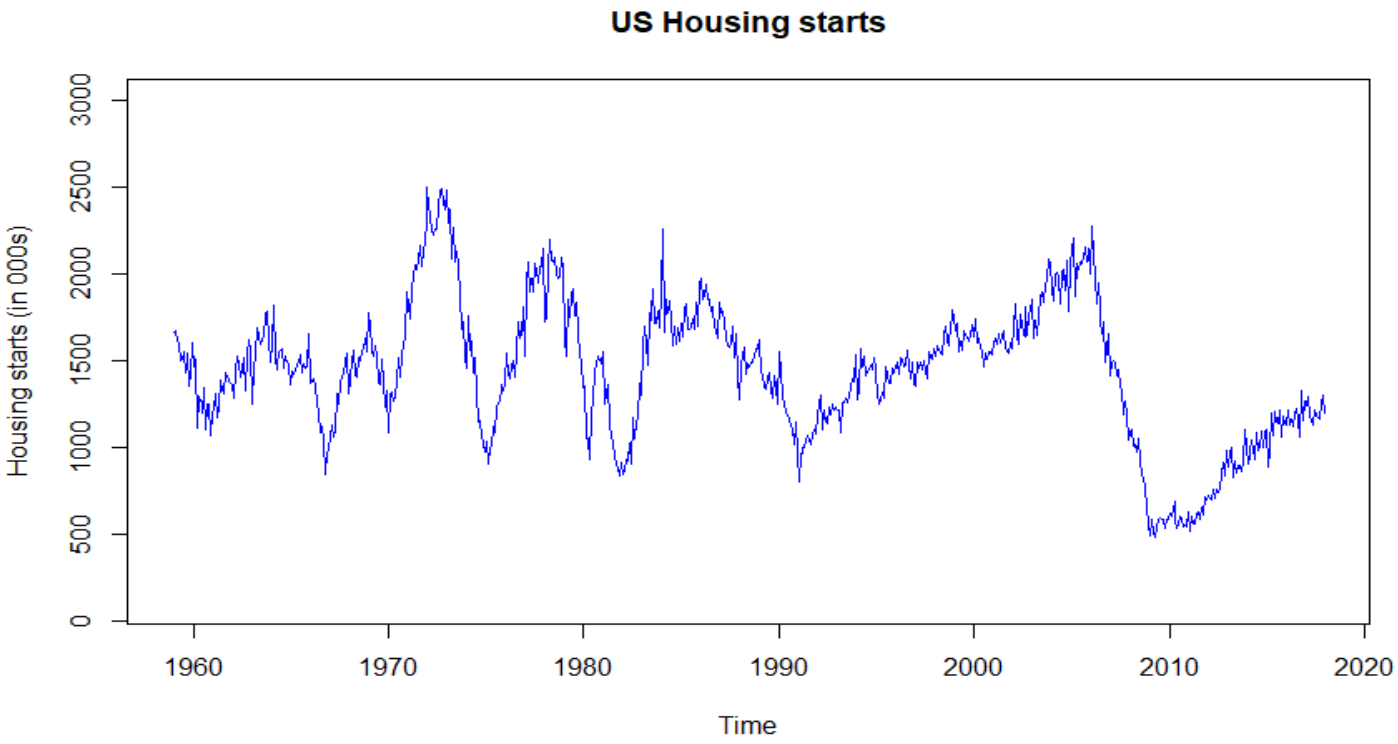
Month – Month of data recorded

Year – Year of data recorded

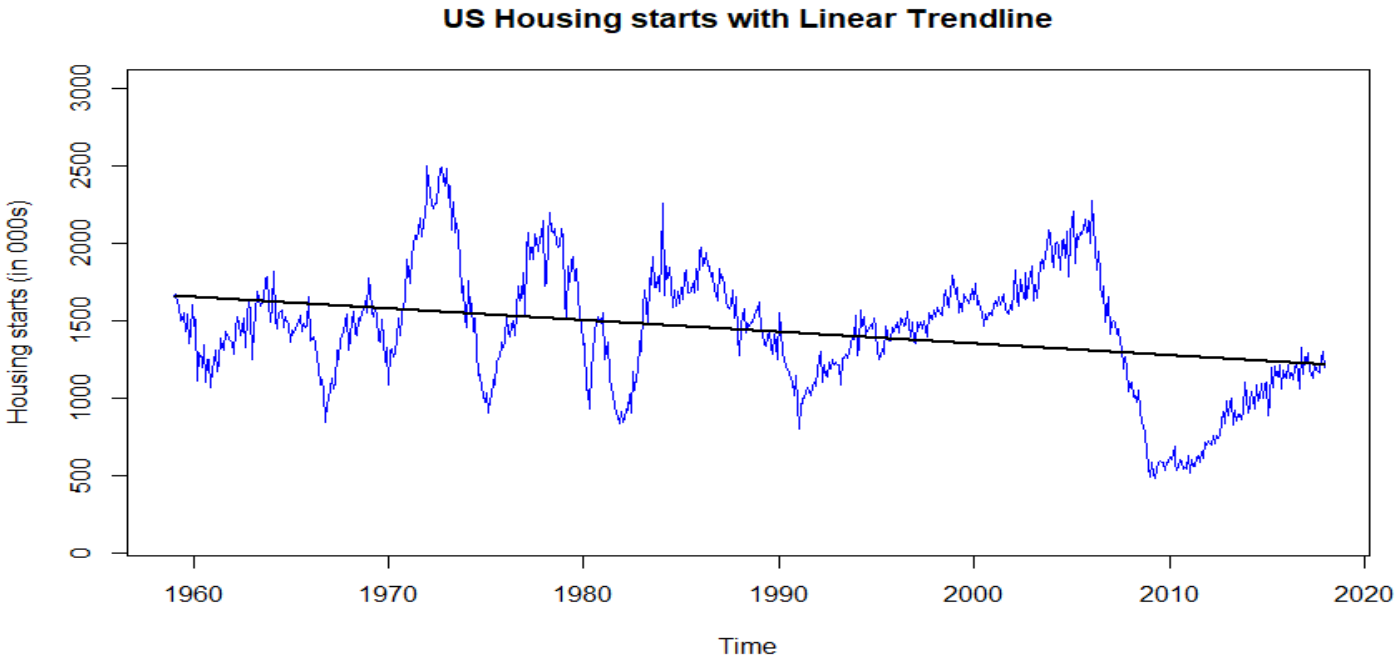
Housing starts - Number of Private housing projects started (recorded in thousands)

Explore and visualize series

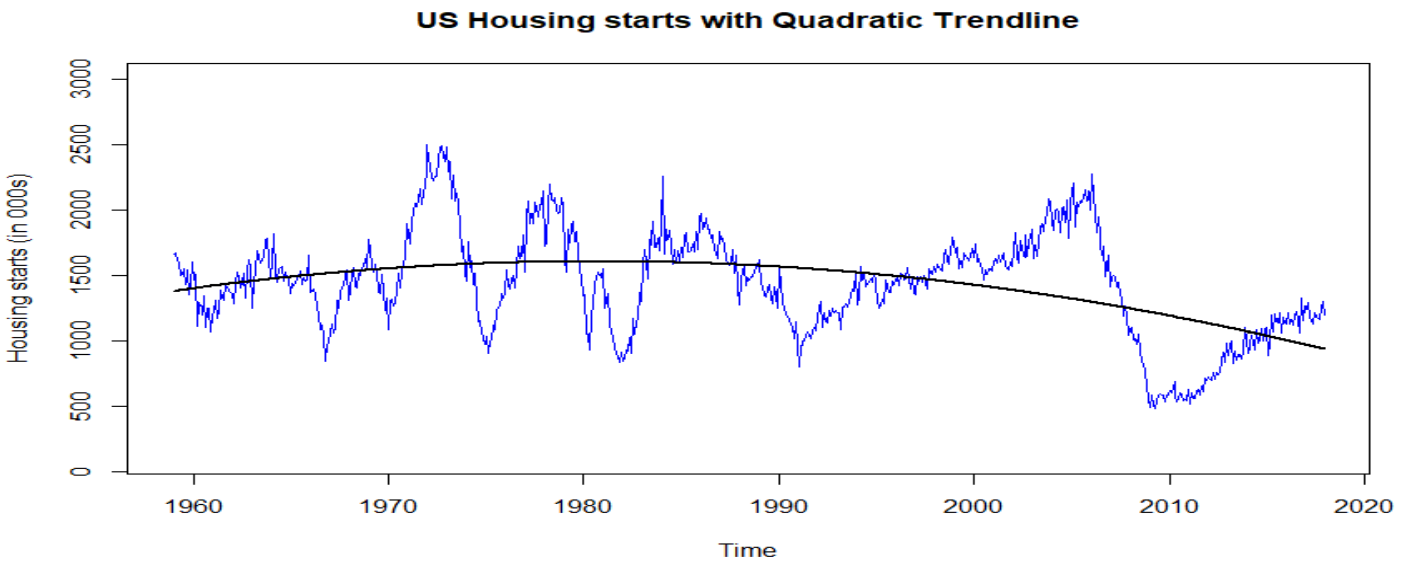
The datapoints from January 1959 to December 2017 is visualized using the continuous line graph.



Fitting a linear trend line to fit the series:



Fitting a Quadratic trend line to fit the series:




```
> round(accuracy(Housing.lin.pred$mean, valid.ts), 3)
                ME     RMSE      MAE      MPE    MAPE  ACF1 Theil's U
Test set -191.323 521.532 448.803 -37.427 50.951 0.978     8.894
> round(accuracy(Housing.quad.pred$mean, valid.ts), 3)
                ME     RMSE      MAE    MPE    MAPE  ACF1 Theil's U
Test set 180.163 453.485 388.013 0.316 34.414 0.971     5.273
> |
```

Observations:

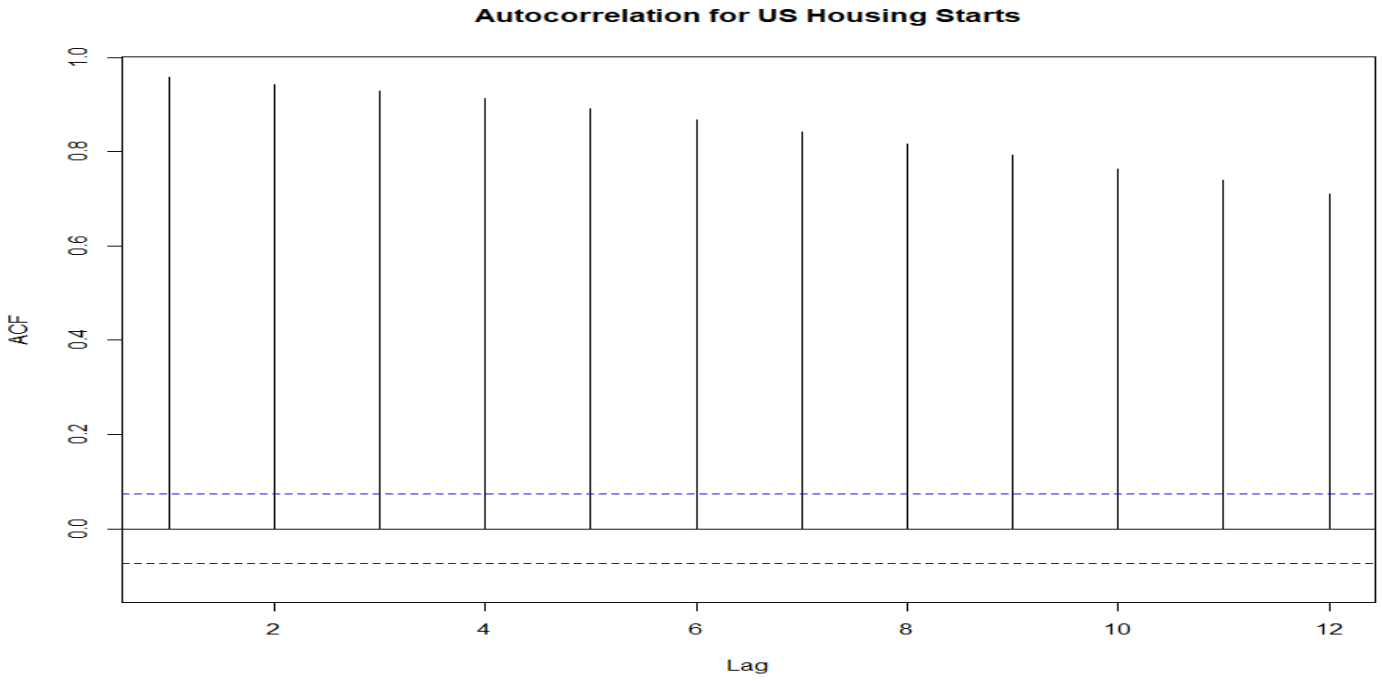
From the above monthly time series data graph, we cannot make any assumptions on the time series components like the trend and seasonality. The Housing starts had a steep growth from 1970 to 1974 and never reached that level thereafter. Also, since 2005 the housing sector has seen the lowest housing starts maybe due to the economic crisis which affected the housing sector very badly and is showing no signs of recovery for a while.

The time series fits well on the quadratic trend line comparatively on the linear trend line and this can be confirmed through the MAPE and RMSE accuracy scores for the two model. Most of the data is covered using a quadratic trend line and can be concluded that the time series is not linear. In other words, the series is neither growing linearly nor dropping linearly.

Also, from the time series graph it can be deduced that the time series is continuous in nature with no missing values and no potential outliers.

Identifying the Time series components in the historical data using autocorrelation:

Autocorrelation represents the correlation between a time series variable and the same variable lagged one or more periods.



Observations:

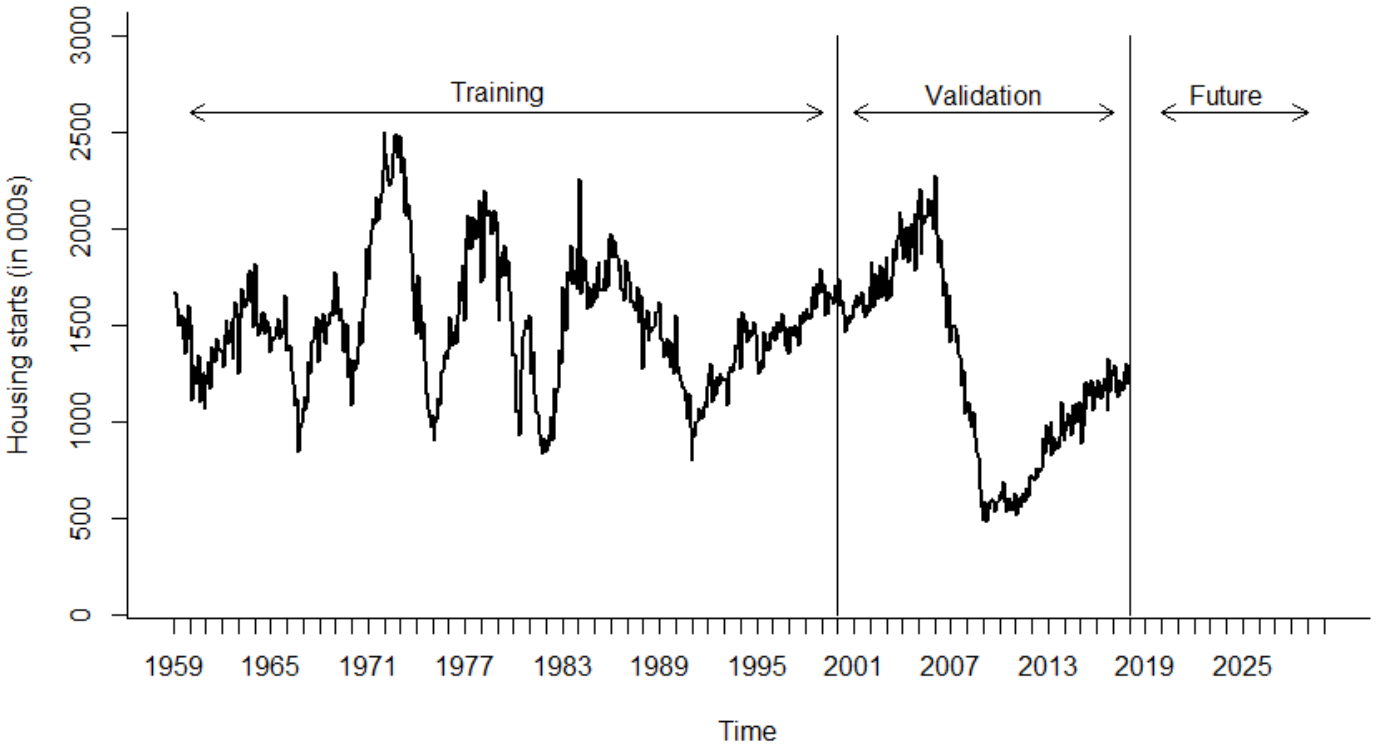
- A positive autocorrelation coefficient in lag 1 is substantially higher than the horizontal threshold which is indicative of a trend component in the Housing dataset.
- A positive autocorrelation coefficient in lag 12, which is also statistically significant points to monthly seasonality.
- As all the autocorrelation coefficients are significantly significant it can be said that the data is not random in nature and can be predicted

Data Preprocessing and data partition

Data preprocessing involves checking for missing values, outliers, irrelevant periods in the data and data entry errors. The time series looks clean and does not require any data cleaning and is ready to be used for forecasting.

Data partition involves splitting the data into training and validation data. Data partition comes from the need to be able to test how well any selected model performs with the new data not included in the model development. By having a validation set for forecasting overfitting can be minimized and usually the model does well on new data.

Here, we have used monthly data for the last 18 years as our validation data which is 216 data points and a training data consisting of remaining 492 data points. The ratio of training to validation dataset is 70:30. The data partition for the time series is shown below.



Training data – January 1959 to December 1999

Validation data – January 2000 to December 2017

The dataset is now ready to be used for forecasting.

Forecasting methods

We will be applying several time series forecasting techniques to decide on the best model for future forecasting. Some of the techniques which we have implemented are:

- Naïve Forecast
- Moving average
- Two level Forecasting
- Simple Exponential smoothing
- Multiple regression models
- Autoregressive and ARIMA models

Naïve Forecast and seasonal naïve forecast

Naïve forecast is a simple forecasting method that uses the most recent value of the time series and seasonal naïve forecast for a seasonal time series is the value of the most recent identical season. The naïve forecast can be used as a benchmark to compare it with the more advanced methods.

The naïve and seasonal naïve forecast technique is applied on the entire dataset and the accuracy for both the model is derived to track model performance. Over the entire project we concentrate on the MAPE and RMSE scores to judge the model performance.

RMSE – Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit.

MAPE – Mean absolute percentage error gives an absolute percentage score of how forecast deviates (on the average) from actual values; useful for comparing performance across series of data that have different scales.

Accuracy measures for Naïve and seasonal naïve forecast model:

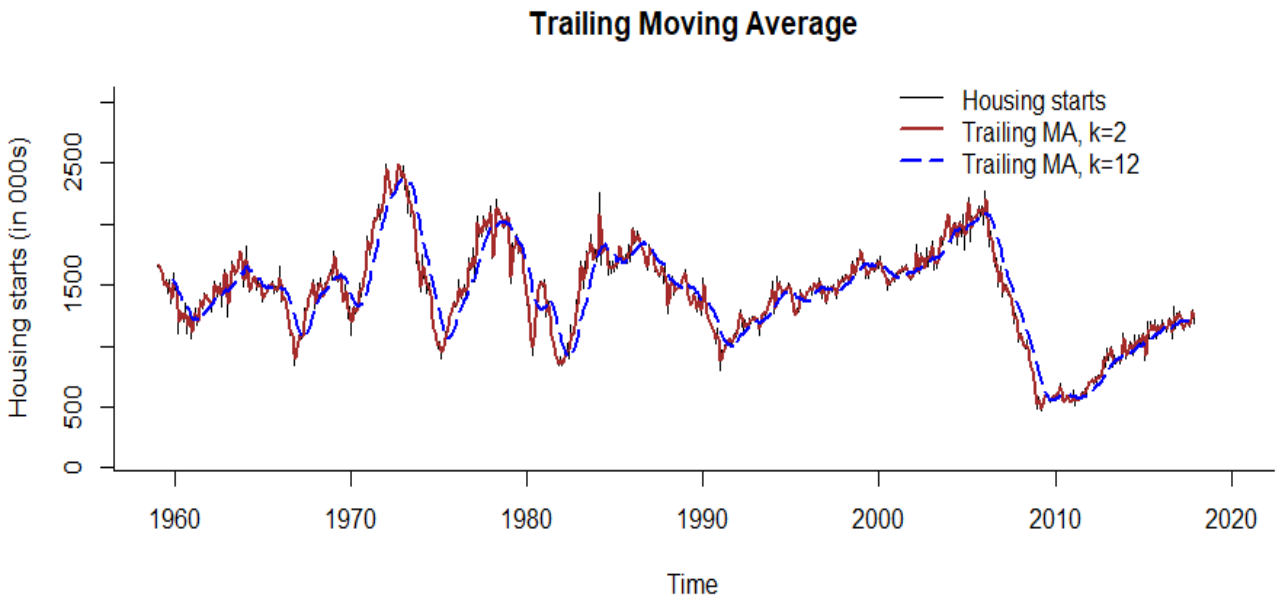
```
> round(accuracy(Housing.naive.pred$fitted, Housing.ts), 3)
               ME    RMSE    MAE    MPE  MAPE  ACF1 Theil's U
Test set -0.658 114.261 86.556 -0.381 6.294 -0.32         1
> round(accuracy(Housing.snaive.pred$fitted, Housing.ts), 3)
               ME    RMSE    MAE    MPE   MAPE  ACF1 Theil's U
Test set -5.675 302.829 228.25 -3.105 17.613 0.839     3.008
>
```

Observations: It is seen that the naïve forecast with an MAPE score of 6.294 and RMSE score of 114.261 does better than the seasonal naïve forecast which has a higher RMSE and MAPE scores.

Moving average

Moving average is a smoothing method that smooth out the noise in the times series to uncover data patterns. The term "moving" represents the fact that as each new actual data point becomes available, a revised MA is computed for the next data period requiring forecasting. The most common applications of moving averages are to identify trend direction and to determine support and resistance levels. Centered moving average and trailing moving average are the two main simple moving average methods. Centered moving average is useful to visualize trends, because the averaging operation can suppress seasonality and noise, making the trend more visible. Trailing moving average is the most popular type of weighted moving average method used in forecasting.

Here we have used trailing moving average with different weights of 2, 5 and 12 to smooth out the data and the plot is generated involving the original data, trailing MA with window width of 2 and trailing MA with window width of 12. Window width with larger value will make the forecast very smooth and expose more global trends while the window width with smaller value will expose local trends.



Accuracy measures for trailing MA with window width of 2, 5 and 12:

```
> round(accuracy(ma.trailing_2, Housing.ts), 3)
               ME   RMSE    MAE    MPE  MAPE  ACF1 Theil's U
Test set -0.329 57.13 43.278 -0.191 3.147 -0.32       0.5
> round(accuracy(ma.trailing_5, Housing.ts), 3)
               ME    RMSE    MAE    MPE  MAPE  ACF1 Theil's U
Test set -1.103 94.645 72.401 -0.505 5.326 0.415     0.847
> round(accuracy(ma.trailing_12, Housing.ts), 3)
               ME    RMSE     MAE    MPE  MAPE  ACF1 Theil's U
Test set -2.185 160.059 120.025 -1.327 9.097 0.768     1.544
> |
```

Observations:

- Trailing MA with window width of 2 fits well and exposes more local trend in the data. It has an MAPE value of 3.147 and RMSE value of 57.13 which is the lowest compared to other weights.

Two level forecasting – Quadratic regression model + Trailing MA for residuals

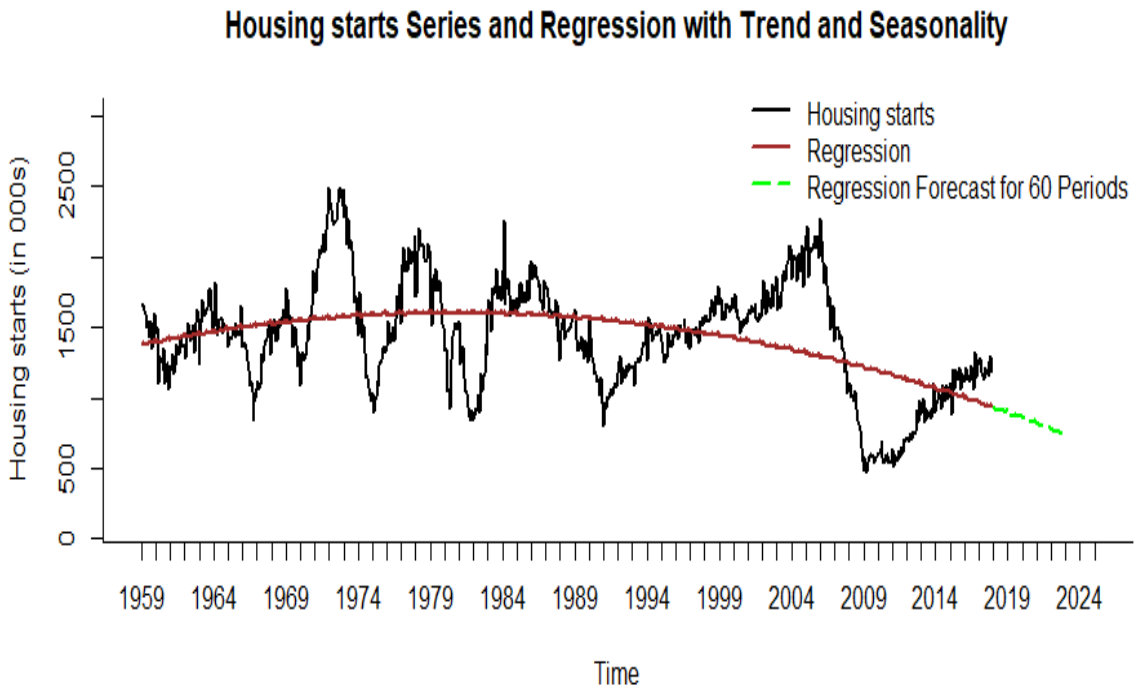
In general, Trailing MA should be used for forecasting in time series that lack trend and seasonality. To apply trailing MA for the series with trend and seasonality we should first de-trend and/or de-seasonalize the data. This is where the two-level forecasting method comes into picture.

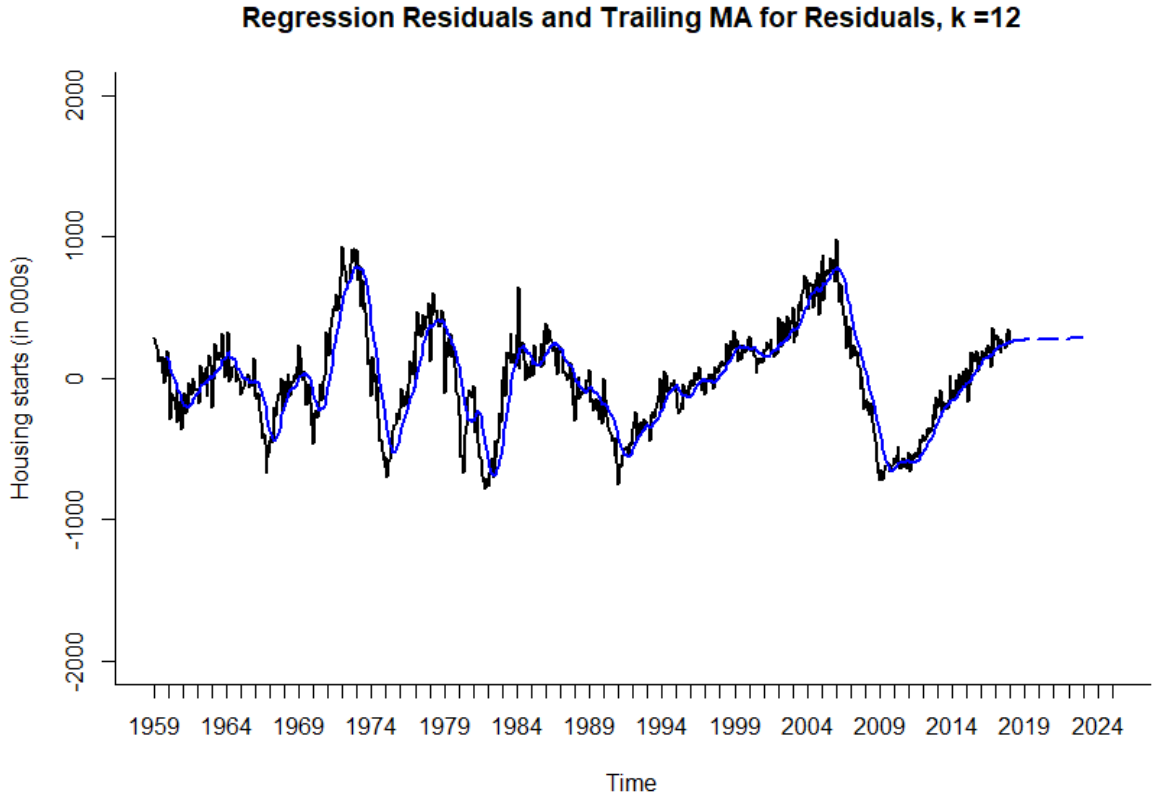
Here we have used the regression approach where the quadratic regression model with trend and seasonality is used to remove the trend and seasonality component in the data. Further, regression residuals are forecasted using the trailing MA method.

```
Call:
tslm(formula = Housing.ts ~ trend + I(trend^2) + season)

Residuals:
    Min      1Q  Median      3Q     Max
-778.69 -225.72   -4.49  207.08  985.64

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.369e+03  6.003e+01  22.805  < 2e-16 ***
trend        1.753e+00  2.633e-01   6.658 5.65e-11 ***
I(trend^2)  -3.358e-03  3.596e-04  -9.338  < 2e-16 ***
season2      2.410e+01  6.583e+01   0.366    0.714
season3     -8.217e-01  6.583e+01  -0.012    0.990
season4     -2.282e+00  6.583e+01  -0.035    0.972
season5     -1.989e+00  6.583e+01  -0.030    0.976
season6      1.573e-01  6.583e+01   0.002    0.998
season7      1.129e+01  6.583e+01   0.172    0.864
season8      3.606e+00  6.583e+01   0.055    0.956
season9      5.519e+00  6.583e+01   0.084    0.933
season10     5.624e+00  6.583e+01   0.085    0.932
season11     1.867e+01  6.583e+01   0.284    0.777
season12     1.752e+01  6.583e+01   0.266    0.790
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 357.5 on 694 degrees of freedom
Multiple R-squared:  0.2048,    Adjusted R-squared:  0.1899
F-statistic: 13.75 on 13 and 694 DF,  p-value: < 2.2e-16
```

The regression model contains 11 seasonal binary (dummy) variables for February (season2) through December (season12). The regression equation is:

$$Yt = 1369 + 1.753\ t - 0.003358t^2 - 24.10\ D2 - 0.8217\ D3 + \ldots + 17.52\ D12$$


Housing starts Series and Regression with Trend and Seasonality
Housing starts
Regression
Regression Forecast for 60 Periods
Housing starts (in 000s)
0
500
1500
2500
1959
1964
1969
1974
1979
1984
1989
1994
1999
2004
2009
2014
2019
2024
Time


Regression Residuals and Trailing MA for Residuals, k =12
Housing starts (in 000s)
-2000
-1000
0
1000
2000
1959
1964
1969
1974
1979
1984
1989
1994
1999
2004
2009
2014
2019
2024
Time

Accuracy measures for trailing MA with window width of 2, 5 and 12:

```
> round(accuracy(reg.trend.seas.pred$fitted, Housing.ts), 3)
         ME    RMSE     MAE    MPE   MAPE  ACF1 Theil's U
Test set  0 353.971 276.897 -7.907 22.985 0.948     4.162
> round(accuracy(reg.trend.seas.pred$fitted+ma.trailing.res_12, Housing.ts), 3)
            ME    RMSE     MAE    MPE  MAPE  ACF1 Theil's U
Test set 1.315 159.887 120.622 -0.969 9.155 0.771     1.543
>
```

Observation:

The regression model with quadratic trend and seasonality is used to remove the trend and seasonality of the time series and the regression residuals are forecasted using the Trailing MA 12 and further the two models are combined to create a two-level forecast model.

The quadratic regression model is used to forecast 60 periods (5 years) into the future and the regression residuals is forecasted for 60 periods in the future. The accuracy measures for the two-level model are shown above. From the accuracy report the two-level model does significantly well to absorb all the components of the data and has a MAPE and RMSE score of 9.155 and 159.887, respectively.

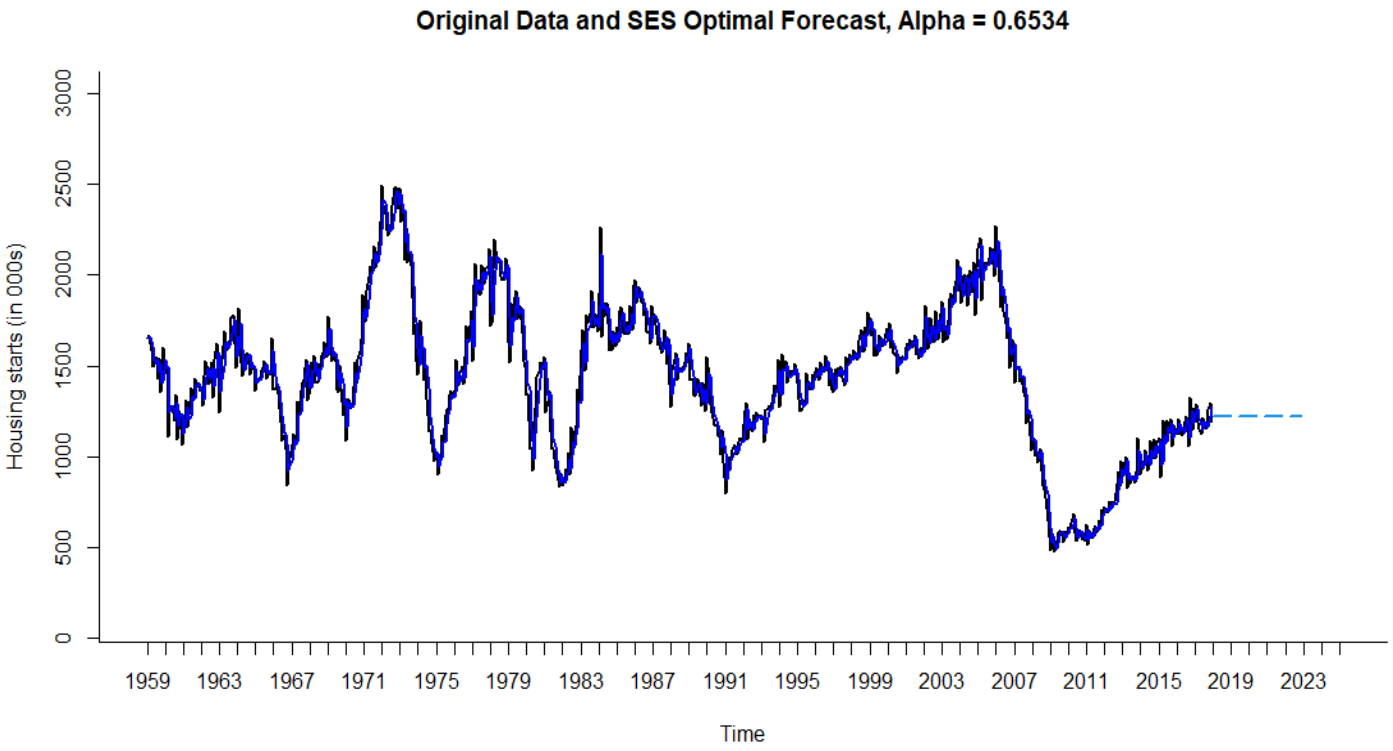
Simple exponential smoothing

Simple exponential smoothing is a popular forecasting method just like the moving average forecasting. The main difference between the two models is that in MA historical data periods are averaged to smooth out the noise and uncover data patterns. Whereas, in the simple exponential smoothing method weighted average of historical data is taken so that the weights decrease exponentially into the past.

Here, more weight is given to the recent data periods, but the older historical periods are not ignored too. This method of forecasting is typically used for series that has no trend and seasonality. So, we have built a simple exponential smoothing model with optimal value of alpha with ANN (additive error, no trend, no seasonality) and the model is used to forecast 60 periods into the future.

```
ETS(A,N,N)

Call:
 ets(y = Housing.ts, model = "ANN")

  Smoothing parameters:
    alpha = 0.6534

  Initial states:
    l = 1651.1651

  sigma:  107.5666

     AIC     AICc      BIC
11274.41 11274.45 11288.10
```

It can be seen from the model's summary that the optimal value for exponential smoothing constant (alpha) is 0.6534.



Accuracy measures:

	ME	RMSE	MAE	MPE	MAPE	ACF1	Theil's U
Test set	-0.933	107.415	80.977	-0.469	5.914	-0.014	0.943

The simple exponential smoothing model with an optimal alpha value of 0.6534 has a MAPE value of 5.914 and RMSE value of 107.415.

Advanced Exponential smoothing – Holt Winter's Model

Holt's Double exponential smoothing method is applied to time series that contains trend. The idea behind Holt's model is to augment simple exponential smoothing by capturing the trend component.

Holt's winter model is used for series which contains trend and seasonality. The idea here is to augment Holt's model by capturing both trend and seasonality components.

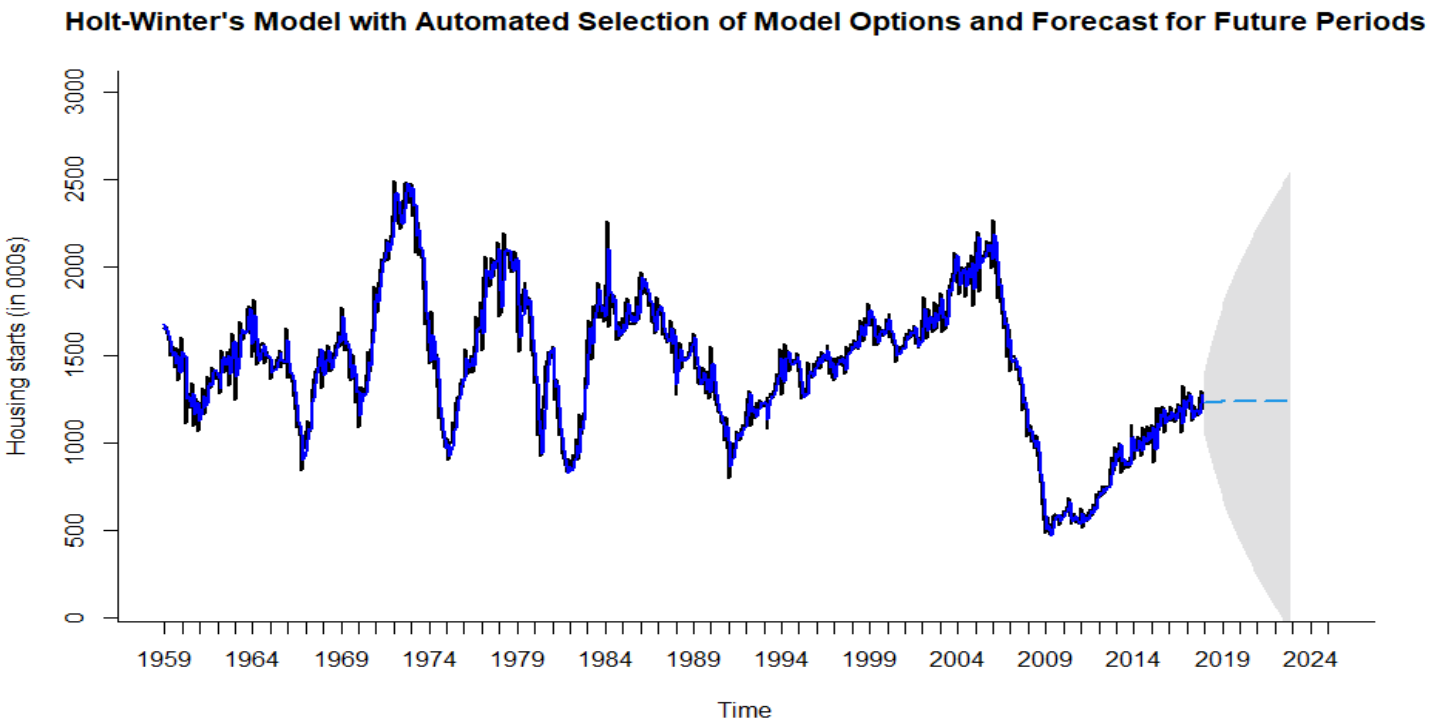
Winter's multiplicative model: Forecast = (Level +Trend) *Seasonal component

Winter's Additive model: Forecast = Level + Trend + Seasonal component

```
ETS(M,Ad,N)

Call:
 ets(y = Housing.ts, model = "ZZZ")

  Smoothing parameters:
    alpha = 0.5663
    beta  = 0.0802
    phi   = 0.8

  Initial states:
    l = 1696.775
    b = -28.1087

  sigma:  0.0761

     AIC     AICc      BIC
11236.66 11236.78 11264.04
```

A summary of the Holt-Winter's (HW) model with the automated selection of the model options and automated selection of the smoothing parameters for the training period is shown above.

This HW model has the (M, Ad, N) options, i.e., multiplicative error, Additive damped trend, and No seasonality. The optimal value for exponential smoothing constant (alpha) is 0.5663, smoothing constant for trend estimate (beta) of 0.0802, and damping coefficient (phi) is 0.8.



Accuracy measures for Holt winter's model:

```
> round(accuracy(HW.ZZZ.pred$fitted, Housing.ts), 3)
               ME    RMSE     MAE    MPE  MAPE   ACF1 Theil's U
Test set -0.558 106.456 80.076 -0.355 5.832 -0.009     0.933
>
```

Observations:

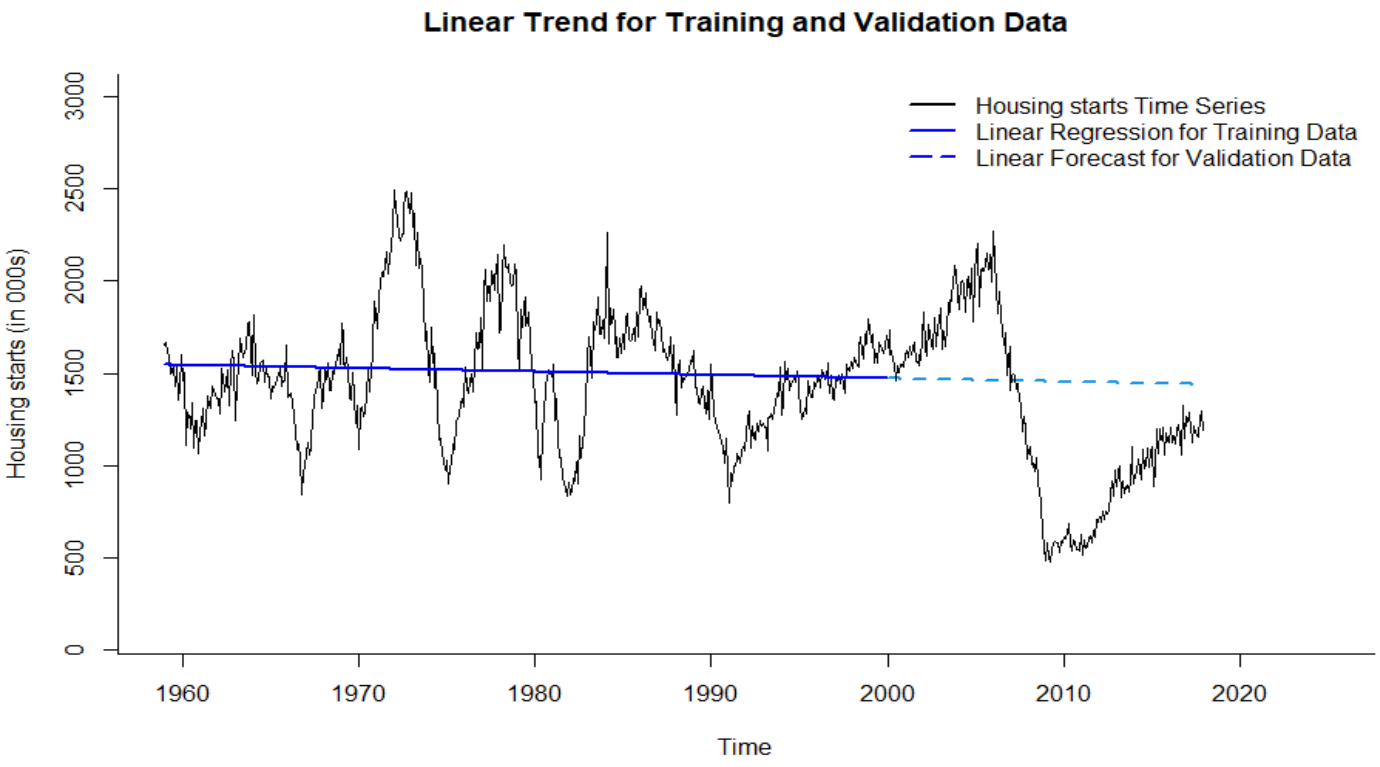
The Holt winter's model does slightly better than the simple exponential smoothing model and has an MAPE score of 5.832 and RMSE score of 106.456.

Regression models

Regression-based methods is a family of time series models based on mostly simple and multiple linear regression models as well as non-linear regression models. Regression based models can be used to fit time series with trend component, seasonal component, trend plus seasonal component and capturing special events. Here we aim to compare 6 regression-based models namely:

- Regression model with linear trend
- Regression model with exponential trend
- Regression model with quadratic trend
- Regression model with seasonality
- Regression model with linear trend and seasonality
- Regression model with quadratic trend and seasonality

1. Regression model with linear trend



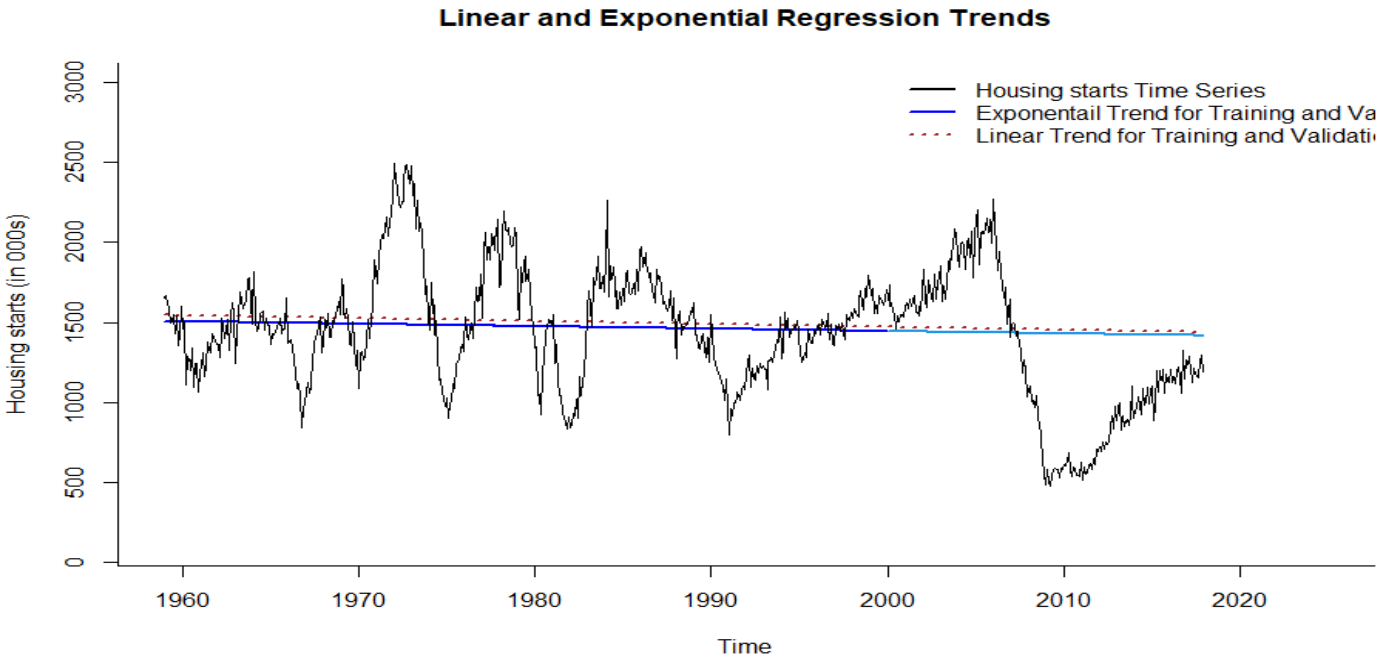
Summary of the model:

```
Call:
tslm(formula = train.ts ~ trend)

Residuals:
    Min      1Q  Median      3Q     Max
-690.63 -202.81  -18.65  177.44  971.18

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1546.3727    28.9378  53.438   <2e-16 ***
trend         -0.1500     0.1017  -1.475    0.141
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 320.4 on 490 degrees of freedom
Multiple R-squared:  0.004418,  Adjusted R-squared:  0.002386
F-statistic: 2.174 on 1 and 490 DF,  p-value: 0.141
```

The regression model with linear trend contains a single independent variable: period index (t). The model's equation is: **yt = 1546.3727 – 0.1500 t**

According to the model summary, the regression model with linear trend is statistically insignificant. It has a low R-squared of 0.004418 and adjusted R-squared of 0.002386, which represents a bad fit for the training data. The intercept and coefficient for the trend (t) variable are statistically insignificant (p-values are greater than 0.05 or 0.01). In addition, the F-statistic is also statistically insignificant (p-value is much greater than 0.05). Therefore, this model may not be used for time series forecasting in this case.

2. Regression model with exponential trend

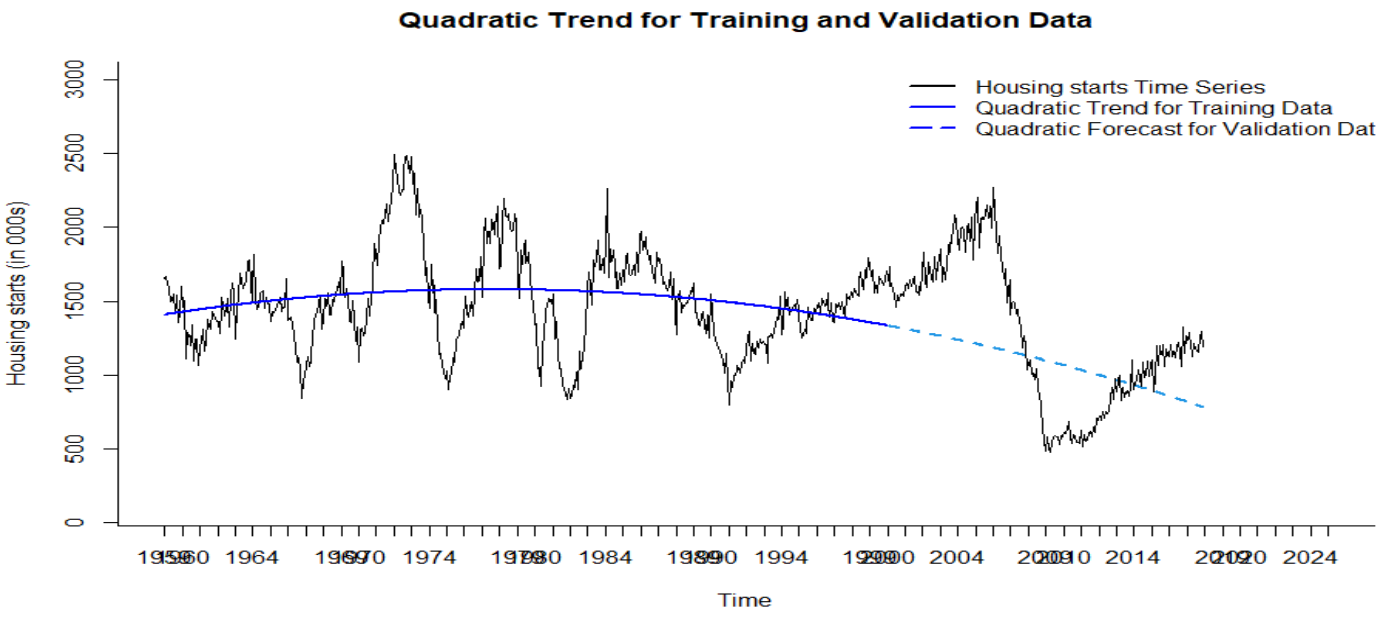


Summary of the model:

```
> summary(train.expo)

Call:
tslm(formula = train.ts ~ trend, lambda = 0)

Residuals:
     Min       1Q   Median       3Q      Max
-0.60301 -0.12260  0.01069  0.13354  0.51690

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.318e+00  1.922e-02 380.710   <2e-16 ***
trend       -8.608e-05  6.757e-05  -1.274    0.203
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2129 on 490 degrees of freedom
Multiple R-squared:  0.003301,  Adjusted R-squared:  0.001267
F-statistic: 1.623 on 1 and 490 DF,  p-value: 0.2033
```

The equation of the exponential model is given by **log(yt)=7.318 – 0.00008608**. The regression model with exponential trend a very low R-squared value and adjusted R-squared value with shows the model has a bad fitting for the data. This means that the regression model with exponential trend should be avoided in our case.

3. Regression model with quadratic trend



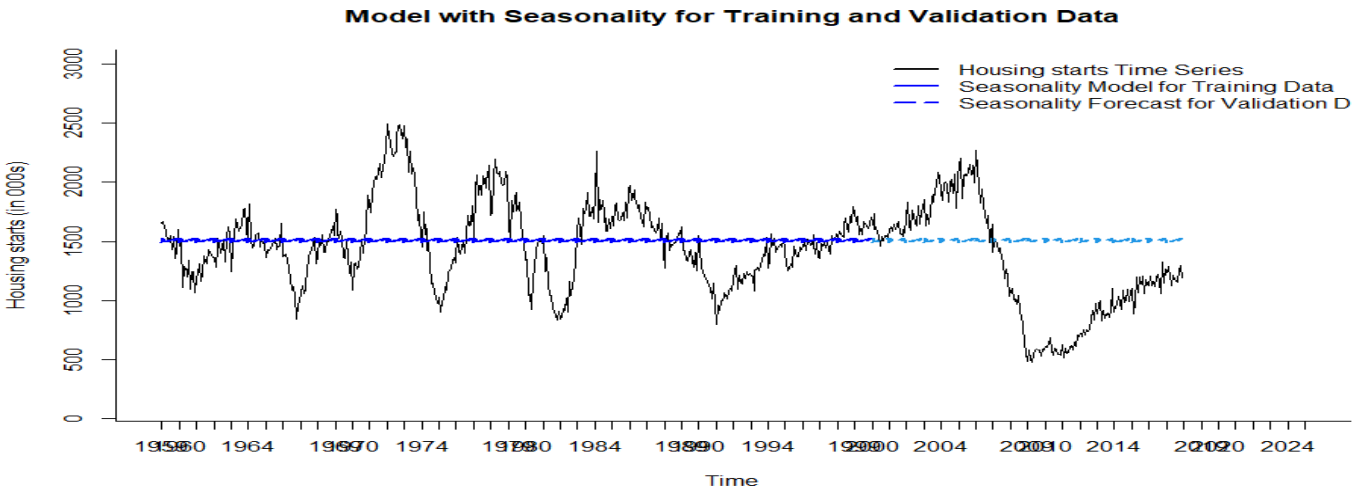
Summary of the model:

```
> summary(train.quad)

Call:
tslm(formula = train.ts ~ trend + I(trend^2))

Residuals:
    Min      1Q  Median      3Q     Max
-734.09 -194.18   -6.92  169.56  929.74

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.408e+03  4.275e+01  32.938  < 2e-16 ***
trend        1.530e+00  4.005e-01   3.820 0.000151 ***
I(trend^2)  -3.407e-03  7.866e-04  -4.331  1.8e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 314.8 on 489 degrees of freedom
Multiple R-squared:  0.0412,    Adjusted R-squared:  0.03728
F-statistic: 10.51 on 2 and 489 DF,  p-value: 3.404e-05
```

The regression model with quadratic trend contains two independent variables: period index (t) and squared period index (t2). The model's equation is: **yt = 1408 + 1.530 t – 0.003407 t2**

According to the model summary, the regression model with quadratic trend is statistically insignificant. It has a low R-squared of only 0.0412 (adj. R-squared is 0.03728), which is a bad fit for the training data. The coefficients for the trend (t) and quadratic trend (t2) variables are statistically insignificant. In addition, the F-statistic is also statistically insignificant. This model may not be used for time series forecasting in our case.

4. Regression model with seasonality



Summary of the model:

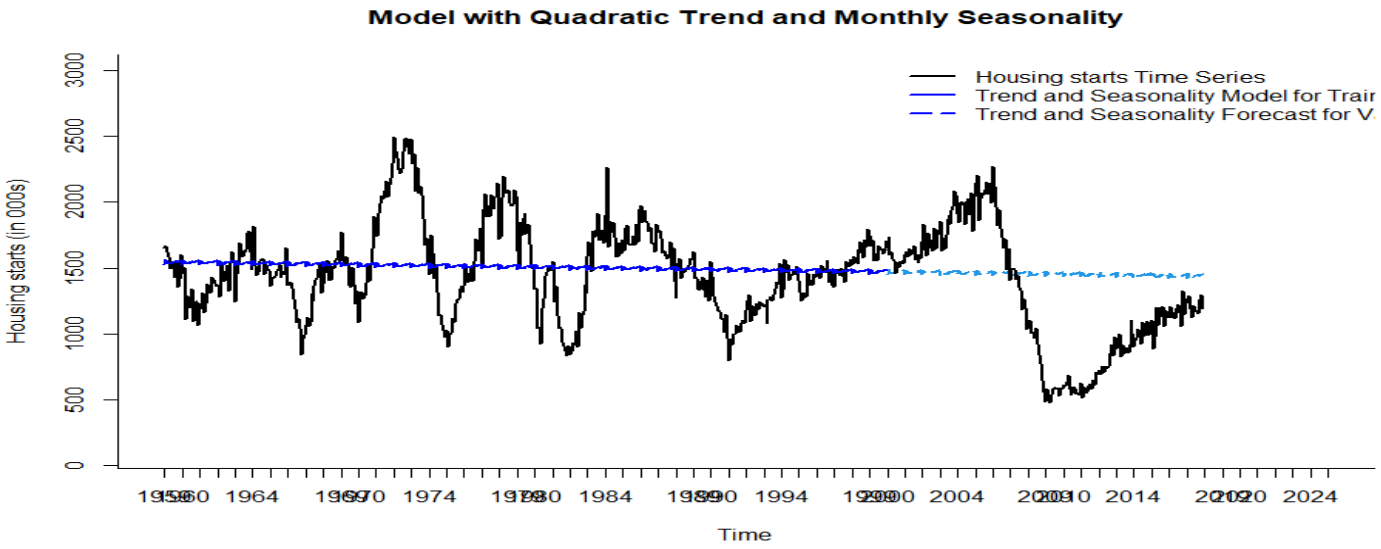
```
Call:
tslm(formula = train.ts ~ season)

Residuals:
    Min      1Q  Median      3Q     Max
-693.22 -202.77  -21.67  166.00 1002.78

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1491.220     50.650  29.442   <2e-16 ***
season2       34.585     71.629   0.483    0.629
season3       14.732     71.629   0.206    0.837
season4        7.854     71.629   0.110    0.913
season5       10.024     71.629   0.140    0.889
season6        7.317     71.629   0.102    0.919
season7       23.976     71.629   0.335    0.738
season8       21.488     71.629   0.300    0.764
season9       19.341     71.629   0.270    0.787
season10      16.683     71.629   0.233    0.816
season11      32.049     71.629   0.447    0.655
season12      30.122     71.629   0.421    0.674
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 324.3 on 480 degrees of freedom
Multiple R-squared:  0.001043,  Adjusted R-squared:  -0.02185
F-statistic: 0.04556 on 11 and 480 DF,  p-value: 1
```

The regression model with seasonality contains 11 independent seasonal dummy variables for M2 (season2 – D2), M3 (season3 – D3) and M4 (season4 – D4) and so on.

The model's equation is: $y_t = 1491.220 + 34.585\ D2 + 14.732\ D3 + \ldots + 30.122\ D12$

The model's summary shows a very low R-squared of 0.001043 and even lower adjusted R-squared -0.2185, statistically insignificant F-statistic (p-value is higher than 0.05). Overall, this regression model is not a good fit and not statistically significant, and thus cannot be applied for time series forecasting in our case.

5. Regression model with linear trend and seasonality



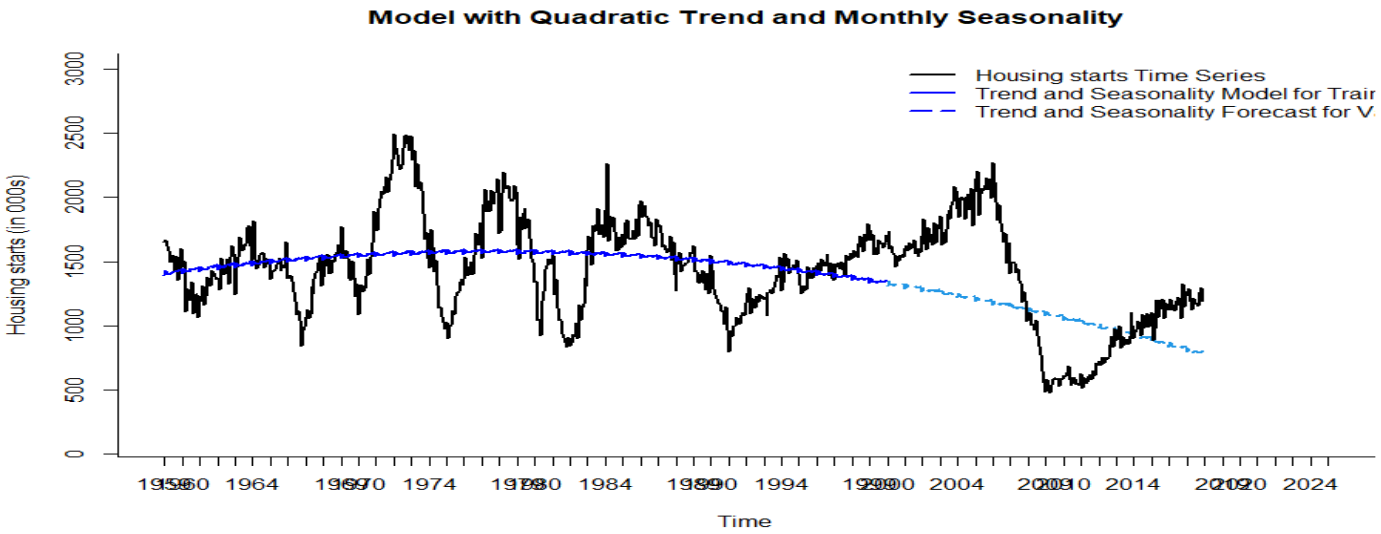
Summary of the model:

```
Call:
tslm(formula = train.ts ~ trend + season)

Residuals:
    Min      1Q  Median      3Q     Max
-688.45 -203.42  -18.08  174.13  990.10

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1527.6028    56.3352  27.116   <2e-16 ***
trend         -0.1510     0.1029  -1.468    0.143
season2       34.7363    71.5434   0.486    0.628
season3       15.0336    71.5436   0.210    0.834
season4        8.3066    71.5440   0.116    0.908
season5       10.6283    71.5445   0.149    0.882
season6        8.0719    71.5452   0.113    0.910
season7       24.8814    71.5460   0.348    0.728
season8       22.5446    71.5469   0.315    0.753
season9       20.5492    71.5480   0.287    0.774
season10      18.0416    71.5493   0.252    0.801
season11      33.5585    71.5507   0.469    0.639
season12      31.7826    71.5523   0.444    0.657
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 323.9 on 479 degrees of freedom
Multiple R-squared:  0.005516,  Adjusted R-squared:  -0.0194
F-statistic: 0.2214 on 12 and 479 DF,  p-value: 0.9974
```

The regression model contains 12 independent variables: trend index (t) and 11 seasonal dummy variables for M2 (season2 – D2), M3 (season3 – D3) and M4 (season4 – D4) and so on. The model's equation is: $y_t = 1527.6028 - 0.1510\,t + 34.7363\,D2 + 15.0336\,D3 + 8.3066\,D4 + \ldots. + 31.7826D12$

The model summary shows a very low R-squared of 0.005516 and adj. R-squared of -0.0194, which is a very low for the training data, and regression coefficients are statistically insignificant (p-value > 0.05). Overall, this regression model is a very bad fit and statistically insignificant, and thus cannot be applied for time series forecasting in our case.

6. Regression model with quadratic trend and seasonality



Summary of the model:

```
Call:
tslm(formula = train.ts ~ trend + I(trend^2) + season)

Residuals:
     Min       1Q   Median       3Q      Max
-748.64 -192.77   -7.86  171.05  948.60

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.389e+03  6.406e+01  21.689  < 2e-16 ***
trend        1.529e+00  4.048e-01   3.776 0.000179 ***
I(trend^2)  -3.407e-03  7.951e-04  -4.285 2.21e-05 ***
season2      3.470e+01  7.028e+01   0.494 0.621700
season3      1.497e+01  7.028e+01   0.213 0.831391
season4      8.225e+00  7.028e+01   0.117 0.906888
season5      1.053e+01  7.028e+01   0.150 0.880934
season6      7.970e+00  7.028e+01   0.113 0.909765
season7      2.478e+01  7.028e+01   0.353 0.724574
season8      2.245e+01  7.028e+01   0.319 0.749560
season9      2.047e+01  7.029e+01   0.291 0.771022
season10     1.798e+01  7.029e+01   0.256 0.798205
season11     3.352e+01  7.029e+01   0.477 0.633611
season12     3.178e+01  7.029e+01   0.452 0.651355
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 318.2 on 478 degrees of freedom
Multiple R-squared:  0.0423,    Adjusted R-squared:  0.01626
F-statistic: 1.624 on 13 and 478 DF,  p-value: 0.07497
```

The regression model with quadratic trend and seasonality contains 13 independent variables: trend index (t), squared trend index (t2), and 11 seasonal dummy variables for M2 (season2 – D2), M3 (season3 – D3) and M4 (season4 – D4). The model's equation is: yt = 1389 + 1.529 t – 0.003407 $t2$ + 34.70 D2 + 14.97 D3 + …+ 31.78 D12

The model summary shows a very low R-squared of 0.0423 and adj. R-squared of 0.01626. Overall, this regression model is a very bad fit and statistically insignificant, and thus cannot be applied for time series forecasting in our case.

Accuracy measures of the above 6 models on the entire dataset:

```
> round(accuracy(train.lin$fitted, Housing.ts), 3)
             ME    RMSE     MAE    MPE   MAPE  ACF1 Theil's U
Test set  0 319.794 243.394 -4.618 17.076 0.929     2.821
>
> round(accuracy(train.expo$fitted, Housing.ts), 3)
              ME    RMSE    MAE    MPE   MAPE  ACF1 Theil's U
Test set 33.36 321.565 242.79 -2.308 16.662 0.929     2.724
>
> round(accuracy(train.quad$fitted, Housing.ts), 3)
             ME    RMSE     MAE    MPE   MAPE  ACF1 Theil's U
Test set  0 313.831 240.824 -4.543 17.117 0.924     2.876
>
> round(accuracy(train.season$fitted, Housing.ts), 3)
             ME    RMSE     MAE    MPE   MAPE ACF1 Theil's U
Test set  0 320.336 242.122 -4.629 16.987 0.93     2.823
>
> round(accuracy(train.ltrend.season$fitted, Housing.ts), 3)
             ME    RMSE     MAE    MPE   MAPE ACF1 Theil's U
Test set  0 319.618 243.212 -4.612 17.061 0.93     2.819
>
> round(accuracy(train.trend.season$fitted, Housing.ts), 3)
             ME    RMSE     MAE    MPE MAPE  ACF1 Theil's U
Test set  0 313.651 240.628 -4.538 17.1 0.926     2.875
>
```

Based on the lowest values of MAPE and RMSE accuracy measures for the entire data set, all the 6 regression models above performed similarly in fitting the time series. It can be said that the above regression models fall short when compared to the MA models developed earlier. The average values of MAPE and RMSE of the above regression models is 17 and 315, respectively. But the Trailing moving average models with window width of 12 performed better with a MAPE and RMSE score of 9 and 160 respectively which is half of the regression models accuracy.

Autoregressive and ARIMA models

In a multiple regression model, we forecast the variable of interest using a linear combination of predictors. In an autoregression model, we forecast the variable of interest using a linear combination of past values of the variable. The term autoregression indicates that it is a regression of the variable against itself. Autoregressive models are remarkably flexible at handling a wide range of different time series patterns.

AR model of order 1 is given by: $\boldsymbol{Y_t = a + b_1 Y_{t-1} + e_t}$

AR model of order 2 is given by: $\boldsymbol{Y_t = a + b_1 Y_{t-1} + b_2 Y_{t-2} + e_t}$

So, basically there are two approaches in AR models:

1. Two level modelling with AR model

In a two-level model firstly, the forecast is done using a regression or a smoothing model. Then the forecast is examined for residuals by utilizing time plot of forecast residuals and ACF function plot and if the autocorrelation of residuals exists, further an AR model is fitted to forecast residual series.

2. Direct AR model

Here an AR model is fitted directly into original series by using an *ARIMA* model.

ARIMA models

Autoregressive Integrated Moving Average (ARIMA) is a class of popular models in time series forecasting which can present any time series components or a combination of the components. In general, the ARIMA model includes three parts:

- Autoregressive (AR) model with various orders of lag
- Moving average (MA) model for the model residuals
- Differenced AR model typically with lag-1 differencing to remove trend and/or seasonality

AR MODEL

AR model is used to identify relationships between time series lags and apply them in forecasting.

The general equation of an AR model is given by ***yt* = *b0* + *b*1*yt*-1 + *b*2*yt*-2 … + *b*p*yt*-p + *et.***

Typically, an AR model is appropriate for level (stationary) time series without trend and/or seasonality.

MA MODEL

Moving average model uses past forecast residuals (errors) of q autocorrelation lags in a regression-like model.

The equation of an MA model is given by **yt = c + et + q1 et-1 + q2 et-2 …+qq et-q**

where c = constant mean of the MA model, also referred to as drift

et = error term like those by the standard regression model (unexplained portion of the response variable)

et-1, et-2, …, et-q = errors in previous (lagged) time periods used to identify y_t

q1, q2 ,…, qq = coefficients of the variables to be estimated

- **Fitting an AR1 model on the time series to identify predictability**

```
> summary(his.ar1)
Series: Housing.ts
ARIMA(1,0,0) with non-zero mean

Coefficients:
         ar1       mean
      0.9578  1434.7563
s.e.  0.0105    97.4666

sigma^2 estimated as 12810:  log likelihood=-4352.98
AIC=8711.95   AICc=8711.98   BIC=8725.64

Training set error measures:
                    ME     RMSE      MAE        MPE     MAPE     MASE        ACF1
Training set -0.5488216 113.0207 86.06783 -0.8026972 6.323281 0.377077 -0.2918996
> |
```

The model's equation is: ***Yt = 1434.7563 + 0.9578Yt-1***

The coefficient of the *ar1* (*Yt*-1) variable, 0.9578, is close to 1. Therefore, the *Housing.ts* time series is very likely to be a random walk and could be hard to predict.

- **Seasonal ARIMA model with order (1,1,1) (1,1,1)**

```
> summary(arima.seas)
Series: Housing.ts
ARIMA(1,1,1)(1,1,1)[12]

Coefficients:
          ar1      ma1     sar1     sma1
      -0.0611  -0.2858  -0.0663  -0.9999
s.e.   0.0868   0.0797   0.0386   0.0159

sigma^2 estimated as 11621:  log likelihood=-4262.21
AIC=8534.43   AICc=8534.51   BIC=8557.15

Training set error measures:
                    ME    RMSE      MAE        MPE     MAPE      MASE         ACF1
Training set 1.418717 106.498 80.20627 -0.2628163 5.847514 0.3513966 -0.002835696
> |
```

ARIMA model, ARIMA (p, d, q) (P, D, Q) m, where:

- • p = 1, order 1 autoregressive model AR (1)
- • d = 1, first differencing
- • q = 1, order 1 moving average MA (1) for error lags
- • P = 1, order 1 autoregressive model AR (1) for the seasonal part
- • D = 1, first differencing for the seasonal part
- • Q = 1, order 1 moving average MA (1) for the seasonal error lags
- • m = 12, for monthly seasonality.

The model's equation is given by:

yt - yt-1 = - 0.0611(yt-1 -yt-2) - 0.2858et-1 - 0.0663(yt-1 -yt-13) - 0.9999ρt-1

- **Auto ARIMA model**

```
> summary(auto.arima)
Series: Housing.ts
ARIMA(2,1,3)(0,0,2)[12]

Coefficients:
         ar1      ar2      ma1     ma2      ma3     sma1     sma2
      1.2501  -0.6472  -1.6244  1.1293  -0.1931  -0.1221  -0.1696
s.e.  0.2291   0.1488   0.2381  0.2623   0.1044   0.0409   0.0403

sigma^2 estimated as 10964:  log likelihood=-4288.67
AIC=8593.34   AICc=8593.55   BIC=8629.83

Training set error measures:
                    ME     RMSE      MAE        MPE     MAPE      MASE          ACF1
Training set -1.118615 104.1165 77.99158 -0.5303578 5.700193 0.3416936 -0.0001789992
> |
```

This is the output of the auto ARIMA model, ARIMA (p, d, q) (P, D, Q) m, where:

- • p = 2, order 2 autoregressive model AR (2)
- • d = 1, first differencing
- • q = 3, order 3 moving average MA (3) for error lags
- • P = 0, no autoregressive model for the seasonal part
- • D = 0, no differencing for the seasonal part
- • Q = 2, order 2 moving average MA (2) for the seasonal error lags
- • m = 12, for monthly seasonality.

The model's equation is given by:

yt - yt-1 = 1.2501(yt-1 -yt-2) – 0.6472 (yt-2 -yt-3) – 1.6244et-1 + 1.1293et-2 – 0.1931et-3 – 0.1221ρt-1 - 0.1696ρt-2

Accuracy comparison of the two ARIMA models:

```
> round(accuracy(arima.seas.pred$fitted, Housing.ts), 3)
               ME    RMSE    MAE    MPE  MAPE   ACF1 Theil's U
Test set 1.419 106.498 80.206 -0.263 5.848 -0.003     0.939
> round(accuracy(auto.arima.pred$fitted, Housing.ts), 3)
              ME    RMSE    MAE   MPE MAPE ACF1 Theil's U
Test set -1.119 104.116 77.992 -0.53  5.7    0     0.912
```

From the accuracy report of the two models, it is seen that the Auto ARIMA model has the lowest MAPE (5.7) and RMSE (104.116) scores and can be used for forecasting in our case.

Model performance

MODEL	MAPE	RMSE
Naïve model	6.294	114.261
Seasonal Naïve model	17.613	302.829
Moving Average **(Trailing k=2)**	3.147	57.13
Moving Average **(Trailing k=12)**	9.097	160.059
Two level Forecasting **Regression + Trailing MA for residuals**	9.155	159.887
Simple Exponential smoothing	5.914	107.415
Advanced Exponential smoothing **Holt's winter model**	5.832	106.456
Regression model **Quadratic trend + seasonality model**	17.1	313.651
Seasonal ARIMA model **(1,1,1) (1,1,1)**	5.848	106.498
Auto ARIMA model	5.7	104.116

Observations:

- From the above accuracy measures table for all the models we can conclude that the trailing MA model with window width of 2 performed very well in capturing most of the data and could be the best model to forecast Housing starts in US
- The trailing MA model with window width of 2 has the lowest RMSE and MAPE score of 57.13 and 3.147 respectively
- Other than the trailing MA model, the exponential smoothing models and the ARIMA models did well in fitting the data but had a RMSE score almost twice the trailing MA model

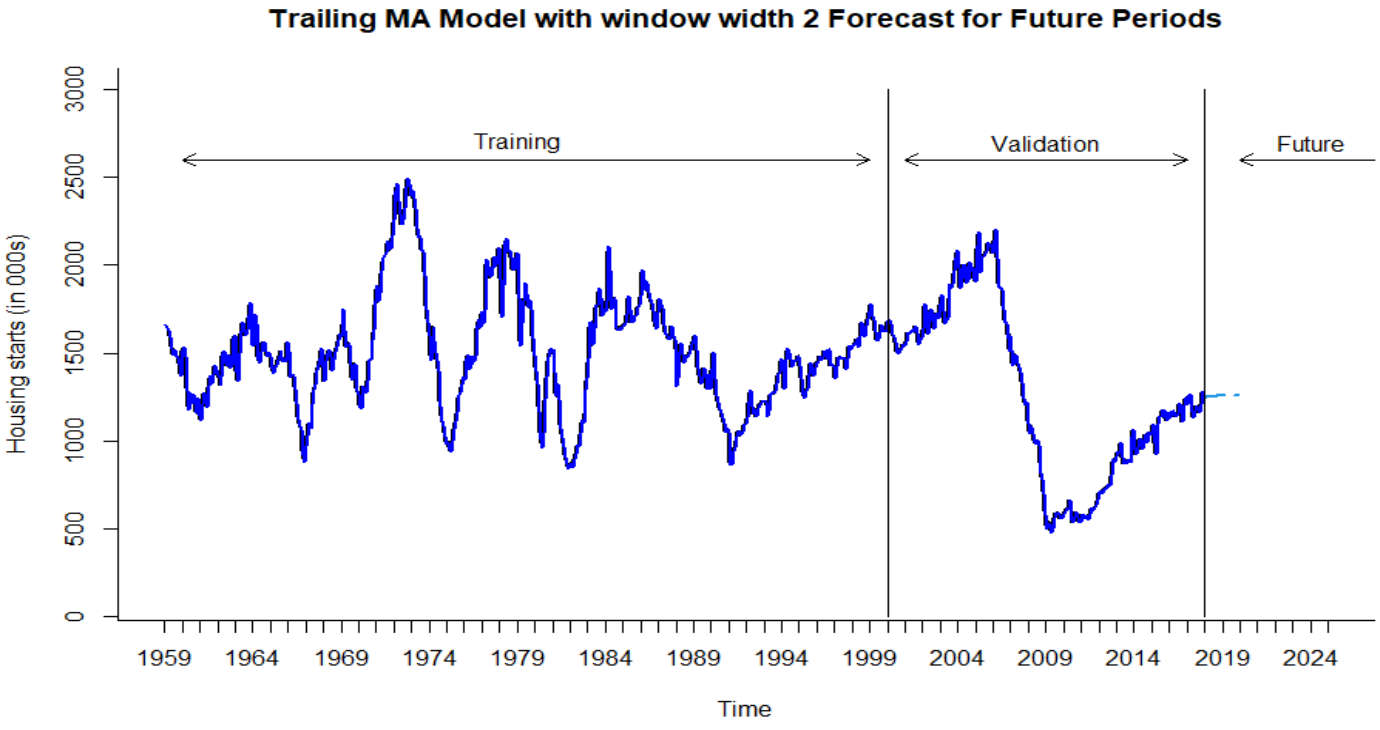
MODEL IMPLEMENTATION

As, the trailing MA model with window width of 2 did the best in the fitting the data we chose to use this model to forecast for the year 2018 and 2019.

	Point Forecast	Lo 0	Hi 0
Jan 2018	1247.154	1247.154	1247.154
Feb 2018	1249.273	1249.273	1249.273
Mar 2018	1250.969	1250.969	1250.969
Apr 2018	1252.326	1252.326	1252.326
May 2018	1253.411	1253.411	1253.411
Jun 2018	1254.279	1254.279	1254.279
Jul 2018	1254.974	1254.974	1254.974
Aug 2018	1255.530	1255.530	1255.530
Sep 2018	1255.974	1255.974	1255.974
Oct 2018	1256.330	1256.330	1256.330
Nov 2018	1256.614	1256.614	1256.614
Dec 2018	1256.842	1256.842	1256.842
Jan 2019	1257.024	1257.024	1257.024
Feb 2019	1257.170	1257.170	1257.170
Mar 2019	1257.286	1257.286	1257.286
Apr 2019	1257.380	1257.380	1257.380
May 2019	1257.454	1257.454	1257.454
Jun 2019	1257.514	1257.514	1257.514
Jul 2019	1257.562	1257.562	1257.562
Aug 2019	1257.600	1257.600	1257.600
Sep 2019	1257.630	1257.630	1257.630
Oct 2019	1257.655	1257.655	1257.655
Nov 2019	1257.674	1257.674	1257.674
Dec 2019	1257.690	1257.690	1257.690

>

Plot of Trailing MA forecast for 2018 and 2019



Conclusion

In this US Housing starts time series forecasting, we conducted time series analysis and forecasted housing starts using various time series forecasting models. From the results, it was observed that the simple model like the trailing MA average model outperformed the advanced models like the ARIMA models and advanced exponential smoothing models. The trailing MA model had a MAPE of 3.147 percent and RMSE of 57.13 and when compared to the ARIMA and Holt's winter model the performance was found to be almost 50% efficient. This suggests that we should explore simple forecasting models in predicting the Housing starts. However, we also must be aware of the model overfitting in the training sets. In our case the model was consistent when tried on the entire dataset as well. As we often assume that past patterns repeat in the future, we can use this model confidently. Also, we can make use of ensembles techniques like the weighted voting, simple averaging, weighted averaging etc. to get precise forecasts.

This forecasting model can be further built as an interactive application which will help the real estate industry and other stakeholders in the real estate market.

Bibliography

- Shmueli, G. and Lichtendahl Jr., K.C. Practical Time Series Forecasting with R, 2nd Edition, Axelrod Schnall Publishers, 2016.
- BAN 673-03Time series Analytics lecture materials– Dr. Zinovy Radovilsky