Sales and Customer Time Series Prediction with R

2023/11

Brian Chung , Jiacheng Sun

Overview

The purpose of this project is to implement time series analysis and models to predict 2018 total sales and distinct customers. The best model will be announced after our experiment.

(R will be our main and tool to accomplish all the tasks mentioned in the following paragraph)

Contribution: Both of the members contributed in all three sections. We gathered useful information and worked together with the code to not only do the data preparation but also get insights from the patterns. Finally, we both tried different parameter combinations to build the best model.

Data Source and Data Preparation (Part I)

About the Dataset:

The Sales dataset (website) from kaggle contains a total of 9994 transactions (rows) and 21 variables (columns). However, in our project, we will only extract two predictors which are **Total Sales** and **Distinct customers** from the dataset to train and predict using time series models.

Data Preparation:

Since we are doing time series analysis, our first goal is to define our time series data format, so it is worth noting that our data needed to be transformed into 'ts' format in R after any group by or any aggregate functions.

We would like to obtain as much historical data as possible so based on the dataset, the time range will be set between 2014 January and 2017 December. Then we group the data by its year and month so that we can also calculate the sum of the sales and the distinct customer numbers.

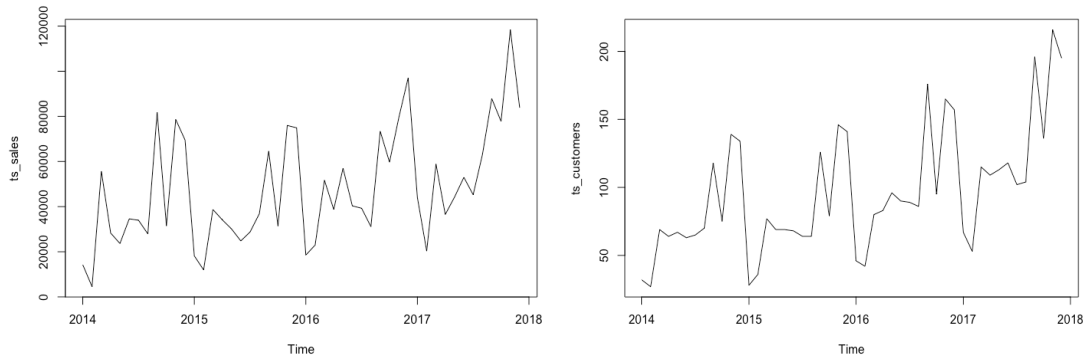
Finally, setting the ts format, the basic unit is month so that we set frequency to 12. Moreover, the start time will begin from January 2014. The results of our two time series are shown below.

```
> print(ts_sales)
           Jan       Feb       Mar       Apr       May       Jun       Jul       Aug       Sep       Oct       Nov       Dec
2014 14236.895  4519.892 55691.009 28295.345 23648.287 34595.128 33946.393 27909.468 81777.351 31453.393 78628.717 69545.620
2015 18174.076 11951.411 38726.252 34195.208 30131.686 24797.292 28765.325 36898.332 64595.918 31404.924 75972.564 74919.521
2016 18542.491 22978.815 51715.875 38750.039 56987.728 40344.534 39261.963 31115.374 73410.025 59687.745 79411.966 96999.043
2017 43971.374 20301.133 58872.353 36521.536 44261.110 52981.726 45264.416 63120.888 87866.652 77776.923 118447.825 83829.319
```

```
> print(ts_customers)
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2014  32  27  69  64  67  63  65  70 118  75 139 134
2015  28  36  77  69  69  68  64  64 126  79 146 141
2016  46  42  80  83  96  90  89  86 176  95 165 157
2017  67  53 115 109 113 118 102 104 196 136 216 195
```

Exploratory Data Analysis (Part II)

The graphs below illustrate the time series for 2014-2018 sales in dollars, and the number of customers respectively. By observing the graph, there appears a seasonality and an upward trend for both sales and customers. Next, decomposing the data could help further examine the trend, seasonality, and irregular component of a time series that can be described using an additive model.

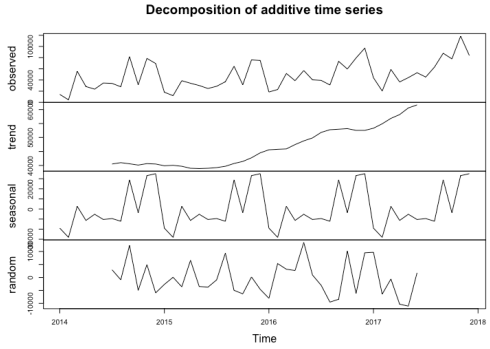
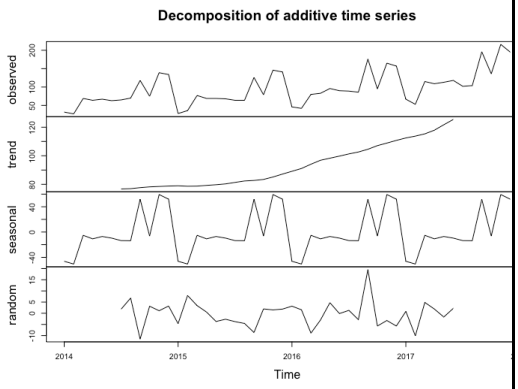


By using a decompose function() in r, the time-series data is separated into the trend, seasonal, and irregular components. By looking at seasonal component data and the graph shown below, both sales and number of customers are lowest at the beginning of the year(January and February). There are two peaks which appear in September and the end of the year. This could be explained that September is when schools start, and November and December are holiday shopping seasons. January and February are right after the holiday shopping season. By looking at the trend component, there is a steady increase in sales and # of customers after Oct 2016, and have been flat before that. The irregular component graph presents no irregular spikes for sale data indicating there might be no significant outliers. For customer data, there is a dip in Sep 2014 and a spike in Sep 2017 indicating data is more volatile in September.

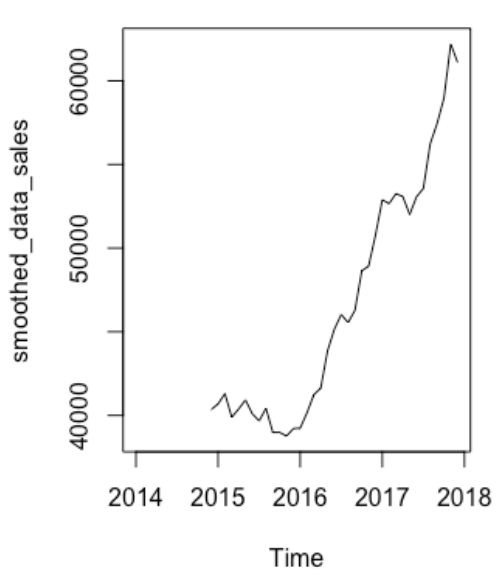
```
> decomposed_data_sales$seasonal
           Jan        Feb       Mar        Apr       May        Jun       Jul        Aug       Sep       Oct       Nov       Dec
2014 -19009.768 -28141.537  2645.882 -11364.640 -5256.485 -10426.904 -9505.787 -12154.785 28868.552 -3702.296 33052.890 34994.878
2015 -19009.768 -28141.537  2645.882 -11364.640 -5256.485 -10426.904 -9505.787 -12154.785 28868.552 -3702.296 33052.890 34994.878
2016 -19009.768 -28141.537  2645.882 -11364.640 -5256.485 -10426.904 -9505.787 -12154.785 28868.552 -3702.296 33052.890 34994.878
2017 -19009.768 -28141.537  2645.882 -11364.640 -5256.485 -10426.904 -9505.787 -12154.785 28868.552 -3702.296 33052.890 34994.878
> decomposed_data_customers$seasonal
           Jan        Feb       Mar        Apr       May        Jun       Jul        Aug       Sep       Oct       Nov       Dec
2014 -46.306713 -50.626157 -5.181713 -10.778935 -7.028935 -9.612269 -13.584491 -13.765046 51.901620 -6.362269 59.373843 51.971065
2015 -46.306713 -50.626157 -5.181713 -10.778935 -7.028935 -9.612269 -13.584491 -13.765046 51.901620 -6.362269 59.373843 51.971065
2016 -46.306713 -50.626157 -5.181713 -10.778935 -7.028935 -9.612269 -13.584491 -13.765046 51.901620 -6.362269 59.373843 51.971065
2017 -46.306713 -50.626157 -5.181713 -10.778935 -7.028935 -9.612269 -13.584491 -13.765046 51.901620 -6.362269 59.373843 51.971065
```

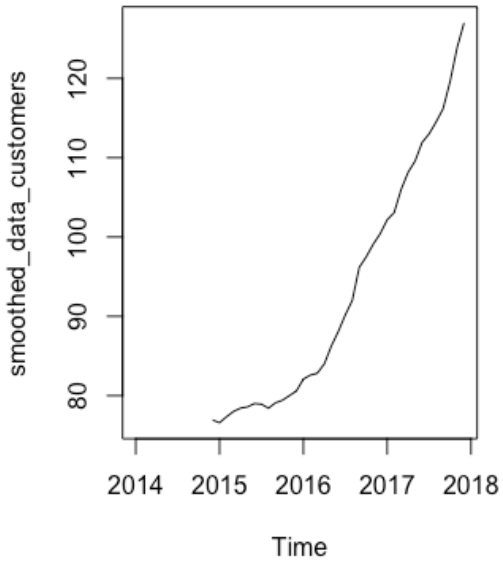
```
> decomposed_data_sales$trend
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct      Nov      Dec
2014       NA       NA       NA       NA       NA       NA 40518.01 40991.70 40594.48 40133.45 40649.42 40511.32
2015 39887.19 40045.85 39704.50 38986.58 38873.89 38987.13 39226.39 39701.22 40701.93 41432.95 42741.73 44508.54
2016 45593.70 45790.10 45916.40 47462.11 48783.86 49847.15 51826.67 52774.64 52961.25 53166.59 52543.46 52539.73
2017 53316.38 54900.05 56835.97 58192.04 60572.25 61650.01       NA       NA       NA       NA       NA       NA
> decomposed_data_customers$trend
           Jan        Feb       Mar       Apr       May       Jun      Jul       Aug       Sep       Oct       Nov       Dec
2014        NA         NA        NA        NA        NA        NA 76.75000  76.95833  77.66667  78.20833  78.50000  78.79167
2015  78.95833   78.66667  78.75000  79.25000  79.70833  80.29167 81.33333  82.33333  82.70833  83.41667  85.12500  87.16667
2016  89.12500   91.08333  94.08333  96.83333  98.29167  99.75000 101.29167 102.62500 104.54167 107.08333 108.87500 110.75000
2017 112.45833 113.75000 115.33333 117.87500 121.70833 125.41667       NA        NA        NA        NA        NA        NA
```

```
> decomposed_data_sales$random
           Jan        Feb       Mar        Apr       May        Jun       Jul        Aug       Sep       Oct       Nov       Dec
2014        NA         NA        NA         NA        NA         NA 2934.17218  -927.45002 12314.31354 -4977.75831 4926.40979 -5960.57251
2015 -2703.35088   47.09569 -3624.12527  6573.26522 -3485.71900 -3762.93411  -955.28151  9351.89800 -4974.56265 -6325.72733  177.94003 -4583.89335
2016 -8041.43967 5330.25032  3153.59343  2652.57347 13460.34832  924.28468 -3058.92041 -9504.47772 -8419.78062 10223.45591 -6184.37956  9464.43612
2017  9664.76081 -6457.37575  -609.49789 -10305.86842 -11054.65905 1758.61970        NA         NA        NA        NA        NA        NA
> decomposed_data_customers$random
           Jan        Feb       Mar       Apr       May       Jun       Jul       Aug        Sep       Oct       Nov       Dec
2014        NA         NA        NA        NA        NA        NA 1.8344907 6.8067130 -11.5682870 3.1539352 1.1261574 3.2372685
2015 -4.6516204  7.9594907 3.4317130 0.5289352 -3.6793981 -2.6793981 -3.7488426 -4.5682870  -8.6099537 1.9456019 1.5011574 1.8622685
2016  3.1817130  1.5428241 -8.9016204 -3.0543981  4.7372685 -0.1377315 1.2928241 -2.8599537  19.5567130 -5.7210648 -3.2488426 -5.7210648
2017  0.8483796 -10.1238426 4.8483796 1.9039352 -1.6793981 2.1956019        NA         NA         NA        NA        NA        NA
```

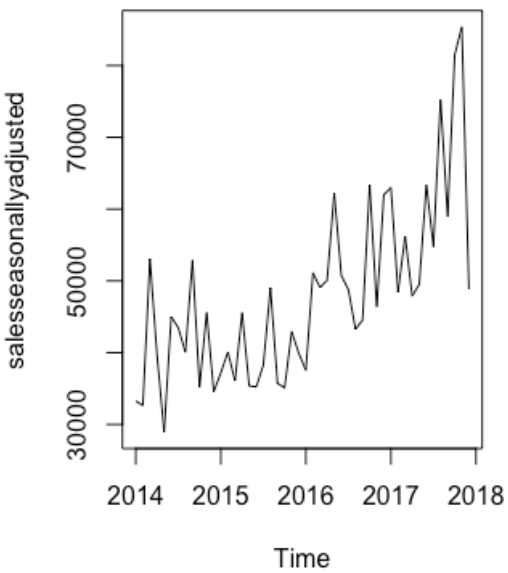
Sales	Customers
	

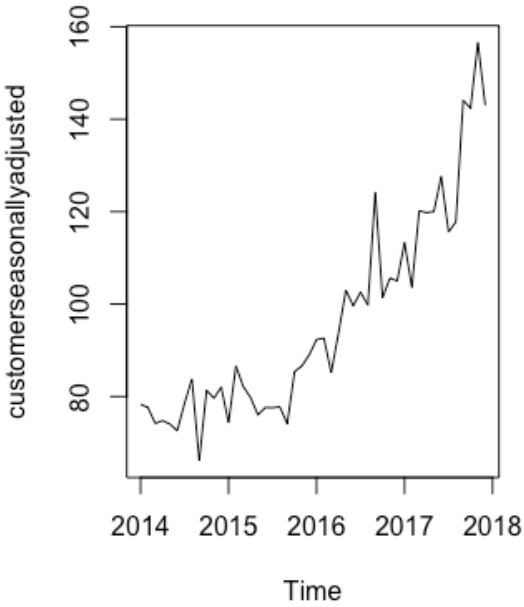
Moving averages smoothing to detrend. The graphs illustrate a one-sided moving average when n=12. Since 12 months is the length of one seasonal cycle, and one observation is one month, by setting n=12, the moving average eliminates seasonal fluctuations. Graphs present an increase in sales after 2016 and an increase in # of customers throughout.





Graph below illustrates sales and customer data after the seasonal component has been removed. The results allagin with previous findings.

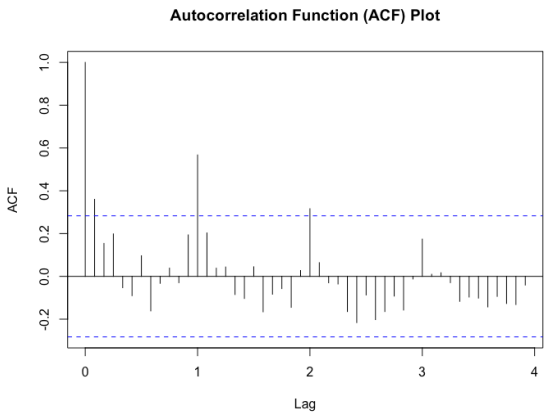
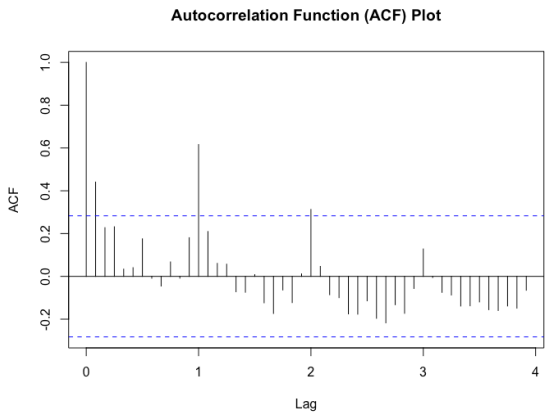




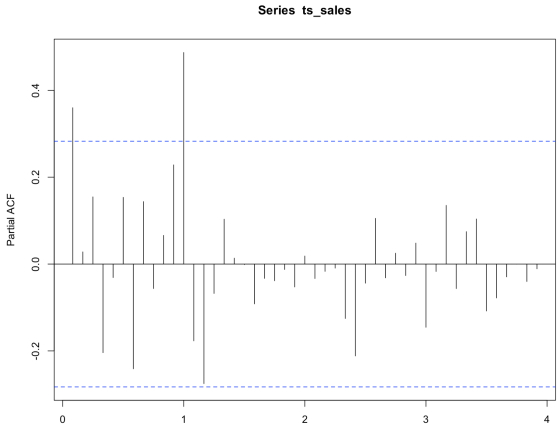
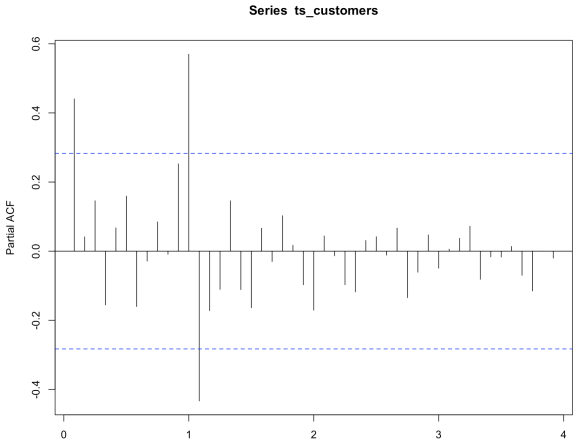
ACFs of the time series

For both Sales(left graph) and customers(right graph), observations at lag 1,12, and 24 are significantly correlated. This means the data of two adjacent months are correlated, and the same month data of one and two adjacent years are correlated. The observations that are mostly correlated are between the same month data of one adjacent year.

ACF

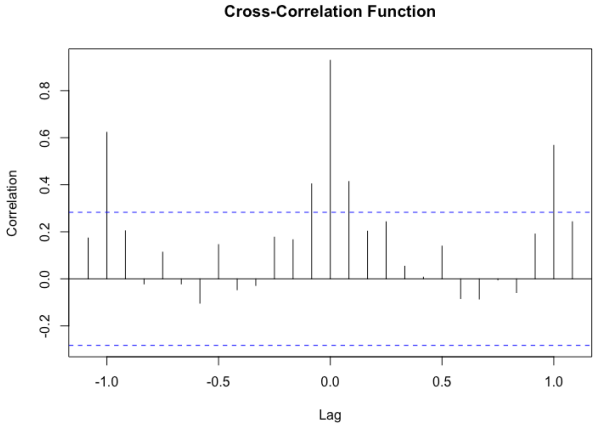
Sales	Customers
	

PACF

Sales	Customers
	

CCF of Sales vs. Customers: Sales and customer data at lag 0 have a high positive correlation with a coefficient 0.929, so sales and # of customers are correlated within the same month. The second

highest correlation is between sales and customers of the same month of a different year. Lastly, the sales and customers from adjacent months are moderately correlated.



Modeling and Forecasting (Part III)

Moving Average model(MA), Autoregressive model(AR), Autoregressive Integrated Moving Average (ARIMA), and Holt-Winters Exponential Smoothing are conducted to predict sales and # of customers in 2018. Simple Exponential Smoothing, Holt Exponential Smoothing, and ARMA are not included due to their inability to deal with trend or seasonality.

Dataset and Evaluation

Our ultimate mission is to 2018's total sales and distinct customers. However, to achieve the goal, we need to train a testing data set to evaluate our model performance before further predictions. We set all the time points before 2017 as train dataset and 2017 time points as test dataset.

Evaluation: To evaluate the time series forecasting models, Time Series Average Residuals (TSAR) which is also called Mean Absolute Error (MAE) is a common method.

$$\text{TSAR/MAE} = \left(\sum_{i=1}^{n} |Y_i - \hat{Y}_i|\right) / n$$

When evaluating the performance, the lower prediction MAE value is better, which means the prediction values are closer than the actual true values. Principle of parsimony will taking into account the ARIMA model.

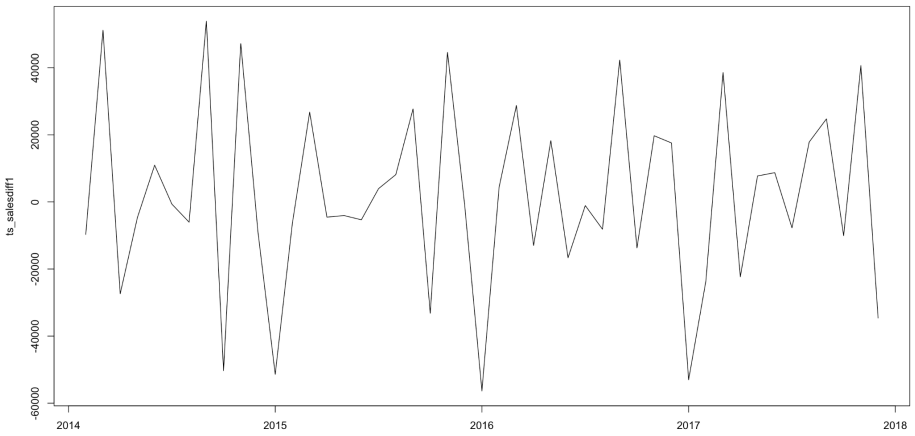
AR, MA, ARMA, Seasonal ARIMA

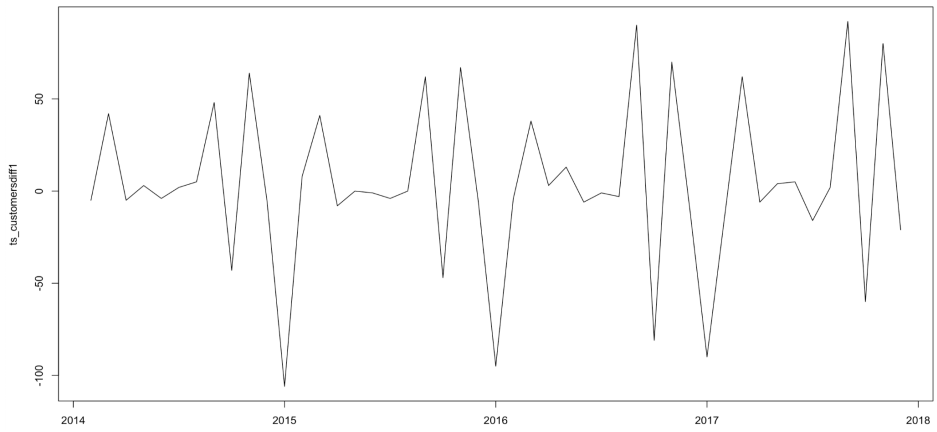
These models are under the same class. ARMA is the same as AR when not including MA(q=0), ARMA is the same as MA when AR is not included(p=0). ARMA requires stationary data, thus non-seasonal differencing and seasonal differencing is used to achieve stationary for our data. After achieving stationarity, the Autocorrelation function(ACF) and Partial Autocorrelation function(PACF) are used to set parameters for MA and AR respectively.

Differencing, ACF, PACF

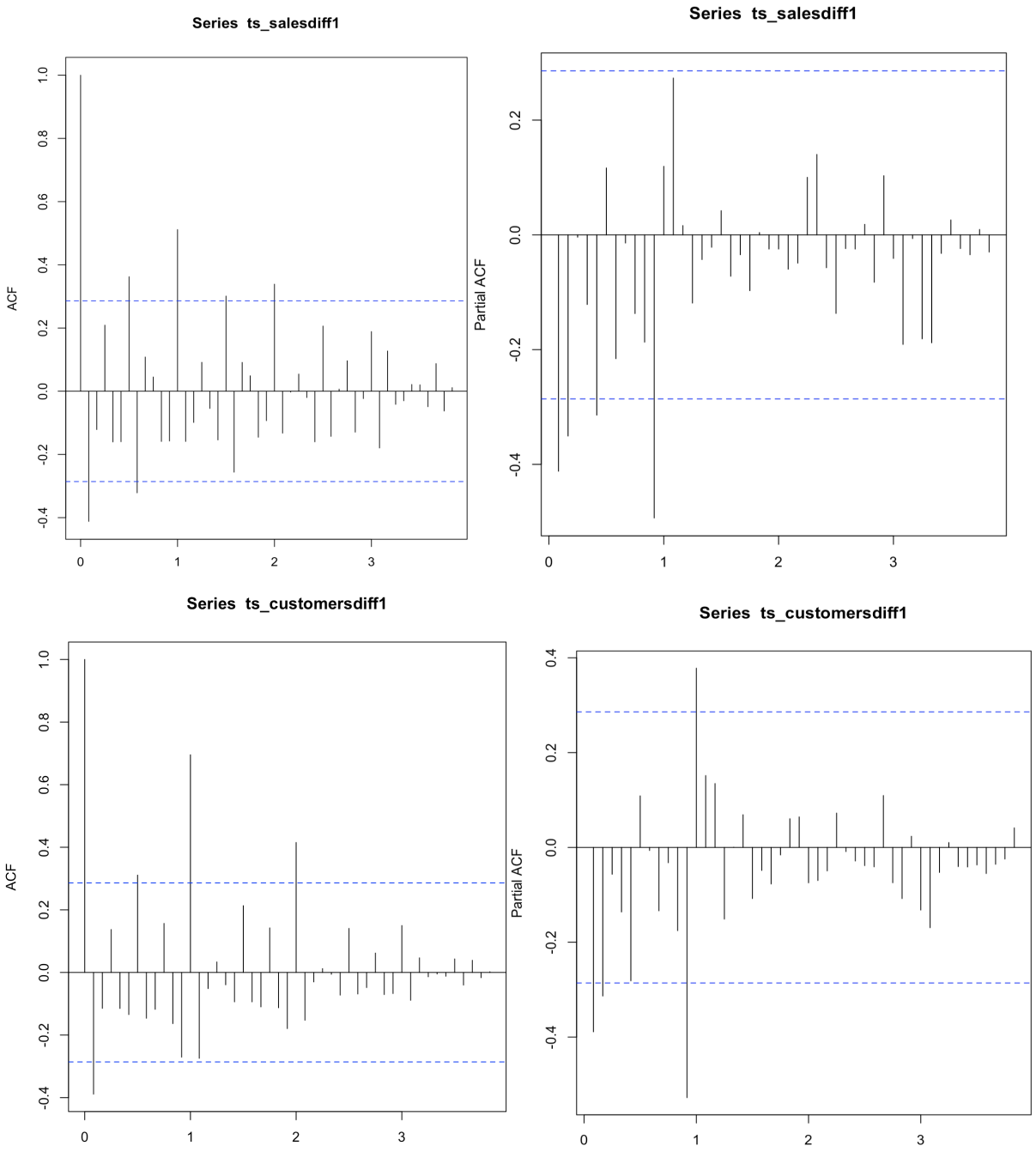
First-order differencing was applied for sales and customer time series data Time series to remove trends. Graph visualization and Dickey-Fuller test(ADF) is used to check if the trend is removed from the time series.

When difference = 1, the graph below shows no apparent trend for sale and customer data, and the p-value for ADF is 0.01 compared to 0.07 before differencing. Thus, null is rejected, and the data is stationary.



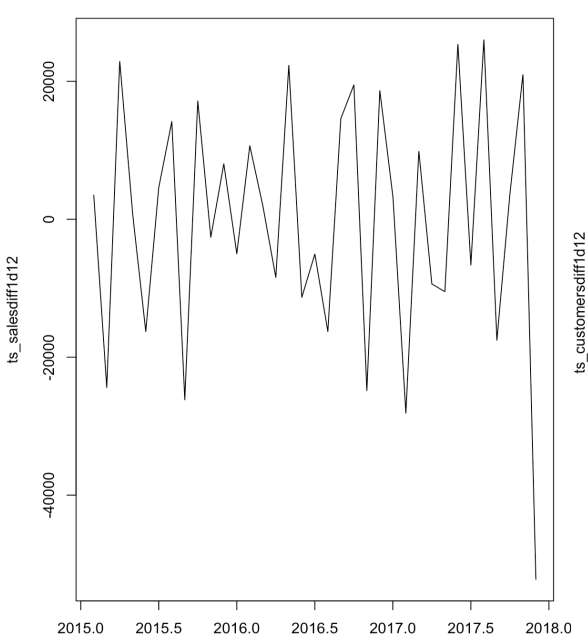


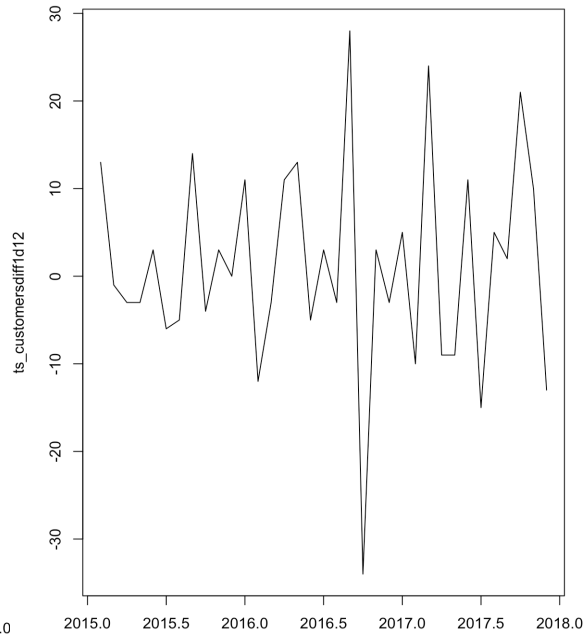
ACF after first-order differencing illustrates seasonal and half-seasonal correlation and tapers off after 2 years for both time series. PACF shows a correlation between different lags while the effect of previous observations(between the lag) is removed. For sales data, there is a negative correlation at 1,2,11,13. For customer data, the pikes are at 1,2,11,12, which also indicates seasonal correlation.



Seasonal differencing and first order differencing

Seasonal differencing is used to remove seasonality. A hegy.test may be used to use seasonality to the time series, yet of included in this project.

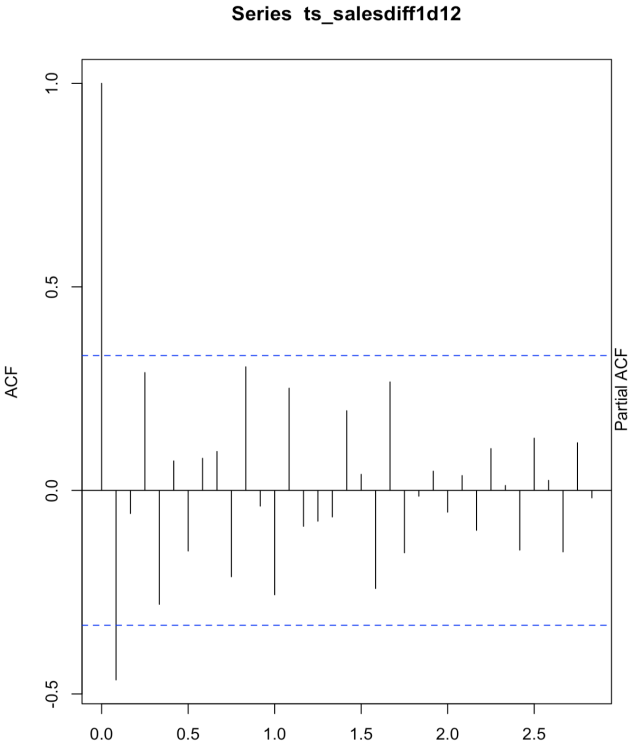


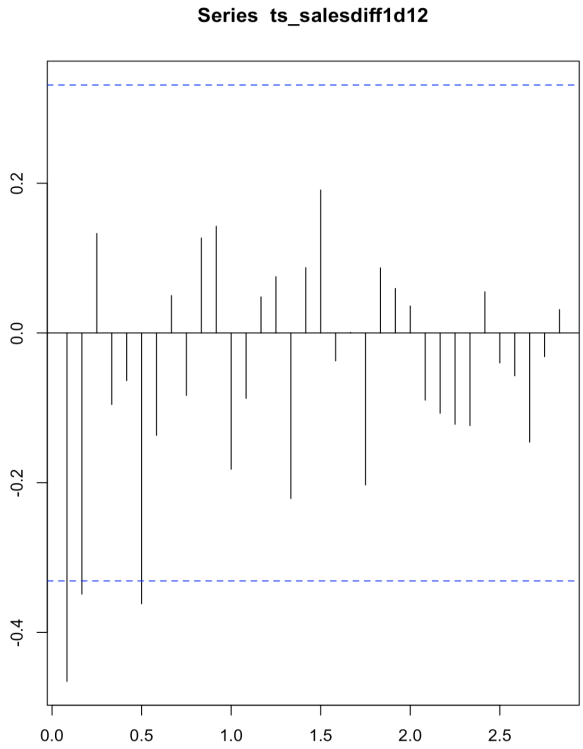


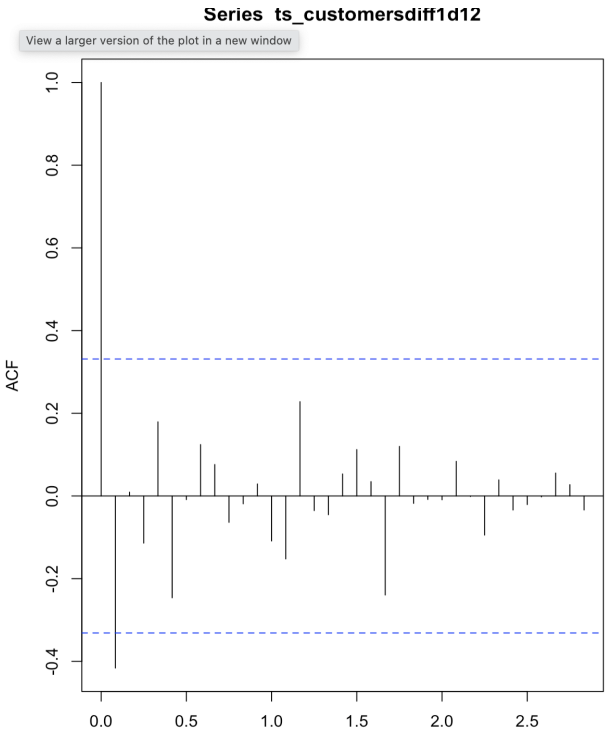
ACF and PACF after Seasonal differencing and first order differencing

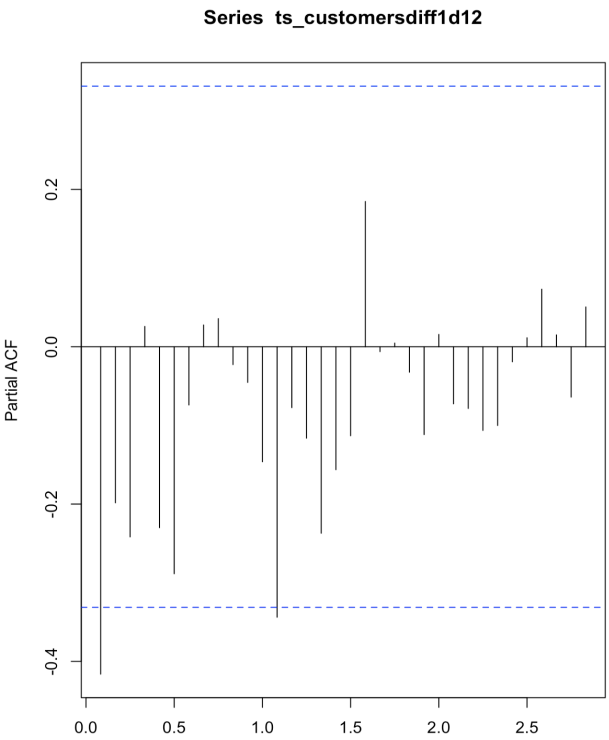
For Sales, ACF spikes at 1, and PACF at 2, a suitable parameter for ARIMA might be: ARIMA(2,1,0)(0,1,0), ARIMA(0,1,1)(010), ARIMA(1,1,1)(010).

For customers, ACF spikes at 1, and PACF at 1. Possible models are ARIMA(1,1,0)(1,1,0), ARIMA(0,1,1)(1,1,0),ARIMA (0,1,1)(0,1,0)









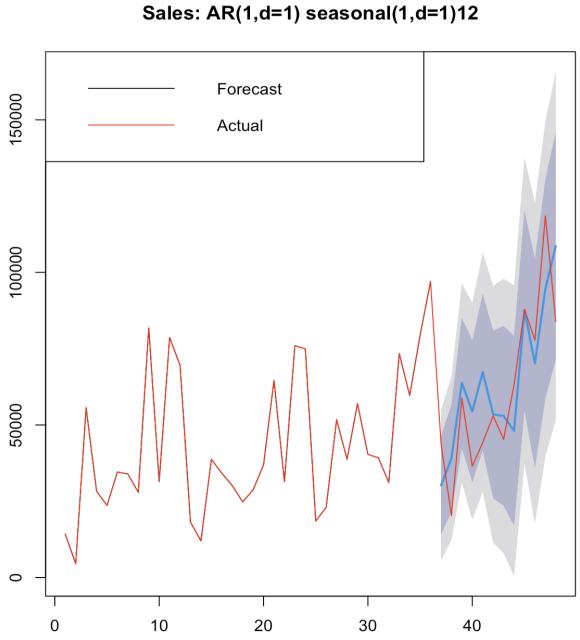
AR(seasonal)

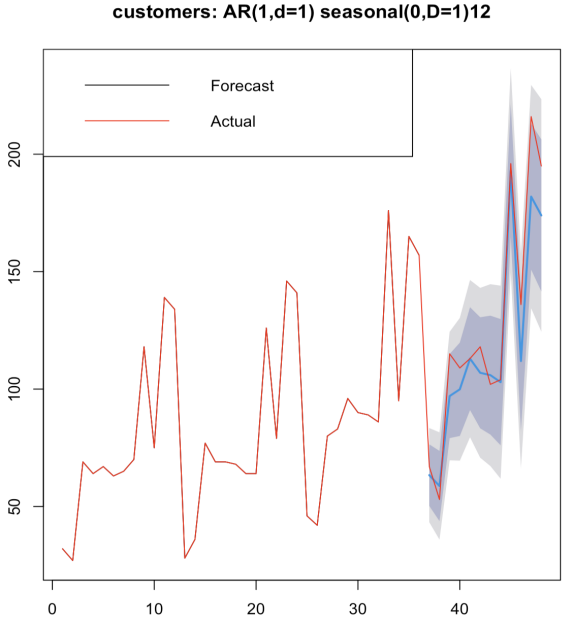
Different combinations of p,P, are calculated. For Sales, after first-order differencing and one seasonal differencing, the model has the best result and fewer parameters, when p=1, P=1 with MAE 13185.39. For customers, when p=1 and P=0, it has the best result with MAE 11.23738.

* best model selected.

Sales AR(p ,d=1)	seasonal P,, m= 12	MAE
p=1	P=2,D=0	13891.03
p = 1*	P=1,D=1	13185.39
p = 1	P=1,D=0	15068.33
p=2	P=1,D=0	14516.3
p = 2	P=1,D=1	14113.26

Customer AR(p ,d=1)	seasonal P,D=1, m= 12	MAE
p=0	P=1	11.35174
p=1*	P=0	11.23738
p = 2	P=1	11.29705
p = 3	P=0	11.22149





MA(seasonal)

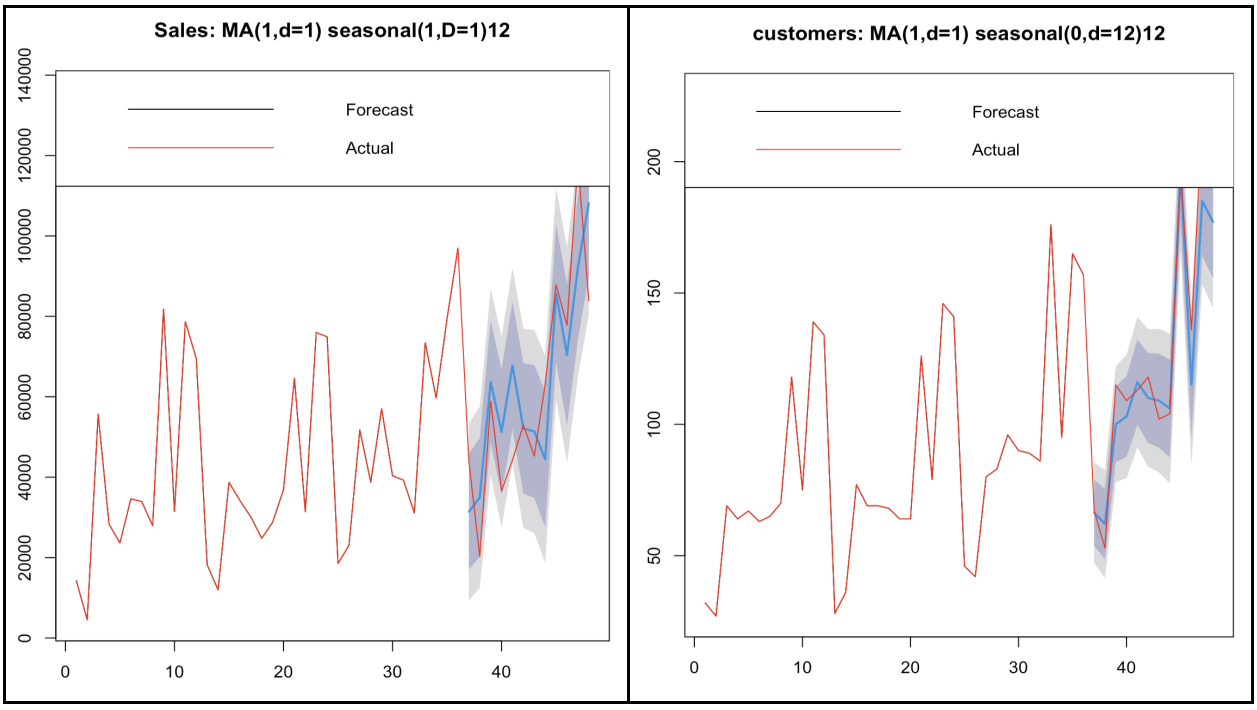
For sales data, best selected is q=1, Q=1 with first-order and one seasonal differencing.

For customer data, best is q=1, Q=0, with first-order and one seasonal differencing.

Sales: MV(q ,d=1)	seasonal(Q,D,) m= 12	MAE
q=1	Q=0,D=0	23792.34
q = 1	Q=0,D=1	13335.35
q = 1*	**Q=1,D=1**	**13022.04**
q=1	**Q=2,D=1**	**12987.33**

q=2	Q=0, D=1	13765.27
q = 2	Q=1,D=1	13337.86

Customers MV(q ,d=1)	seasonal(Q,D,) m= 12	MAE
q=0	Q=1,D=0	11.48941
q = 1*	**Q=0,D=1**	**10.03212**
q = 1	Q=1,D=1	10.18881
q=1	Q=2,D=1	10.20054
q=2	Q=0, D=1	10.05329
q = 2	Q=1,D=1	10.44923



SARIMA

SARIMA is the model integrating the characteristics of MA and AR with non-seasonal and seasonal differencing. Based on the PACF and ACF plot we discussed above, the parameters pdqPDQ will be estimated but other combinations are also calculated for comparison.

Parameter m for *((P,D,Q)m)* will be set to 12 because the seasonality pattern is cycled by year.

Sales(p,d,q)	(P, D, Q)m=12	MAE	AICc
0,1,1	0,0,0	23792.34	1084.13
1,1,1	0,1,1	13121.25	769.8803
1,1,1	0,1,0	13619.57	772.9736
0,1,2	**2,0,0**	**12875.19**	**1046.37**
0,1,2	0,1,0	13765.27	773.004
0, 1, 1*	**0, 1, 1**	**13022.04**	**767.5101**

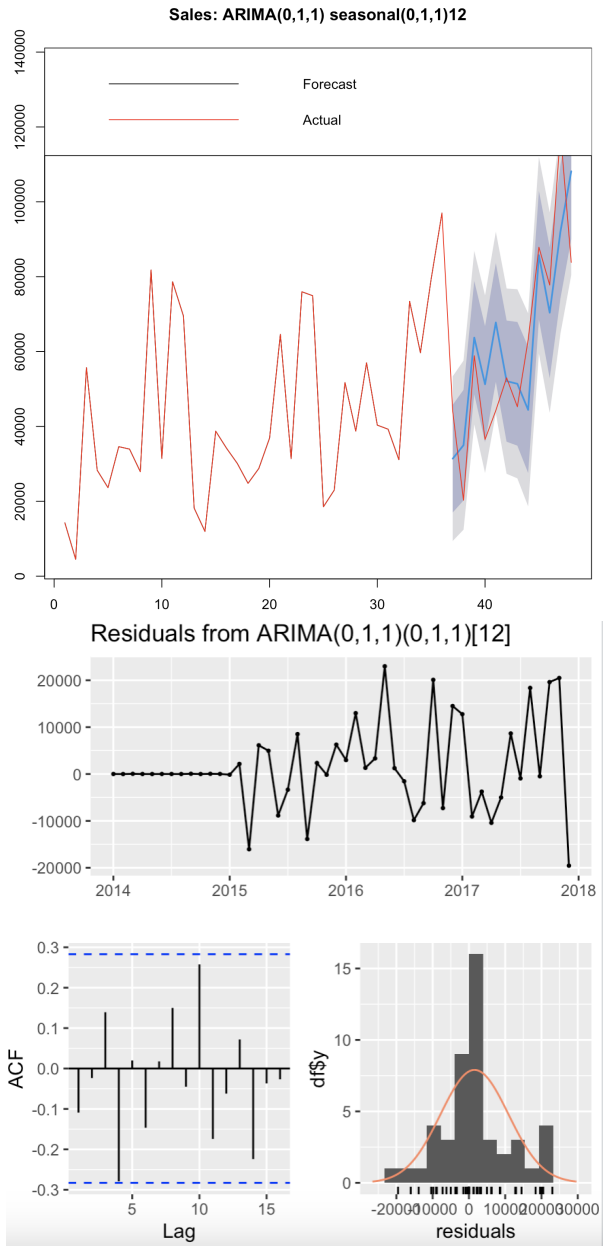
For Sales data, based on the principle of parsimony and MAE, the best model selected is (011)(011). Note the AR is not integrated, and it is same as MA with(q=1,Q=1)

Customers(p,d,q)	(P, D, Q)m=12	MAE	AICc
1,1,0	0,1,0	11.23738	270.858
2,1,0	0,1,0	11.29705	273.3983
1,1,1	0,1,0	10.0499	270.9135
1,1,1	0,1,1	10.39754	271.9579
0,1,1	0,1,1	10.18881	269.4131
0,1,2	0,1,0	10.05329	270.9431
0,1,1*	**0,1,0**	**10.03212**	**268.28**

For customer data, based on the principle of parsimony and MAE, the best model selected is (011)(010). Based on AIC and BIC, this is also the best model.

ACF, Box-Ljung test, distribution graph are calculated for model's residuals. For both data, ACF and Box-Ljung show no autocorrelation for residuals. Although the graph shows the residuals are not normally distributed, the selected models illustrated lowest MSE, and AIC.

sales

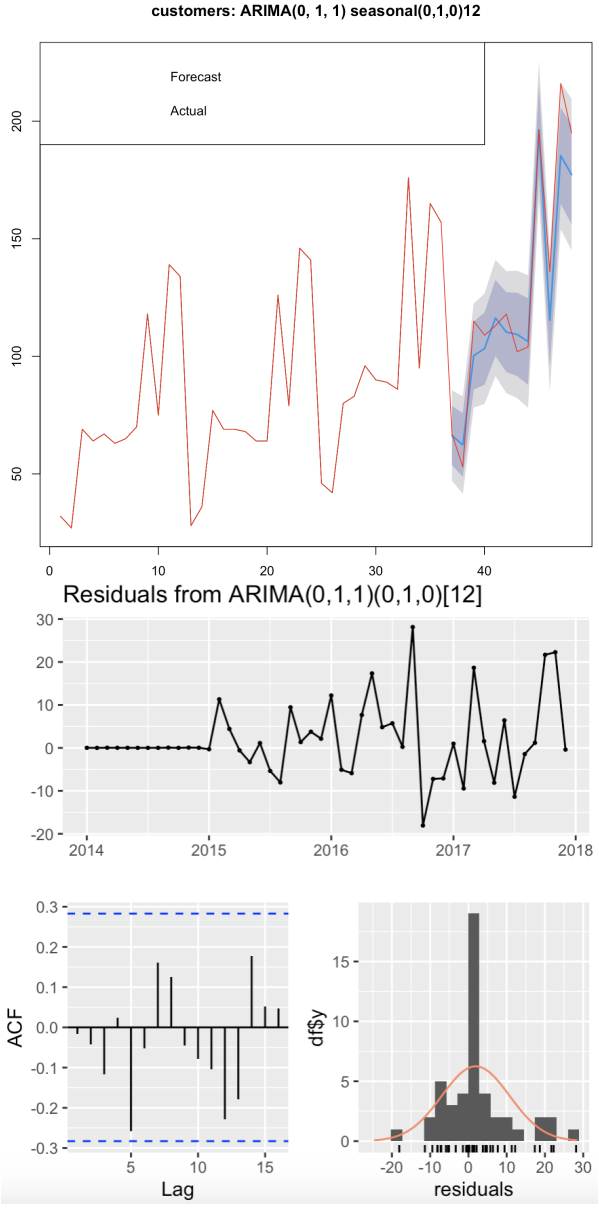



```
        Ljung-Box test

data:  Residuals from ARIMA(0,1,1)(0,1,1)[12]
Q* = 12.865, df = 8, p-value = 0.1166

Model df: 2.   Total lags used: 10
```

customer



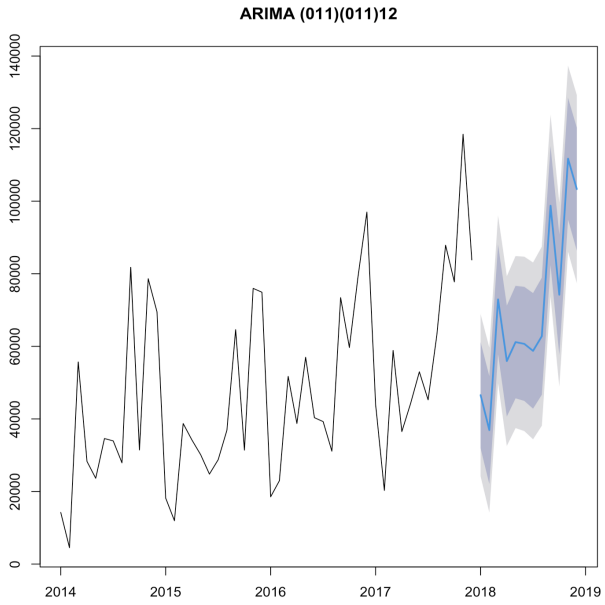

```
        Ljung-Box test

data:  Residuals from ARIMA(0,1,1)(0,1,0)[12]
Q* = 7.714, df = 9, p-value = 0.5632

Model df: 1.   Total lags used: 10
```

Table below shows the 2018 forecast for Sales and Customers.

2018 Sales forecast



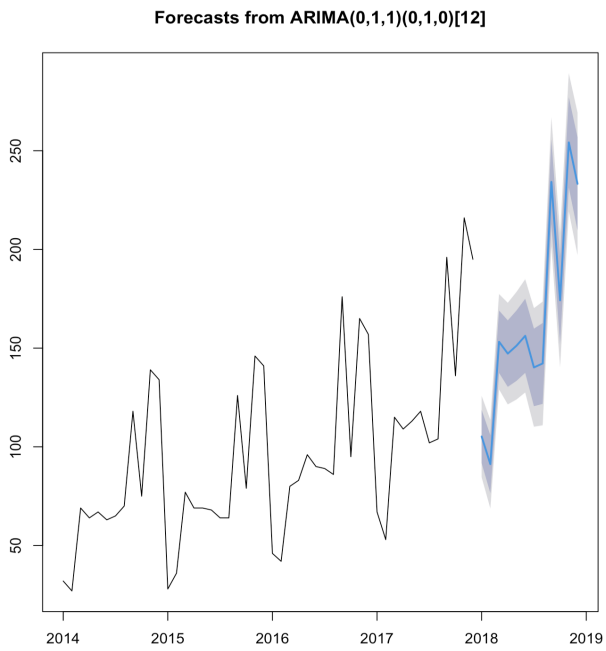
Coefficients:

ma1	sma1
-0.8212	-0.7559
s.e. 0.1049	0.6452

sigma^2 estimated as 121190436: log likelihood = -380.37, aic = 766.74

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2018	46565.13	31959.75	61170.52	24228.13	68902.14
Feb 2018	36923.66	22090.95	51756.38	14238.98	59608.34
Mar 2018	72936.38	57879.77	87992.99	49909.28	95963.48
Apr 2018	55940.29	40663.06	71217.52	32575.79	79304.80
May 2018	61178.12	45683.41	76672.82	37481.01	84875.22
Jun 2018	60687.03	44977.86	76396.20	36661.92	84712.13
Jul 2018	58754.91	42834.17	74675.66	34406.23	83103.60
Aug 2018	62791.28	46661.73	78920.83	38123.26	87459.30
Sep 2018	98749.89	82414.21	115085.57	73766.62	123733.16
Oct 2018	74224.10	57684.85	90763.35	48929.50	99518.70

2018 Customers forecast



Coefficients:

ma1
-0.5662
s.e. 0.1611

sigma^2 estimated as 110.1: log likelihood = -132.14, aic = 268.28

Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2018 105.22162	91.77162	118.6716	84.65162	125.7916
Feb 2018 91.22162	76.56049	105.8828	68.79936	113.6439
Mar 2018 153.22162	137.44205	169.0012	129.08885	177.3544
Apr 2018 147.22162	130.39780	164.0454	121.49181	172.9514
May 2018 151.22162	133.41469	169.0286	123.98826	178.4550
Jun 2018 156.22162	137.48308	174.9602	127.56349	184.8798
Jul 2018 140.22162	120.59564	159.8476	110.20627	170.2370
Aug 2018 142.22162	121.74663	162.6966	110.90782	173.5354
Sep 2018 234.22162	212.93145	255.5118	201.66112	266.7821

Nov 2018 111720.07 94979.73 128460.40 86117.93 137322.20 Dec 2018 103338.62 86399.58 120277.67 77432.60 129244.65 **Total: 843809.5**	Oct 2018 174.22162 152.14635 196.2969 140.46041 207.9828 Nov 2018 254.22162 231.38824 277.0550 219.30097 289.1423 Dec 2018 233.22162 209.65449 256.7888 197.17880 269.2644 **Total: 1982.659**

Dynamic Regression Models: Regression with ARIMA errors in R

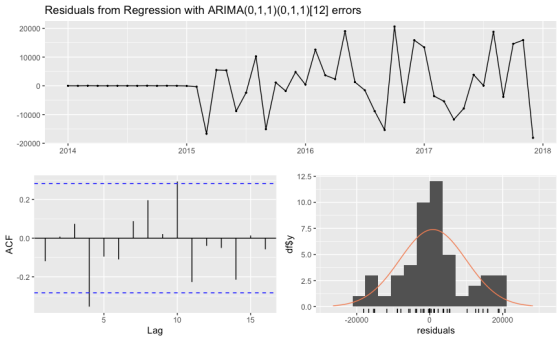
We found out that ARIMA is able to include an additional variable in errors when training the data. According to the CCF plot, we found out that when the lag is 0, Sales and Customers series are highly correlated. So we use one to support the other series prediction directly.

```
Series: ts_sales
Regression with ARIMA(0,1,1)(0,1,1)[12] errors

Coefficients:
          ma1     sma1      xreg
      -1.0000  -0.6450  308.9294
s.e.   0.1539   0.4189  145.2720

sigma^2 = 123545681:  log likelihood = -379.33
AIC=766.65   AICc=767.98   BIC=772.87
```

MAE: 22143.76

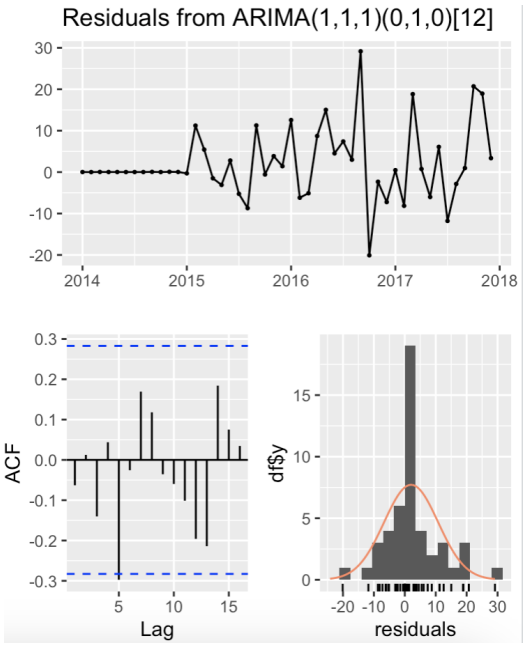



```
Series: ts_customers
Regression with ARIMA(0,1,1)(0,1,0)[12] er

Coefficients:
          ma1    xreg
      -0.5625  1e-04
s.e.   0.1723  2e-04

sigma^2 = 116:  log likelihood = -132.01
AIC=270.02   AICc=270.8   BIC=274.69
```

MAE: 23.2431



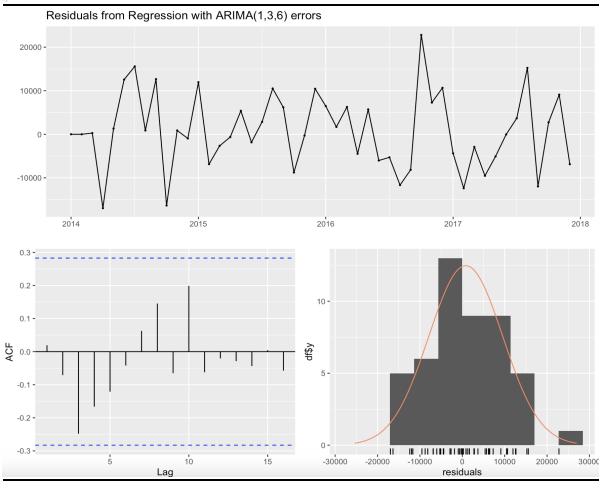
MAE is lowest with ARIMA(1,3,6). Hence we kept the best model in Dynamic Regression without seasonality parameters. We assume that the error influences more than the seasonality trend itself.

Sales: ARIMA(1,3,6) with xreg=customers: *MAE: 15536.68*	

```
Series: ts_sales
Regression with ARIMA(1,3,6) errors

Coefficients:
          ar1      ma1     ma2      ma3      ma4     ma5      ma6      xreg
      -0.1926  -2.7719  2.8873  -1.3034  -0.5117  1.3116  -0.6118  524.3072
s.e.   0.2622   0.2138  0.6618   0.9045   0.7421  0.4783   0.1680   32.3745

sigma^2 = 97509265:  log likelihood = -483.01
AIC=984.01   AICc=989.15   BIC=1000.27
```



total 2018 sale: 1827561

HOLT-WINTERS

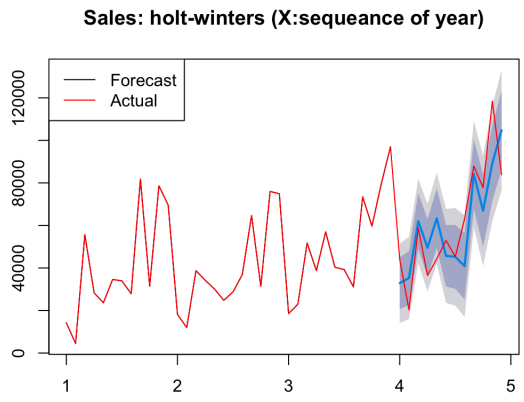
When fitting HOLT-WINTERS model, the input does not need to be stationary pre-processed. The reason is that the model considers three main components including Trend, Seasonal, and Error.

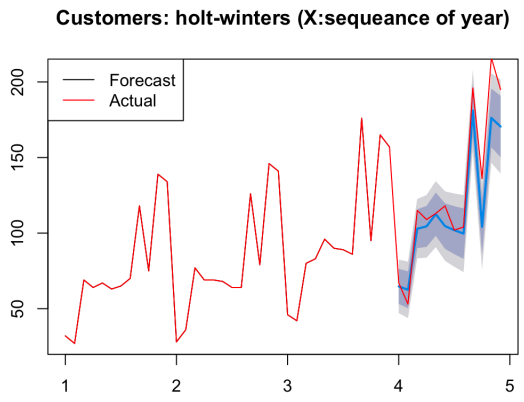
Sales	MAE on 2017 data
HoltWinters(x = ts_sales)	12972.6

Smoothing parameters: alpha(recency): 0.0932574 "alpha that are close to 0 mean that little weight is placed on the most recent observations when making forecasts of future values" beta(trend) : 0.2126965 gamma(seasonality): 0.2635965 Coefficients: a 66497.978 b 1306.376 s1 -18623.021 s2 -27743.882 s3 2081.994 s4 -8502.618 s5 -5572.860 s6 -10153.220 s7 -9012.645 s8 -8136.120 s9 32388.712 s10 1201.666 s11 39962.499 s12 30030.772	

Customers	MAE
HoltWinters(x = ts_customers) Smoothing parameters: alpha: 0.3957018 beta : 0.1014773 low trend data gamma: 1 base on very recent data Coefficients: a 148.126712 b 3.279657 s1 -41.374021 s2 -52.583290 s3 2.630199 s4 -5.533998 s5 -2.259108 s6 -3.232955 s7 -18.207328 s8 -17.101650 s9 68.852014 s10 -1.384985 s11 68.565059 s12 46.873288	13.13

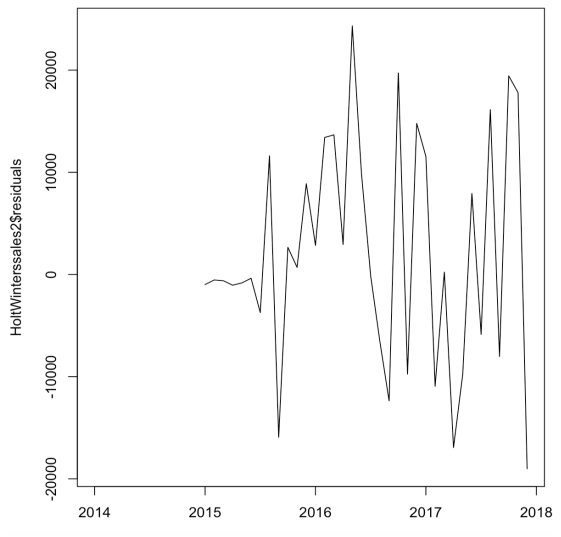
2017 actual vs predicted value

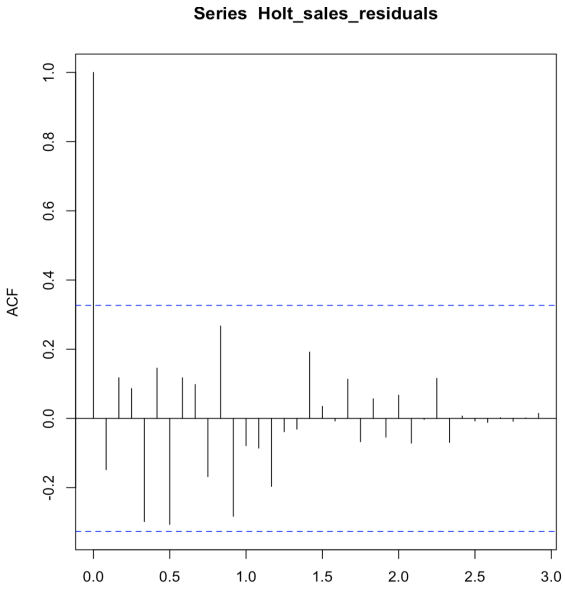




Based on the table below, for Sales and customer data, ACF and Box-Ljung test for residual shows no autocorrelation, graphs show residuals are normally distributed,hence the model is valid.

Sales residuals diagnostics



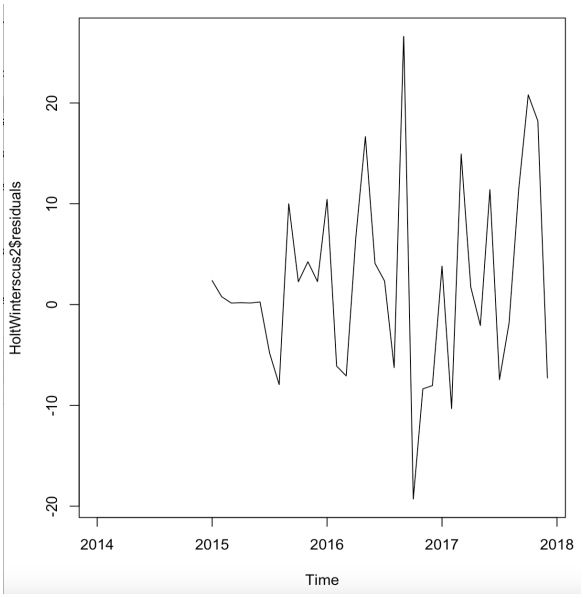


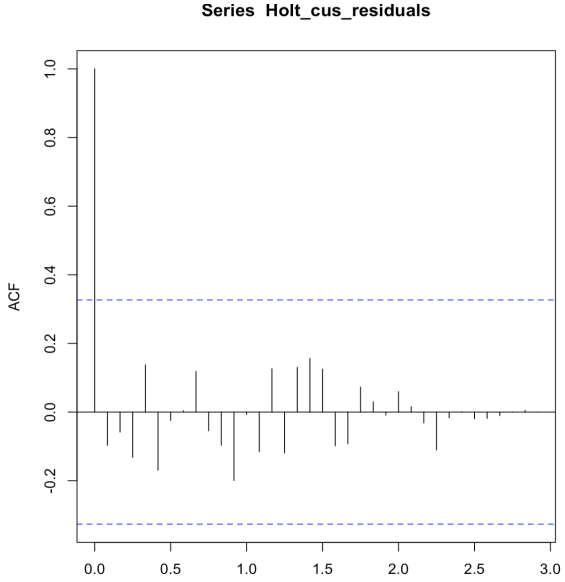
Box-Ljung test

data: HoltWinterssales2$residuals

X-squared = 28.633, df = 20, p-value = 0.09524

Customers residuals diagnostics



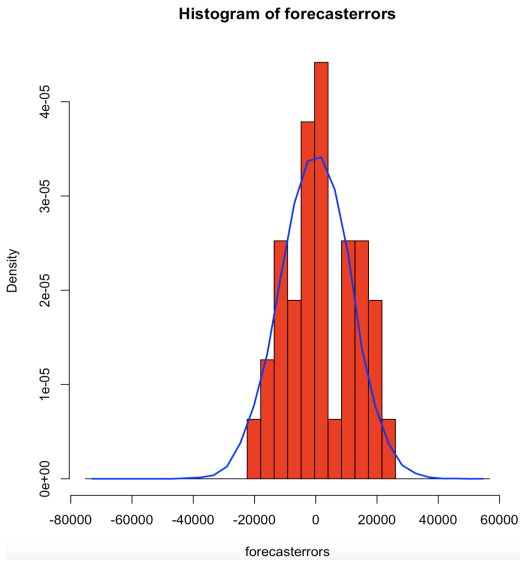


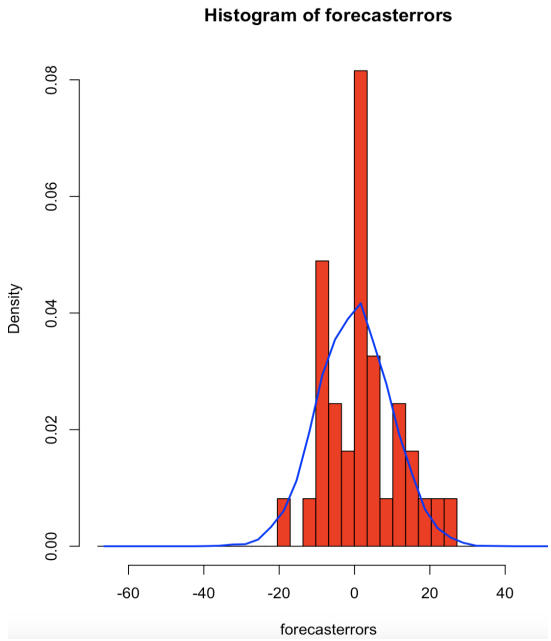
customer:

Box-Ljung test

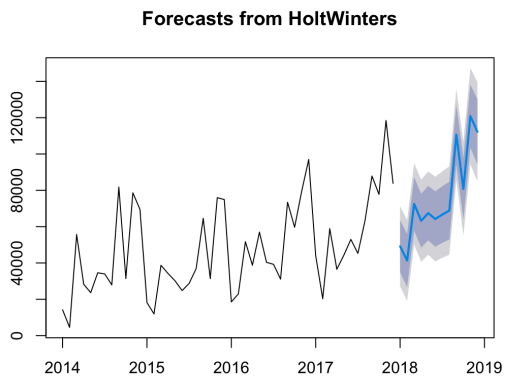
data: HoltWinterscus2$residuals

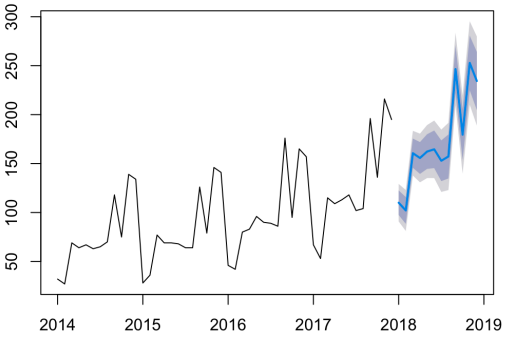
X-squared = 15.165, df = 20, p-value = 0.7669


Histogram of forecasterrors
Density
0e+00
1e-05
2e-05
3e-05
4e-05
-80000
-60000
-40000
-20000
0
20000
40000
60000
forecasterrors


Histogram of forecasterrors
Density
0.00
0.02
0.04
0.06
0.08
-60
-40
-20
0
20
40
forecasterrors

2018 Sales and customers forecast using Holt-Winters model are illustrated below.

HOLT WINTERS SALES MODEL	Plot	Values
Sales		Point Forecast Jan 2018 49181.33 Feb 2018 41366.85 Mar 2018 72499.10 Apr 2018 63220.86 May 2018 67457.00 Jun 2018 64183.02 Jul 2018 66629.97 Aug 2018 68812.87 Sep 2018 110644.08 Oct 2018 80763.41 Nov 2018 120830.62 Dec 2018 112205.27 **total:917794.4**

Customers		Point Forecast Jan 2018 110.0323 Feb 2018 102.1027 Mar 2018 160.5959 Apr 2018 155.7113 May 2018 162.2659 Jun 2018 164.5717 Jul 2018 152.8770 Aug 2018 157.2623 Sep 2018 246.4956 Oct 2018 179.5383 Nov 2018 252.7680 Dec 2018 234.3559 **total: 2078.577**

THE BEST MODEL

The best model is chosen using mainly MAE which is prediction error, however, we also consider AIC or Ljung-Box test to check if the models fit the overall dataset to avoid overfitting.

For sales data, the results from HOLT WINTERS and SARIMA are very similar but HOLT WINTERS is slightly better.. For customer data, SARIMA (0,1,1)(0,1,0) is the best.

Best Model for Sales
HoltWinters(x = ts_sales)
Smoothing parameters:
alpha(recency): 0.0932574
beta(trend) : 0.2126965
gamma(seasonality): 0.2635965

MAE:12972.6
total sales 2018 predicted: 917794.4

Best Model for Customer
SARIMA(0,1,1)(0,1,0)12

MAE:10.03212

total # customer predicted 2018: 1982.659

Summary

Summary about the optimal model for sales

In sales Holt Winters, we can found that alpha is relatively small, which means that instead of taking more lag into consideration, closer observation data points are more important. On the other hand, the beta and gamma values are relatively higher so we can conclude that trend and seasonality affect the model more. To sum up, the optimal model relies on seasonality data points more than recency data points.

"Best" forecast on Sales

Take our best Sales model into practice and predict the sales of 2018, it can be observed that the overall sales will be better each month compared to last year. Like previous years, sales will not significantly grow before august. However, sales in 2019 in November may achieve an all-time high that is estimated up to 120830.

Project Conclusion

Two monthly time-series data, Sales and Customers from 2014-2017 are cleaned and prepared to build the best predictive model for 2018. Time series analysis techniques and common models are utilized in this project. Exploratory analysis discovers data features and prepares for time-series modeling.

Important concepts such as ACF, PACF, stationarity, trend, and seasonality are considered in the model selection process. ARIMA parameters, (pdq)(PDQ)m, are manipulated to find the best result. Principles of parsimony and MAE are used to evaluate results. ACF, Box-Ljung test, and distribution of residual are used to evaluate model validity.

Limitation

One challenge we faced was to change data into stationary. It could be difficult to observe from the graph. Additional tests could be utilized. Since the data is too small, we did not conduct cross-validation, other methods may be used to test for overfitting.

Reference

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice ; a comprehensive indtroduction to the latest forecasting methods using* R *; learn to improve your forecast accuracy using dozenss of real data examples*. Otexts.