Two Test Exercises in Time Series Modeling

Nelson R. Manohar
September 3, 2015
nelsonmanohar@yahoo.com
http://nelsonmanohar.wordpress.com

PART I:
Based on the ***Sales*** data set, **predict how many pints of ice cream** the store should stock for **September, October, November, and December.**

A. Preprocessing
The Sales data was clean, quite small (***92*** samples of three variables). It did not require preprocessing.

B. Stationary
Visual inspection of the time series indicated a **yearly repeating pattern** subject to a **slight additive effect yet stationary enough** to predict on without transformation (***log*** or ***diff***) as indicated by the ***ndiffs***() command.

C. Exploratory Modeling
The particularly strong and clear seasonality pattern indicated that a ***SEASONAL.ARIMA*** model was needed. Using ***AIC***-based exploration of model choices, I arrived to an **ARIMA(1,0,0)(1,1,0[12] with drift)** model. For comparative and ensemble-building purposes, ***LM*** and ***EWMA*** models were also considered. **Illustration 1** provides a visual summary of the analysis of the ***ICE_CREAM_SALES*** time series. The ***S1*** plot shows the time series as given, with a repeating pattern of year-end ramp ups on sales of ice-cream. The ***S2*** plot shows the residuals resulting from the fitting of the above **ARIMA(1,0,0)(1,1,0[12] with drift)** model to the series (with time in months). The histogram of the residuals is shown in ***S3*** and it is relatively normal and zero-mean centered. Plot ***S4*** and ***S5*** show the ***ACF*** and ***PACF*** for the residuals, both showing the absence of significant spikes. Finally, plot ***S6*** illustrates the forecast values for September, October, November, and December obtained from this model, which are in agreement to both the expected year-end repeating pattern and the slight additive component observed. Finally, the model's p-values for the ***Ljung-Box statistic*** were high for all lags.

The final **ARIMA(1,0,0)(1,1,0[12] with drift** given below comprises one ***ar*** and one ***seasonal ar*** term:

```
Series: sales$Ice_Cream_Sales
ARIMA(1,0,0)(1,1,0)[12] with drift
Coefficients:
       ar1   sar1  drift
      0.32  -0.34   25.4
s.e.  0.12   0.12    8.3
sigma^2 estimated as 632727:  log likelihood=-546
AIC=1100   AICc=1100   BIC=1109
Training set error measures:
                ME RMSE MAE  MPE MAPE MASE    ACF1
Training set 4.4  684 470 -3.8   12 0.71 -0.0071
```

D. Forecast
Using the above model, the forecast was computed for Sep/2015 through Dec/2015 as shown below in the shaded column.

Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95	end_year_dates	number_pints
Sep Y8	5467.7	4448.3	6487.1	3908.7	7026.8	**2015-09-01**	**5467**
Oct Y8	6936.4	5867.6	8005.1	5301.8	8570.9	**2015-10-01**	**6936**
Nov Y8	10565.7	9492.1	11639.2	8923.8	12207.5	**2015-11-01**	**10565**
Dec Y8	13769.1	12695.1	14843.1	12126.5	15411.7	**2015-12-01**	**13769**

E. Software:
The ***R*** script implementing the approach described above is found on: "**verizon_sales.R".**

F. Caveat
Partly because data was scarce, all data was used and no cross-validation dataset was set aside.

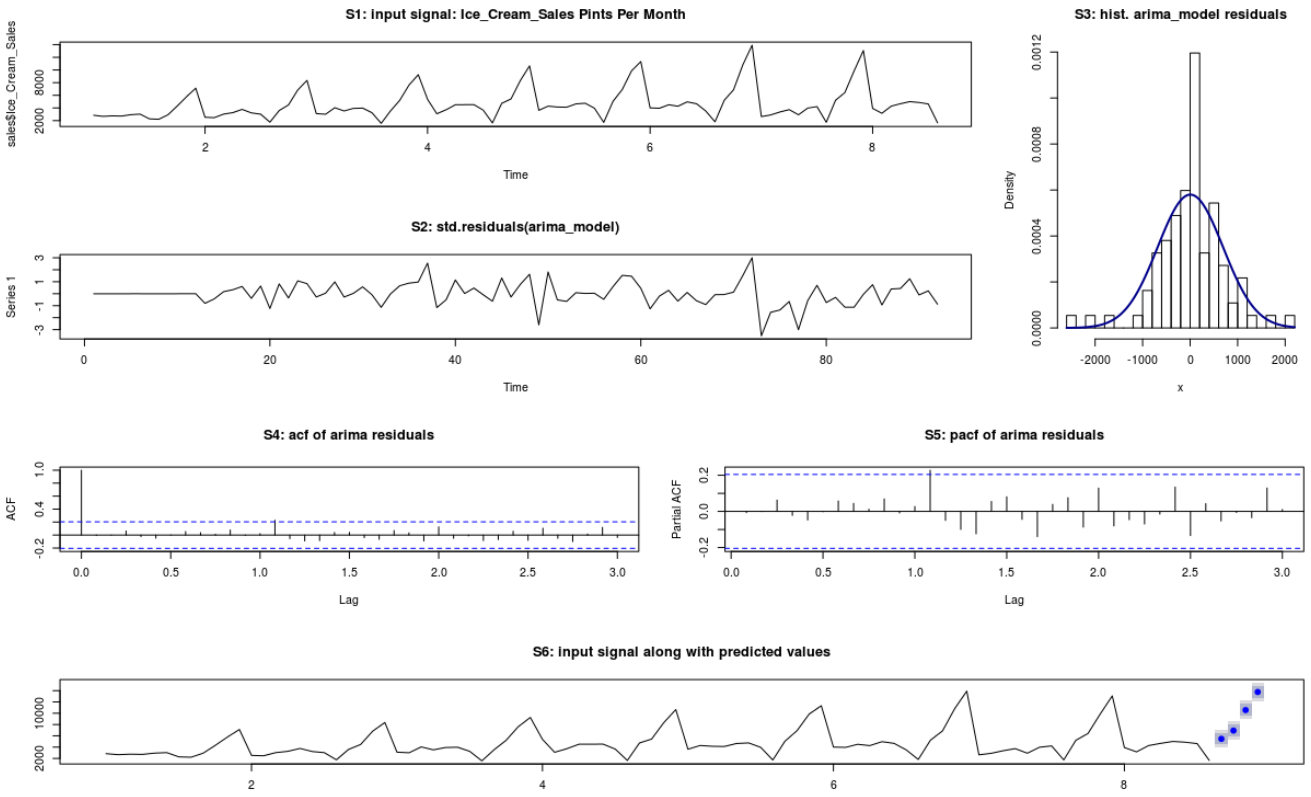

Illustration 1: ICE CREAM SALES: S1 (input), S2(residuals), S3(hist. residuals), S4 (acf), S5 (pacf), S6 (forecast).

PART II:

The ***Transactions*** data set is the number of transactions that occurred on a specific cycle date. ***Transactions*** cannot occur on the ***5th, 11th, 14th, 17th, 24th, 27th, 29th, 30th, or 31st***. The company is interested in **predicting the number of transactions that will occur every day in March of 2015.**

A. Preprocessing

The ***Transactions*** data had gaps as specified. The time series had ***606*** samples and it was augmented via SQL with several time-based factors such as ***quarter***, ***day_of_week***, ***week_of_year***, ***day_of_year***, ***month***, etc. to aid in the exploratory analysis.

B. Stationary Data Conditioning and Miscellaneous Data Issues

After examining basic models using the gapped data, I decided to **interpolate** the gaps to further the analysis. For each gap being interpolated, its value was smoothed as the mean of the preceding three samples and the subsequent three samples. The interpolated time series had ***698*** samples. However, even after the gaps were interpolated, the transaction data exhibited several fitness issues that needed to be addressed prior to modeling. First, the ***Transactions*** time series represents a growth process with a multiplicative trend, so a log was taken. Then, the ***log***(***Transaction***) time series was not stationary enough and though ***ndiffs***() recommended a total of 1 difference to be applied, both one and two differences were investigated. Then, the ***diff(log(Transactions))*** exhibited a monthly pattern, indicating the need for a ***SEASONAL.ARIMA*** model. Finally, the ***diff(log(Transactions))*** exhibited apparent ***short-range*** and ***long-range*** dependencies. Specifically, at beginning of each month, a 2-week slide downwards (AR components) was then followed by a stationary 1-week (MA components)

followed by a 1-week slide downwards (AR components) – thus pointing to the possibility of a high degree model.

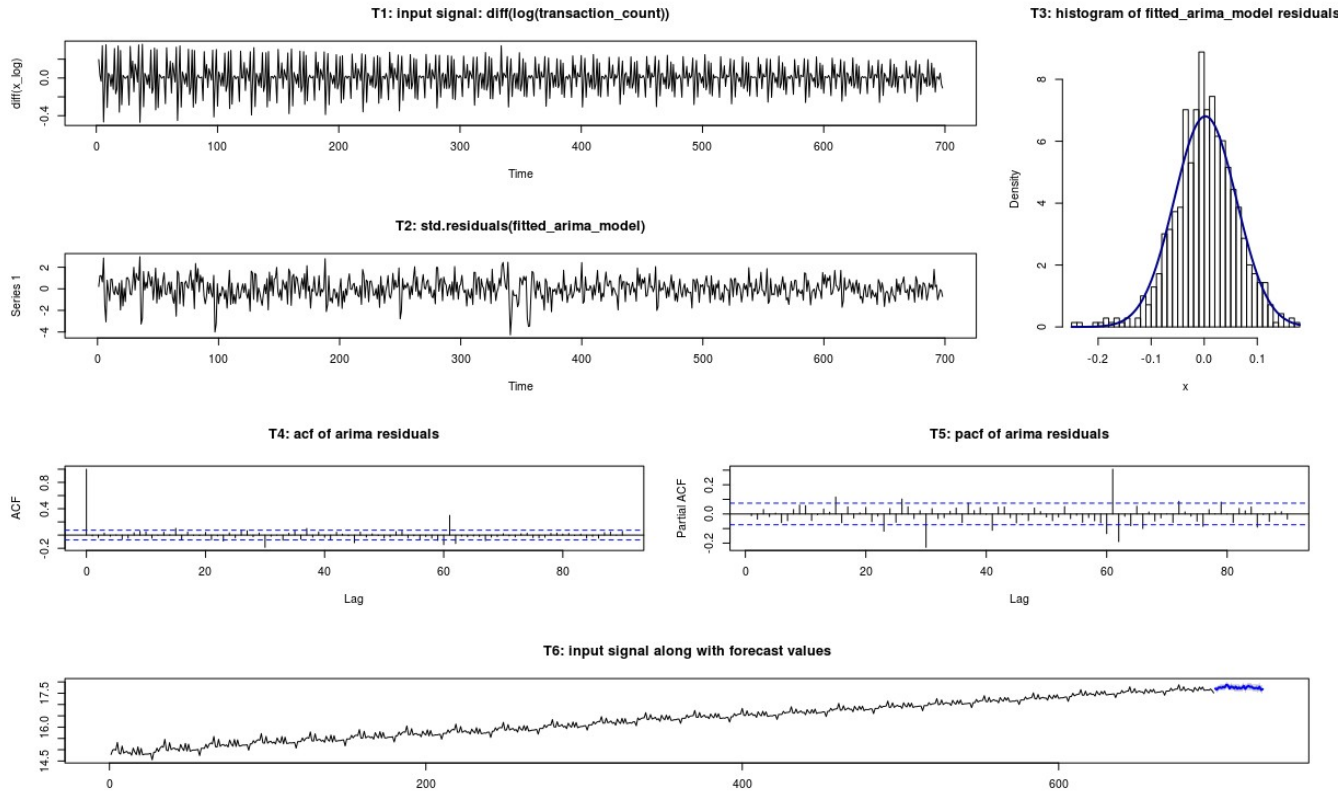

Illustration 2: TRANSACTION_COUNT: T1 (input signal), T2(residuals), T3(hist. of residuals), T4 (acf), T5 (pacf), T6 (forecast). All are in accordance to expectations, except for a mild seasonality spike at 30 & 60, indicative of an additional AR degree.

C. Exploratory Modeling

After exploring several grid-search ***auto-arima*** models using ***AIC***-based exploration model choices, I found that the resulting ***acf*** plots always exhibited higher-order lag spikes that I could not fix with small degree **SEASONAL.ARIMA** models. However, trial-and-error exploration of the higher-degree model choices, yielded a misleading ***AIC*** plateau. I decided to investigate lag significance using high degree ***ARMA*** models and cherry picked lags into an ***ARMA*** model that: **1)** had an acceptable ***acf***, **2)** had a low ***AIC***, and ***3)*** had only few and all highly significant lags. I integrated these highly significant lags back into a high-order **SEASONAL.ARIMA** model and via trial-and-error, I stepwise added missing lags to remove spike in the resulting ***acf*** plot. At the end, the resulting model had better residuals, lower **AIC**, large **p-values** for more lags on the **Ljuang-Box** test, almost no significant spikes but along with a seemingly random pattern on its acf plot, etc. The resulting model however, had only one statistically significant seasonal acf spike at lag ***30*** which at the current time I been not able to remove.

An cherry-picked lag ***SEASONAL.ARIMA(30,1,3)(1,0,1)[12]*** model was built using the most significant lags as given below. Note that lags ***6, 7, 11, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 29*** of this high degree model are actually set to ***0***. The hints of which lags to retain was obtained from the statistical significant of lags under a similar ***ARMA*** model and then followed by migration and trial-and-error under to the ***SEASONAL.ARIMA*** model. The analysis is documented on the attached ***R*** script.

```
Series: x_log
ARIMA(30,1,3)(1,0,1)[12]
Coefficients:
       ar1    ar2  ar3    ar4   ar5  ar6  ar7  ar8    ar9  ar10  ar11   ar12  ar13   ar14  ar15   ar16  ar17  ar18
      0.68  -0.10    0  -0.08  0.07    0    0    0  -0.25  0.32     0  -0.33  0.30  -0.10  0.18  -0.17     0     0
s.e.  0.03   0.03    0   0.03  0.03    0    0    0   0.03  0.03     0   0.04  0.05   0.04  0.04   0.03     0     0
      ar19  ar20  ar21   ar22  ar23  ar24  ar25  ar26  ar27  ar28  ar29  ar30    ma1   ma2    ma3  sar1  sma1
         0     0     0  -0.06  0.09     0     0     0     0     0     0  0.43  -2.23  1.93  -0.66  0.45  -0.1
s.e.     0     0     0   0.03  0.03     0     0     0     0     0     0  0.02   0.04  0.06   0.03  0.09   0.1
sigma^2 estimated as 0.00344:  log likelihood=973
AIC=-1906   AICc=-1905   BIC=-1816
Training set error measures:
                ME RMSE  MAE  MPE MAPE MASE  ACF1
Training set 0.002 0.06 0.04 0.01  0.3  0.4 -0.01
```

Illustration 2 provides a visual summary of the analysis of the ***TRANSACTION_COUNT*** time series. The ***T1*** plot shows the ***diff(log(TRANSACTION_COUNT), -1)*** time series, showing a stationary time series. Plot ***T2*** shows the residuals resulting from the fitting of the above ***SEASONAL.ARIMA(30,1,3)(1,0,1)[12]*** model to the series (with time shown in calendar days). The histogram of the residuals is shown in ***T3*** and it is *relatively* normal and zero-centered. Plot ***T4*** and ***T5*** show the ***ACF*** and ***PACF*** for the residuals, both showing the almost absence of statistically significant spikes – with the one ***acf*** exception stated at lag ***30***, such repeating seasonally. Plot ***T6*** illustrates, within the context of the original Transaction_Count series, the ***March 2015*** forecast. Finally, although not shown, the model's ***p-values*** for the ***Ljung-Box statistic*** were high to relatively high for a *reasonable* number of initial lags.

D. Forecast

The forecast values for the month of ***March 2015*** are given below. As given in the specified input data, a gap format is also used. Specifically, even though the model implicitly generates forecasts for ***5th, 11th, 14th, 17th, 24th, 27th, 29th, 30th, or 31st***, the semantics of the problem specify such as gaps and as a result, the forecasts for these dates are simply ***dropped*** *as* itemized and tabulated below**.**

2015-03-01	2015-03-02	2015-03-03	2015-03-04	2015-03-05	2015-03-06	2015-03-07	2015-03-08	2015-03-09	2015-03-10	2015-03-11	2015-03-12
48752739	46908092	50811312	51554365	50408377	52808374	51688047	57742391	56451900	49137468	53537048	52391271
2015-03-13	**2015-03-14**	**2015-03-15**	**2015-03-16**	**2015-03-17**	**2015-03-18**	**2015-03-19**	**2015-03-20**	**2015-03-21**	**2015-03-22**	**2015-03-23**	**2015-03-24**
49715902	53141755	48785146	51353173	49213658	50904072	54861214	48335033	52857320	55128337	52662197	53203991
2015-03-25	**2015-03-26**	**2015-03-27**	**2015-03-28**	**2015-03-29**	**2015-03-30**	**2015-03-31**					
49984087	49674495	50619107	48692084	52903100	46163523	47674723					

t	Forecast	Lo 80	Hi 80	Lo 95	Hi 95	march_dates	transaction_count
699	17.70	17.63	17.78	17.59	17.82	2015-03-01	48752739
700	17.66	17.58	17.75	17.53	17.79	2015-03-02	46908092
701	17.74	17.66	17.83	17.61	17.88	2015-03-03	50811312
702	17.76	17.67	17.85	17.62	17.89	2015-03-04	51554365
703	17.74	17.65	17.83	17.60	17.87	2015-03-05	50408377
704	17.78	17.69	17.87	17.64	17.92	2015-03-06	52808374
705	17.76	17.67	17.85	17.62	17.90	2015-03-07	51688047
706	17.87	17.78	17.96	17.73	18.01	2015-03-08	57742391
707	17.85	17.76	17.94	17.71	17.99	2015-03-09	56451900
708	17.71	17.62	17.80	17.57	17.85	2015-03-10	49137468
709	17.80	17.70	17.89	17.65	17.94	2015-03-11	53537048
710	17.77	17.68	17.87	17.63	17.92	2015-03-12	52391271
711	17.72	17.62	17.82	17.57	17.87	2015-03-13	49715902
712	17.79	17.69	17.89	17.64	17.94	2015-03-14	53141755
713	17.70	17.60	17.80	17.55	17.85	2015-03-15	48785146
714	17.75	17.65	17.86	17.60	17.91	2015-03-16	51353173
715	17.71	17.61	17.81	17.56	17.87	2015-03-17	49213658
716	17.75	17.64	17.85	17.59	17.90	2015-03-18	50904072
717	17.82	17.72	17.92	17.66	17.98	2015-03-19	54861214
718	17.69	17.59	17.80	17.54	17.85	2015-03-20	48335033
719	17.78	17.68	17.89	17.63	17.94	2015-03-21	52857320
720	17.83	17.72	17.93	17.67	17.98	2015-03-22	55128337
721	17.78	17.67	17.88	17.62	17.94	2015-03-23	52662197
722	17.79	17.68	17.90	17.63	17.95	2015-03-24	53203991
723	17.73	17.62	17.83	17.56	17.89	2015-03-25	49984087
724	17.72	17.61	17.83	17.56	17.88	2015-03-26	49674495
725	17.74	17.63	17.85	17.57	17.91	2015-03-27	50619107
726	17.70	17.59	17.81	17.54	17.87	2015-03-28	48692084
727	17.78	17.67	17.89	17.62	17.95	2015-03-29	52903100
728	17.65	17.54	17.76	17.48	17.82	2015-03-30	46163523
729	17.68	17.56	17.80	17.50	17.86	2015-03-31	47674723

E. Software:

The script implementing the approach described above is found on: "**verizon_transactions.R".**