A Brief Explanation of the Holt-Winters Forecasting Method

The Holt-Winters forecasting method is a part of a family of time-series forecasting methods based upon the idea of exponential smoothing. At its essence exponential smoothing is a weighted average of past values. New predictions, $\hat{y}_{n+1|n}$, are generated by the following equation:

(1) $\hat{y}_{n+1|n} = \sum_{i=0}^{\infty} w_i y_{n-i}$

What makes the model *exponential* is that the weights decline by a constant ratio, with the most recent observation given the highest weight. Hence the weights are a geometric series of the form:

(2) $w_i = \alpha(1-\alpha)^i$ where $\alpha \in (0,1)$

α is the *smoothing constant*. Obviously a high value of α shifts the weight to the most recent observations and vise versa. Typically it is set between 0.05 and 0.3, but R optimizes for this parameter by minimizing the sum of squared errors. Thus simple exponential smoothing gives us the following model.

(3) $\hat{y}_{n+1|n} = \sum_{i=0}^{\infty} \alpha(1-\alpha)^i y_{n-i} = \alpha y_n + (1-\alpha)\hat{y}_{n|n-1}$

where $\hat{y}_{n|n-1}$ is the previous estimate.

Unfortunately, this technique is limited to data without systematic trend or seasonal components. The Holt-Winters method extends the above model to encompass these variations. At each point we extrapolate a linear model to forecast the next data point, but the parameters of the linear model are generated using the aforementioned simple exponential smoothing model. The method's general forecast function is:

(4) $\hat{y}_{n+l|n} = c_{n-s+l}(m_n + l b_n)$

where l is the number of time steps ahead to forecast, m_n is the level component, b_n is the trend component (slope), and c_{n-s+l} is the relevant seasonal component, with s signifying the seasonal period (e.g. 4 for quarterly data and 12 for monthly data.) All three components m_n, b_n, c_{n-s+l} are constructed using exponential smoothing.

To construct the update equation for m_n we simply "de-season" the standard exponential smoothing model by dividing y_n by c_{n-s}:

(5) $m_n = \alpha \frac{y_n}{c_{t-s}} + (1-\alpha)(m_{n-1} + b_{n-1})$

The last term is equivalent to y_{n-1} de-seasoned by c_{n-s-1}. Similarly we use the difference between the *levels* $m_n - m_{n-1}$, the difference equation per the slope, to update the slope parameter b_n:

$$(6)\ b_n = \beta(m_n - m_{n-1}) + (1-\beta)b_{n-1}$$

And last but not least, the seasonal component is updated by "de-leveling" y_n by m_n:

$$(7)\ c_n = \gamma \frac{y_n}{m_n} + (1-\gamma)c_{n-s}$$

Additionally we can correct for the autocorrelation of residuals by adding the term $r_n\varepsilon_n$ to equation (4) where r_n is the autocorrelation coefficient, and ε_n is the prior step forecast error.

The Problem of Non-Stationarity

The Holt-Winters method was conceived in 1960 by Peter Holt, who was Charles Winters' student at the time. Despite its age, it remains one of the best and most popular time forecasting methods. However, stationarity is its Achilles' Heel. This is in fact the central assumption of most time-series forecasting methods. In short it states that the first and second order moments are unaffected by a change of time origin (i.e. a lag), that is the model's average and variance for level ($E[m_n]$ & $var(m_n)$) are constant over any time interval with covariance and correlation of Y_n being functions of the time lag only. Thus we have the following restrictions on the Holt-Winters model:

a) One and only one trend exists for all points
b) Seasonal parameters do not vary over time
c) No autoregressive structure or adaptive model structure
d) No outliers
e) No level shifts in the data i.e. no intercept changes
f) Model errors have constant mean and variance (homoscedastic)

It is the first two restrictions that are the most problematic. Looking at the data, it is apparent that the same trend does NOT exist for all points. Peak hours of the day have an overall increasing trend (with different trends for the different peak periods of the day) and at night the trend over time is flat. There is next to no traffic. Furthermore the seasonal parameters DO vary over time. If one takes the week as the season interval the month as the period of season, one can see that from the first month to the second month the peaks noticeably increase. Similarly if we take the model to be the 168 hours of the week as the season and the period week, there is a noticeable change of in the level of the peaks from week to week within the month.

Furthermore tests for non-stationarity following the Box-Jenkins methodology are all positive to the second degree! That is, we must difference the time series twice to get a signal that does not reject the stationary null using the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test. The results are in the below table.

# of Differences	KPSS Test Statistic	P-Value
0	0.828	0.01
1	0.0685	0.05
2	0.0044	0.1

Table 1: KPSS test for Stationarity (Null = Stationary)

An Innovative solution to Non-stationarity

The preferred approach for dealing with non-stationarity are Seasonal Autoregressive Integrated Moving Average (SARIMA) models. These incredibly versatile models that relax restrictions (a), (b), and (c) above. We will not get into the math behind these models, but the degrees to which the signal is nonstationary specifies the degrees of the polynomial used to predict $\hat{y}_{n+1|n}$ from y_n given a non-seasonal model with no auto-regressive lag (i.e. we only look at the immediate prior point to forecast the next point). In a model with lag and seasonality the number of parameters now grows with the number of seasons plus the number of lags. As a result these models are incredibly difficult to parameterize properly and require an intense amount of computation when searching through the parameter space to optimize. The number of parameters for a weekly forecast model given our dataset would be 3 (second degree polynomial) + 24 (number of hourly lags calculated from the Box-Jenkins approach)+ 168 (each hour of the week is a season). It is nearly impossible specify such a sophisticated model.

Additionally the autocorrelation (AC) and partial autocorrelation (PAC) figures below (once again following the Box-Jenkins methodology) indicate that the moving average component of the model contains more predictive power than the auto-regressive component. Notice that the PAC rapidly falls off after the first peak, suggesting the number of past points necessary in the autoregressive component.

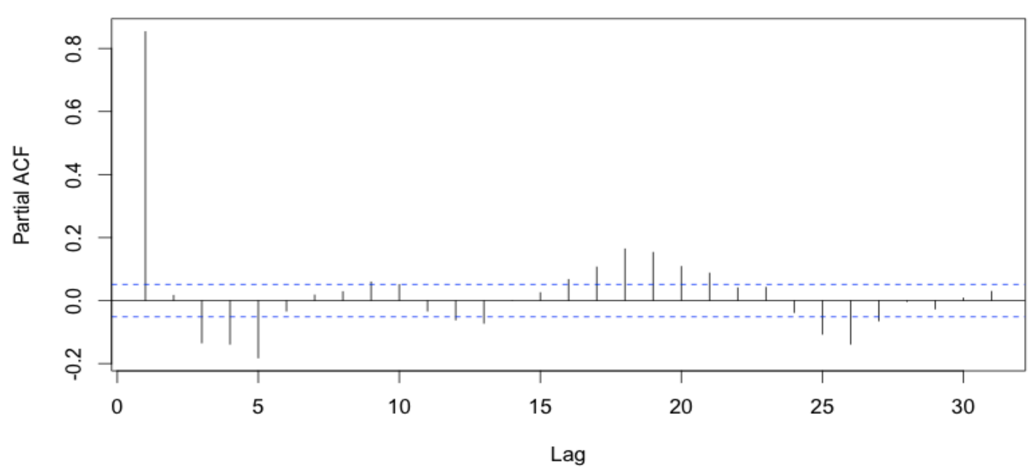

Figure 1: Partial Autocorrelation of Series

AC on the other hand exceeds the significance bounds 30 lags out. This of course is unsurprising given seasonality, but gives credence to our selection of the Holt-Winters model. It is fundamentally a moving average model.

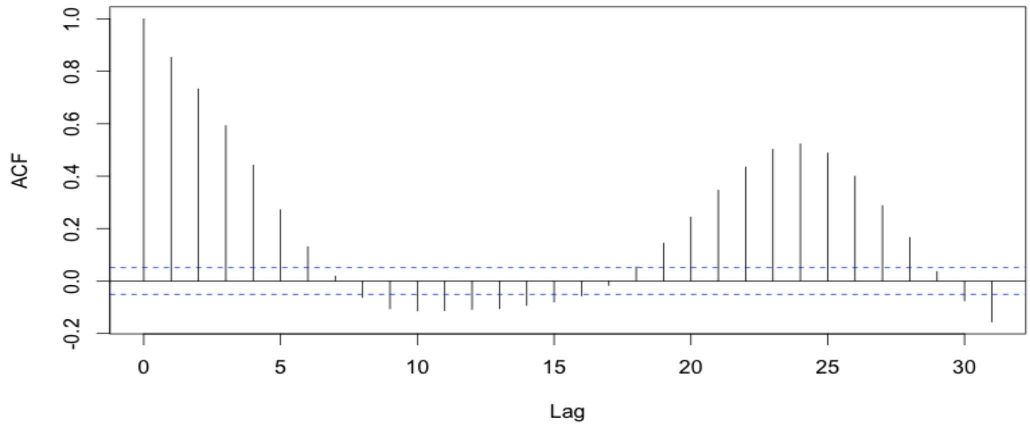

Figure 2: Partial Autocorrelation

<u>The NESTED Holt-Winters Model</u>

My approach to tackle non-stationarity can be accused of “statistical redneckism” partially due to my own naivette in time-series forecasting, but it is simple—almost stupidly simple—fast, and innovative. I have not seen anything like this in any of the literature I have reviewed.

Examining the data one sees two periodicities of seasonality. The most obvious is at the week level. The same days of every week have the same relative level intra-week. Peaks gradually build from Monday to Friday, explode on Saturday and crash to pre-Friday levels on Sunday. Levels are also remarkable consistent at the hourly level. The intra-day pea is slightly after noon, with a secondary peak around dinnertime, and two smaller peaks on either side—in the morning when people head to work and late in the evening when people come home from their various outings. There is remarkable consistency in the relative intra-week levels of each hour of the week from week to week.

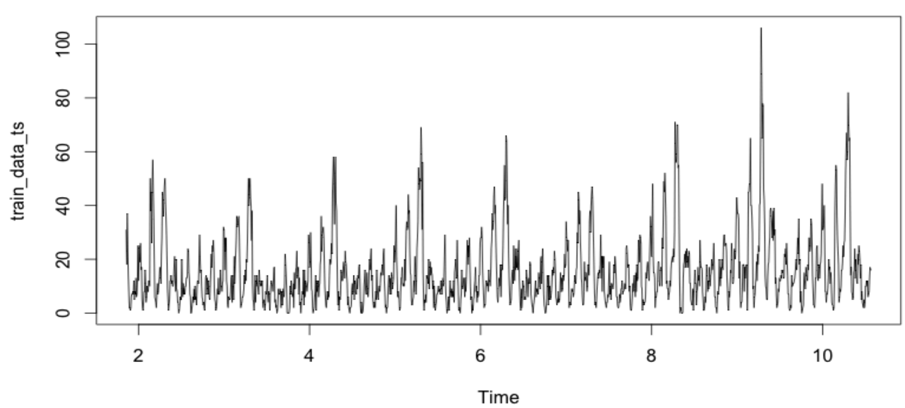

Figure 3: Hourly Traffic

The second period of seasonality follows from week to week over the month. Just examining the Saturday peaks, one can see that the second Saturday of every month has the lowest peak, whereas the 4th Saturday has the highest. The first of Saturday of the next month then tapers off from the prior Saturday eventually reach the nadir on the second and climbing up again on the 3rd Saturday. This trend extrapolates to the Friday and Sunday peaks as well.

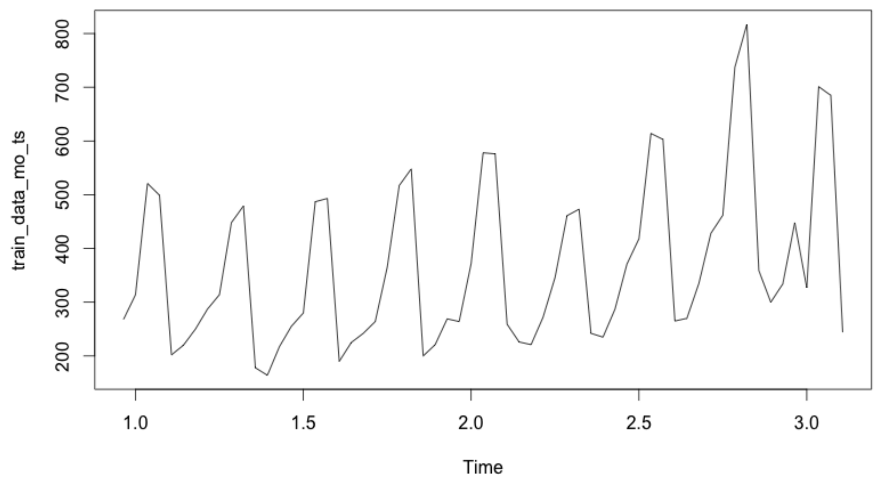

Figure 4: Daily Traffic

Hence my solution was very simple. Notice that if we force β=0 in equation (6) that the trend b_n will be zero. This will then satisfy the restrictions (a), (b), and (c). The former is the most important of these restrictions, for as we mentioned there is not a single underlying trend for all data points in the sample. If we construct a model using hours as the season and the week as the period, the forecast, because the trend is zero, will necessarily be stationary over one week intervals. That is the forecast will simply repeat itself after a period of a week.

Fortunately, if we sum the hours to show daily sums over time, we satisfy the stationarity condition on the level and restriction (a)! Daily sums are remarkably consistent relative to the month they belong to, and because there is no floor they must sink down to (0 for hourly intervals) a single trend can de-bias all the data points to impose stationarity once the data points have also been de-seasoned.
Ergo, we use daily sums to construct a monthly forecast model (forecasts daily sums for a specified number of months — $\hat{y}_{d+1|d}$), use hourly sums to construct a weekly forecast model (forecasts hour sums for a one week period — $\hat{y}_{h+1|h}$), and then scale the weekly forecast by the monthly forecast, producing the following model:

(8) Monthly Model: $\hat{y}_{d+1|d} = c_{d-D}(m_d + b_d)$ †*D* denotes monthly seasonality

(9) Weekly Model: $\hat{y}_{h+1|h} = c_{h-H}(m_h)$ †*H* denotes weekly seasonality

(10) Scaling Coefficients: $s_{d+1|h} = \frac{\hat{y}_{d+1|d}}{\sum_{h\epsilon(d+1)}\hat{y}_{h+1|h}}$

(11) Combined Model: $\tilde{y}_{h+1|h} = s_{d+1|h} * \hat{y}_{h+1|h} = \frac{c_{h-H}\, c_{d-D}(m_h)(m_d+b_d)}{\sum_{h\epsilon(d+1)}\hat{y}_{h+1|h}}$

Performance Characteristics

Unfortunately training any seasonal model requires at least two whole periods of data, and in our case we had two exactly full months of data. Hence we have no out-of-sample data to test the full model with. However, for in-sample testing, we can see from the below figure that the model works incredibly well.

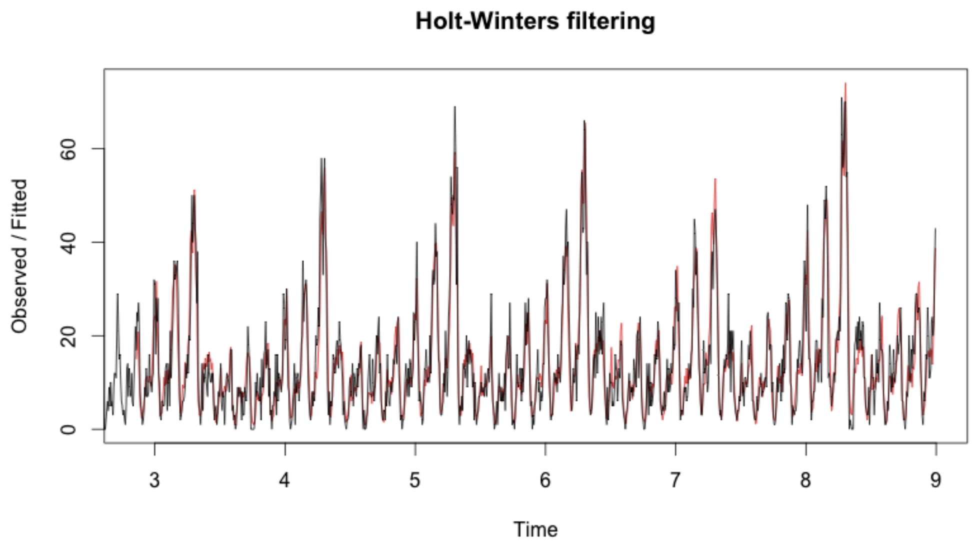

Figure 5: Combined Model Results (Actual=Black, Combined=Orange)

The Mean Absolute Percentage Error is approximately 14.7% († this calculation excluded points where the actual forecast was zero). A moving average model naturally is weak at the extrema. It undershoots at the peaks, and overshoots at the troughs.

Second most importantly, we can see that our errors are normal! This satisfies the homoscedasticity assumption we had to make. The standard deviation of our error is approximately 2.9 which is quite reasonable.

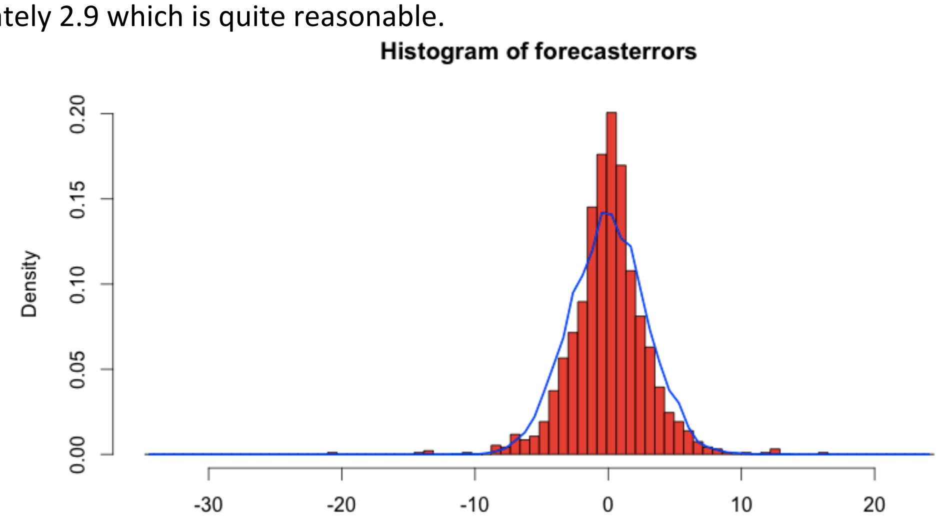

Figure 6: Normality of In-sample Errors

Finally, the partial autocorrelation function shows that for the most part our errors are uncorrelated (we ignore the lag at 0). There is some correlation at one lag beyond the 5% significance level, but the degree is still minimal.

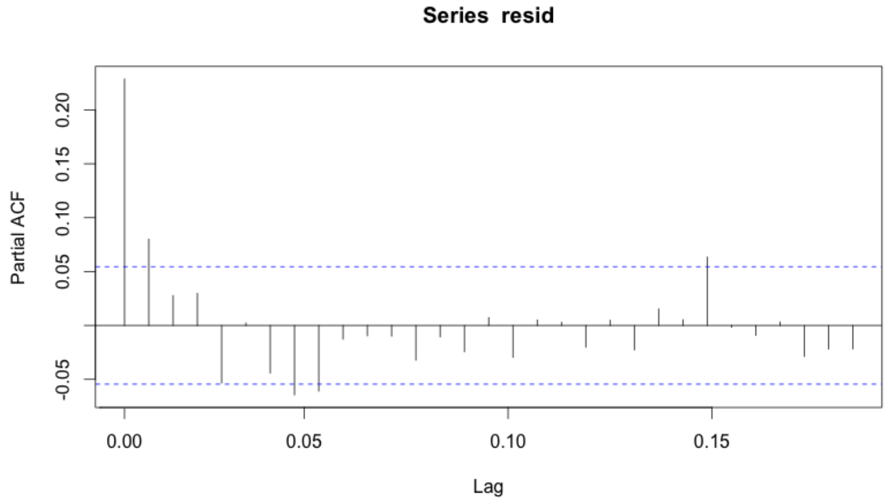

Figure 7: Partial Autocorrelation of In-sample Errors

The result is verified by the Ljung-Box test on the errors, which returns a p-value of 0.10 (R would not return an exact value) for the null that the data is independent and identically distributed.

As per out-of-sample testing, we are limited to testing only the weekly model; however, this model is the primary component of our combined forecasting model. The out-of-sample test was performed on the final 264 hours of the dataset, using the first 1200 hours (7+ weeks) exclusively for training.

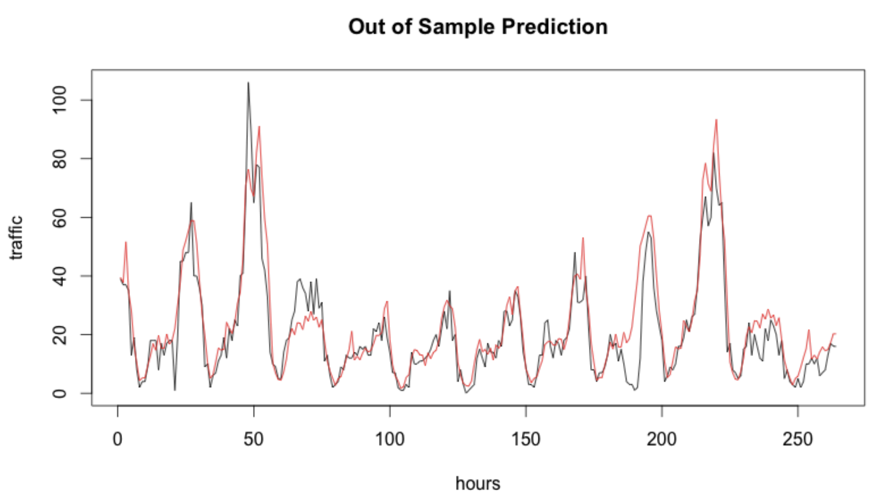

Figure 8: Out of sample test on final 264 hours of dataset (Black=Actual, Red=Predicted)

As one can see the results are not nearly as good. Our mean absolute percentage error is now 83.7%, driven primarily by the error on Friday of the week (between hours 168 and 196). Its important to note that measuring performance characteristics as a single metric is quite difficult because errors at the extrema can dramatically increase measurements.

One can also see from the histogram of errors that their exists negative skew in the distribution (mass to the right). This is expected as there exists trend in the data but that is only captured by the monthly model which is not included in the out-of-sample forecast. Nonetheless the performance is significantly worse. Our standard deviation has now increased to 7.75.

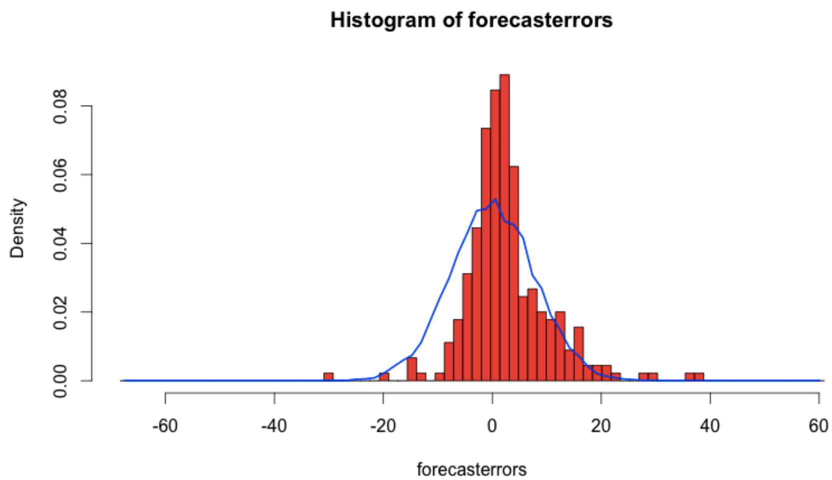

Figure 9: Normality of Out-of-Sample Errors

Fortunately the partial auto-correlation results maintain the original in-sample finding that the errors are uncorrelated. The Ljung-Box test gives the same p-value of 0.10 for the null that the errors are independent and identically distributed.

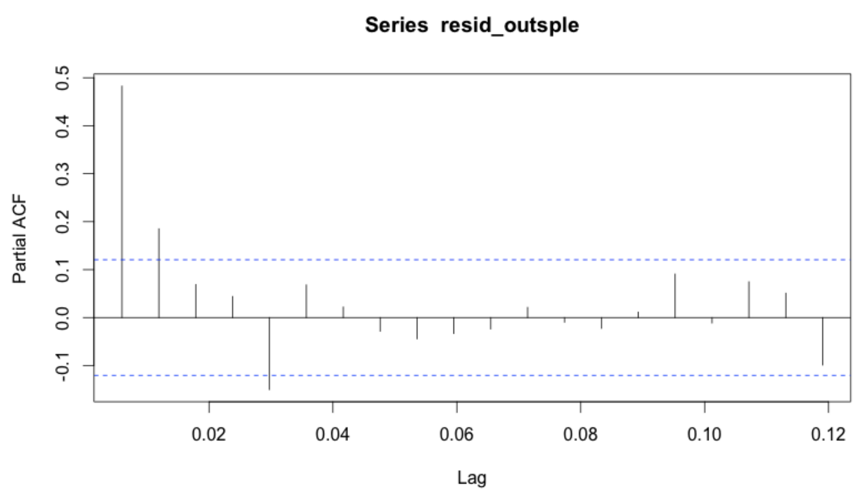

Figure 10: Partial Autocorrelation of out-of-sample errors

Finally the below plot demonstrates the limits of our weekly forecast model. The bands denote the variance about the prediction (mean). For the first week, the model's variance is quite low. However, in the second week the variance increases dramatically. The variance is as high as 200, twice the mean at some of the latter hours. Thus a weekly predictive model is of little value beyond the first week of prediction.

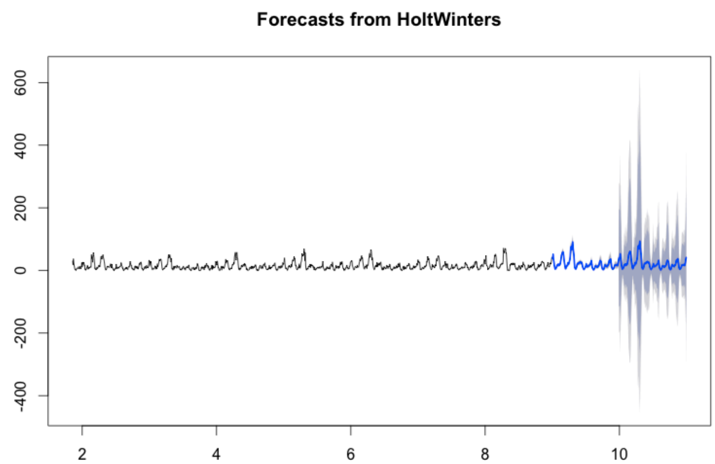

Figure 11: Variance of weekly forecasts

Potential Improvements

The beauty of this modeling approach is that it can be nested in all sorts of ways depending on the periodicities of seasonality and trend. I chose the minimal amount of nesting for performance purposes, but for example we could use hourly sums to forecast days, daily sums to forecast weeks, weekly sums to forecast months and create a triple nested model.

Furthermore it can be combined with multiple kinds of models as well. The SARIMA approach is in every way worth considering in a triple-nested model. Although slower, such an approach dramatically reduces the number of parameters and there is indeed a need to incorporate auto-regressive lag into the model structure. A production version of such a forecasting method should indeed consist of the optimal combination of Holt-Winters and SARIMA in the optimal nested pattern.

Last but not least there is a chance that our model was suffering from overfitting issues as R optimizes to min the in-sample error. It would be ideal to train the model to min-the out-of-sample error on a long string of data.

Finally it should be noted that an entirely alternate approach for forecasting non-stationary seasonal time series is wavelet process modeling. This is a relatively new method, but seems to be quite promising per the literature. The forecasting method relies upon decomposing the time signal into waves (just like the Fourier transform) however each wave is allowed scaling, bias, and trend in time (imagine a Fourier transform where we keep the wave in the time domain). Such a model would allow us to capture the trend and seasonalities that occur at many different levels in this data structure.