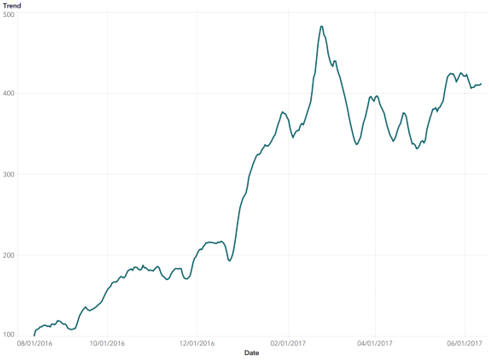
Time Series Modelling

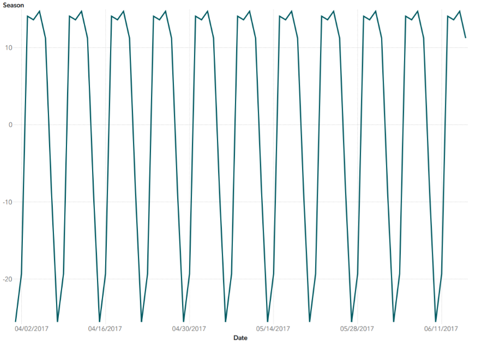
TSM is very good at using historical data to identify existing patterns and use them to predict what will happen in the future. Unlike most advanced analytics solutions, time series modeling is a low-cost solution that can provide powerful insights.

This post will walk through the three fundamental steps of building a quality time series model: making data stationary, selecting the right model, and evaluating model accuracy.

**Step 1: Making Data Stationary**

Time series involves the use of data that are indexed by equally spaced increments of time (minutes, hours, days, weeks, etc.). Due to the discrete nature of time series data, many time series data sets have a seasonal and/or trend element built into the data. The first step in time series modeling is to account for existing seasons (a recurring pattern over a fixed period of time) and/or trends (upward or downward movement in the data). Accounting for these embedded patterns is what we call making the data stationary.  Examples of trending and seasonal data can be seen if figures 1 and 2 below.

  
*Figure 1: Example of Upward Trending Data*

  
*Figure 2: Example of Seasonal Data*

### What is Stationarity?

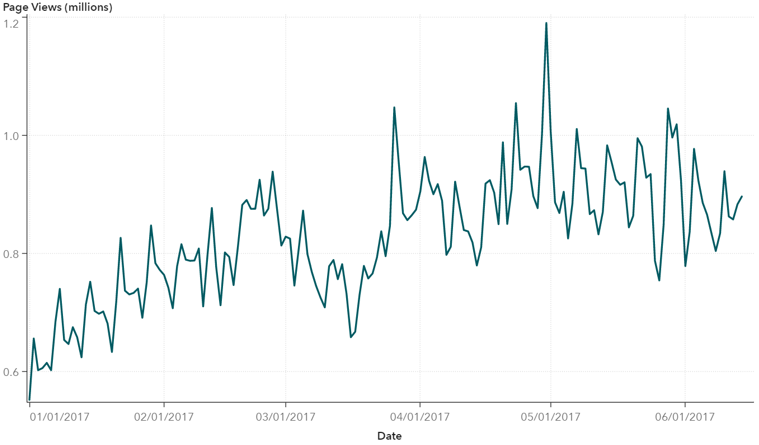
As we previously mentioned, the first step in time series modeling is to remove the effects of the trend or season that exist within the data to make it stationary.

A stationary series is one where the mean of the series is no longer a function of time. With trending data, as time increase the mean of the series either increases or decreases with time (think of the steady increase in housing prices over time). For seasonal data, the mean of the series fluctuates in accordance with the season (think of the increase and decrease in temperature every 24 hours).

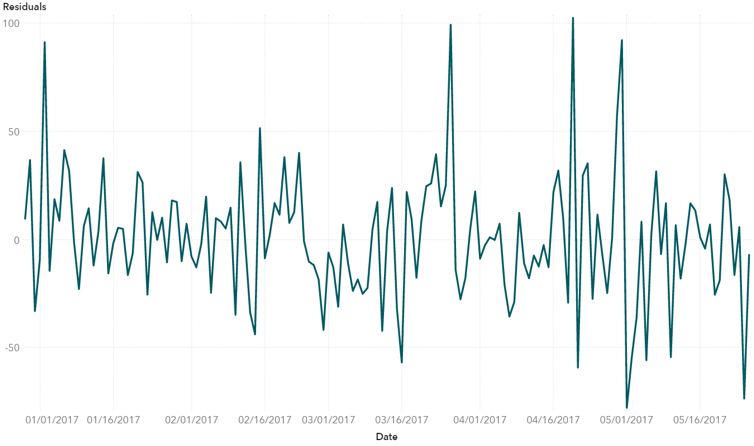
### How do we achieve stationarity?

There are two methods that can be applied to achieve stationarity, difference the data or linear regression. To take a difference, you calculate the difference between consecutive observations. To use linear regression, you include binary indicator variables for your seasonal component in the model. Before we decide which of these methods to apply, let’s explore our data.

Historical Daily page views



The initial pattern seems to repeat itself every seven days indicating a weekly season. The prolonged increase in the number of page views over time indicates that there is a slightly upward trend. With a general idea of the data we then applied a statistical test of stationarity, the Augmented Dickey-Fuller (ADF) test. The ADF test is a unit-root test of stationarity. We won’t get into the details here, but a unit-root indicates if the series is nonstationary so we use this test to determine the appropriate method to handle the trend or season (differencing or regression). Based on the ADF test for the data above we removed the seven-day season by regressing on dummy variables for day of the week and removed the trend by differencing the data. The resulting stationary data can be seen in the figure below.



### Building Your Time Series Model

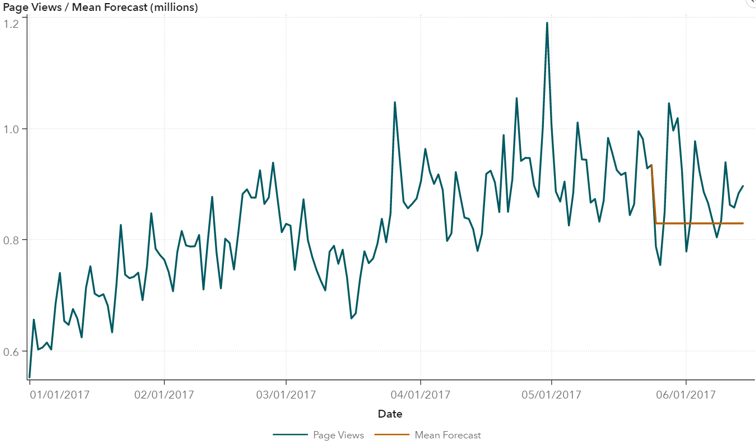
Now that the data is stationary, the second step in time series modeling is to establish a base level forecast. We should also note that most base level forecasts do not require the first step of making your data stationary. This is only required for more advanced models such as ARIMA modeling

### Establish a Base Level Forecast

There are several types of time series models. To build a model that can accurately forecast future page views (or whatever you are interested in forecasting), it is necessary to decide on the type of model that is appropriate for your data.

The simplest option is to assume that future values of y (the variable you are interested in forecasting) are equal to the most current value of y. This is considered the most basic, or “naïve model”, where the most recent observation is the most likely outcome for tomorrow.

A second type of model is the average model. In this model, all observations in the data set are given equal weight. Future forecasts of y are calculated as the average of the observed data. The forecast generated could be quite accurate if the data is level, but would provide a very poor forecast if the data is trending or has a seasonal component. The forecasted values for the page views data using the average model can be seen below.



If the data has either a seasonal or trend element, then a better option for a base level model is to implement an exponential smoothing model (ESM). ESMs strike a happy medium between the naïve and average models mentioned above, where the most recent observation is given the greatest weight and the weight of all previous observations decrease exponentially into the past. ESMs also allow for a seasonal and/or trending component to be incorporated into the model. The following table provides an example of an initial weight of 0.7 decreasing exponentially at a rate of 0.3.

|  |  |
| --- | --- |
| **Observation** | **Weight (α = 0.7)** |
| Yt (current observation) | 0.7 |
| Yt-1 | 0.21 |
| Yt-2 | 0.063 |
| Yt-3 | 0.0189 |
| Yt-4 | 0.00567 |

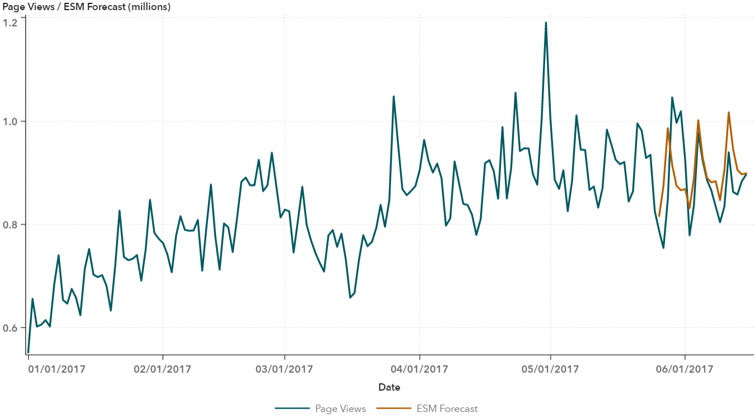
*Table 1: Example of the exponentially decreasing effect of past observations of Y.*

There are various types of ESMs that can be implemented in time series forecasting. The ideal model to use will depend on the type of data you have. The table below provides a quick guide as to what type of ESM to use depending on the combination of trend and season in the data.

|  |  |  |
| --- | --- | --- |
| **Type of Exponential Smoothing Model** | **Trend** | **Season** |
| Simple ESM |  |  |
| Linear ESM | X |  |
| Seasonal ESM |  | X |
| Winters ESM | X | X |

*Table 2: Model Selection Table*

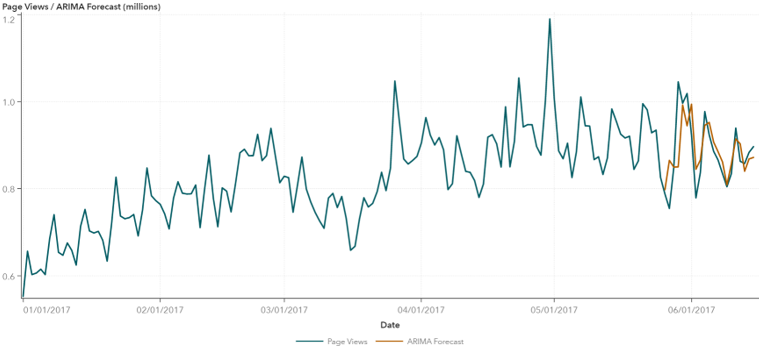
Because of the strong seven-day season and upward trend in the data, we selected an additive winters ESM as the new base level model. The forecast generated does a decent job of continuing the slight upward trend and captures the seven-day season. However, there is still more pattern in the data that can be removed.

  
*Figure 6: Additive Winters ESM Forecast*

**ARIMA Modeling**

After identifying the model that best accounts for the trend and season in the data, you ultimately have enough information to generate a decent forecast, as we see in Figure 2 above. However, these models are still limited in that they do not account for the correlation that the variable of interest has with itself over previous periods of time. We refer to this correlation as autocorrelation, which is commonly found in time series data. If the data has autocorrelation, as ours does, then there may be additional modeling that can be done to further improve upon the baseline forecast.

To capture the effects of autocorrelation in a time series model, it is necessary to implement an Autoregressive Integrated Moving Average (or ARIMA) model. ARIMA models include parameters to account for season and trend (like using dummy variables for days of the week and differencing), but also allow for the inclusion of autoregressive and/or moving average terms to deal with the autocorrelation imbedded in the data. By using the appropriate ARIMA model, we can further increase the accuracy of the page views forecast as seen in Figure 3 below.

  
*Figure 7: Seasonal ARIMA Model Forecast*

**Step 3: Evaluating Model Accuracy**

While you can see the improved accuracy of each of the models presented, visually identifying which model has the best accuracy is not always reliable. Calculating the MAPE (Mean Absolute Percent Error) is a quick and easy way to compare the overall forecast accuracy of a proposed model – the lower the MAPE the better the forecast accuracy. Comparing the MAPE of each of the models previously discussed, it is easy to see that the seasonal ARIMA model provides the best forecast accuracy. Note that there are several other types of comparison statistics that can be used for model comparison.

|  |  |
| --- | --- |
| **Model Error Validation** | |
| **Model** | **MAPE** |
| Naïve | 9.0 |
| Mean | 10.2 |
| Winters ESM | 6.9 |
| Seasonal ARIMA | 4.4 |

*Table 3: Model Error Rate Comparison*

**Summary**

In summary, the trick to building a powerful time series forecasting model is to remove as much of the noise (trend, season, and autocorrelation) as possible so that the only remaining movement unaccounted for in the data is pure randomness. For our data we found that a seasonal ARIMA model with regression variables for day of the week provided the most accurate forecast. The ARIMA model forecast was more accurate when compared to the naïve, average, and ESM models mentioned above.