

The difference between R's auto.arima and Robust Modeling. Think All Blacks vs Wallabies. Light years. Generational Gap. Neandrathal vs Human. Night and Day.

Posted on Jun 15, Posted by [Tom Reilly](#) Category [Forecasting](#)

Let's evaluate auto.arima vs a Robust Solution from an example by it's author, Rob Hyndman, in his [R text book](#) .

Do this exercise in R and see for yourself!

Now, this is only one example that we are discussing here, but it reveals so much about R's auto.arima. It's a "pick best" approach of models that minimizes the AIC with no attempt to diagnose other effects like outliers, changes in level/trend/seasonality. There are many other examples we have seen that show flaws in auto.arima as we have seen them discussed on crossvalidated.com.

The data set we are examining is found in Chapter 8.5 of Rob's book called "U.S. Consumption" and it is a quarterly dataset with 164 values. Rob observes no seasonality, so he forces the model to not look for seasonality. Now this might be true, but there is something else going on here that smells funny as the name of the software is "auto", but a human is intervening. Is it really necessary to have the need for this? What consequences would happen if the user wasn't there to do this we wondered? We explore that down below.

The auto.arima generated model is an MA3 with lags 1,2,3 and a constant. All coefficients are significant and the forecast looks good so everything is great, right? Well, not really.

```
Series: usconsumption[, 1]
ARIMA(0,0,3) with non-zero mean

Coefficients:
      ma1      ma2      ma3  intercept
      0.2542  0.2260  0.2695      0.7862
s.e.    0.0767  0.0779  0.0692      0.0844

sigma^2 estimated as 0.3856: log likelihood=-154.73
AIC=319.46   AICc=319.84   BIC=334.96

Training set error measures:
              ME          RMSE         MAE      MPE      MAPE         MASE       ACF1
Training set -4.013475e-05  0.6209988  0.4578466 -Inf   Inf  0.6590496  0.0101825
```

The forecasts from a Robust application(Autobox) and auto.arima end up being just about the same, but the "style" and "process" from these two tools need to be highlighted as their are stark differences in the modeling assumptions and detection of other patterns. Box-Jenkins laid out a path of Identification, Estimation, Necessity and Sufficiency with perhaps model revision and then forecasting. Auto.arima is not following this path. Auto.arima's approach is a "one and done" approach where the model is identified and that it is it.

Let's see what is really going on in terms of methodology and what assumptions are made and what is missing in auto.arima. auto.arima's process leaves a set of residuals which are obviously NOT random. The residuals show artifacts from which information can be learned about the data and perhaps a better model. Here are the auto.arima's residuals which clearly show that the first half of the data is very different from the second half. Do residuals matter? Yes. The first thing you learn when studying time series is that if the [errors are not random](#) and show pattern(s) then your model is not sufficient and needs more work.

