Exploring meaningful visual effects and quantities of interest from dynamic models through dynamac

Soren Jordan[1] **and Andrew Q. Philips**[2]

1 Assistant Professor, Department of Political Science, Auburn University **2** Assistant Professor, Department of Political Science, University of Colorado Boulder

DOI: 10.21105/joss.02528

Software

- Review
- Repository
- Archive

Editor: Sebastian Benthall

Reviewers:

- @stulacy
- @dormezil

Submitted: 27 May 2020
Published: 10 October 2020

Summary

dynamac (2020) implements a framework of estimating and interpreting autoregressive distributed lag (ARDL) models in R. dynamac uses stochastic simulation techniques (Jordan & Philips, 2018a, 2018b) to easily recover traditional quantities of interest, such as short- and long-run effects, even from complex dynamic specifications, including models with cointegration. These simulation techniques bring a wider set of inferences from complex models to users in fields as diverse as environmental science (Danish, 2020) and economics (Sharma, 2019). Users can make dynamic inferences from their models, including measures of uncertainty, without the need for any formulae or algebraic solutions. Although other packages may estimate ARDL models (e.g., dynlm (Zeileis, 2019), which only provides model estimates; ardl (Natsiopoulos & Tzeremes, 2020), which uses dynlm for estimation but provides support for the bounds test for cointegration and automated lag selection; dlagM (Demirhan, 2020), which allows for forecasting), dynamac is unique in that it provides post-estimation diagnostics for autocorrelation in the residuals and is designed specifically for providing inferences through counterfactual simulation.

The procedure first estimates the parameters using an ARDL model:

$$\begin{aligned} y_t, \Delta y_t = & \alpha_0 + \delta T + \sum_{p=1}^{P} \phi_p y_{t-p} + \sum_{l_1=0}^{L_1} \theta_{1l} x_{1,t-l_1} + \cdots + \sum_{l_k=0}^{L_k} \theta_{kl} x_{t-l_k} + \\ & \sum_{m=1}^{M} \alpha_m \Delta y_{t-m} + \sum_{q_1=0}^{Q_1} \beta_{1q_1} \Delta x_{1,t-q_1} + \cdots + \sum_{q_k=0}^{Q_k} \beta_{kq_k} \Delta x_{k,t-q_k} + \epsilon_t \end{aligned} \tag{1}$$

The dependent variable (appearing in level form or first-differences) is a function of a constant, a deterministic linear trend T, up to P and L lags of the dependent and independent variables, up to M and Q lags of the first-differenced dependent and independent variables, and error term ϵ_t. Users should use a combination of theory, information criteria, unit-root, cointegration and residual-based tests to arrive at a more restrictive specification of Equation 1.

Next, through `dynardl()`, the estimated parameters are used to draw s simulations from a multivariate normal distribution, with mean $\hat{\beta}$ and variance obtained from the estimated variance-covariance matrix. Stable predicted values for y using the s simulations of $\hat{\beta}$ are then created, using starting values of the independent variables (usually means for continuous variables or modes for categorical ones). Users can choose from either expected values or—by incorporating fundamental model uncertainty—predicted values. Next, at a defined time period t, one of the independent variables is "shocked" by an amount determined by the researcher, and the effect on the dependent variable is observed across the resulting time periods graphically through `dynardl.simulation.plot()`.

This functionality simplifies the production and interpretation of dynamic models to a broader class of users. Basic interpretation of dynamic models usually involves calculating the short-run effect of a one-unit increase in an independent variable, as well as the long-run effect. The former is easy to estimate and observe (represented by a $\hat{\beta}$ coefficient in an ARDL model), but the latter—and its associated confidence intervals—are harder to obtain, since they involve a non-linear combination of parameter estimates. Inferences can also be obscured due to the complexity of the model (like including multiple lags and/or first-differences of dependent and independent variables). For short-run effects, increased model complexity encourages a focus on "boring" effects (the one-unit effect reported by the $\hat{\beta}$ in the regression output); for long-run effects, greater model complexity results in increasingly intractable closed-form estimates of the effects. Stochastic simulations can solve both of these problems by allowing for short-run dynamics beyond the traditional one-unit effect, and also by removing the need for analytic calculations when discussing long-run effects.

Plotting functionality

dynamac implements six new visualizations. These can be presented one at a time using the command `dynardl.simulation.plot()`, or all six using `dynardl.all.plots()`. The resulting plots are:

- The response path of the level of the dependent variable, given a change (a "shock", such as a one standard deviation increase or decrease) in a single independent variable at one point in time.
- Deviations between the predicted and average values of y_t: $\hat{y}_t - \bar{y}$. While this plot follows a response path identical to the plot above, if a shock dissipates and the series reverts to its mean, this is easier to observe if we do not have to subtract the stable starting value from the plot. If the series reverts to a value other than its mean, we would be especially interested in this new long-run mean in response to the shock. This is analogous to an impulse response function.
- The period-to-period response in y_t by differencing the response path. This allows us to visualize successive movements over time.
- The size of the shock itself over time. This allows us to visualize dynamic persistence, so we can make statements like "how many time points until the shock is at 50 percent of its original magnitude?" or "how many time points until the shock is effectively zero?" This information is poorly represented by either the short-run or long-run effects as traditionally calculated.
- The cumulative response in y_t to a shock in x_t. This helps visualize what the "final" effect of a shock is on the dependent variable. A variable might first cause a positive response, but ultimately may be overwhelmed by a negative movement. Unlike other plots, this considers the cumulative histories in each individual simulation when plotting the response in y over time, making the confidence intervals more conservative. This is analogous to the traditional long-run effect of an ARDL model.
- Total movement that occurs in y_t given a change in x_t, which is analogous to the "absolute" effect of a shock to x_t on y_t. In this setup, each of the period-over-period changes sums towards the "total" movement in y_t, so that all of the movement in y is attributed as an effect to the independent variable.

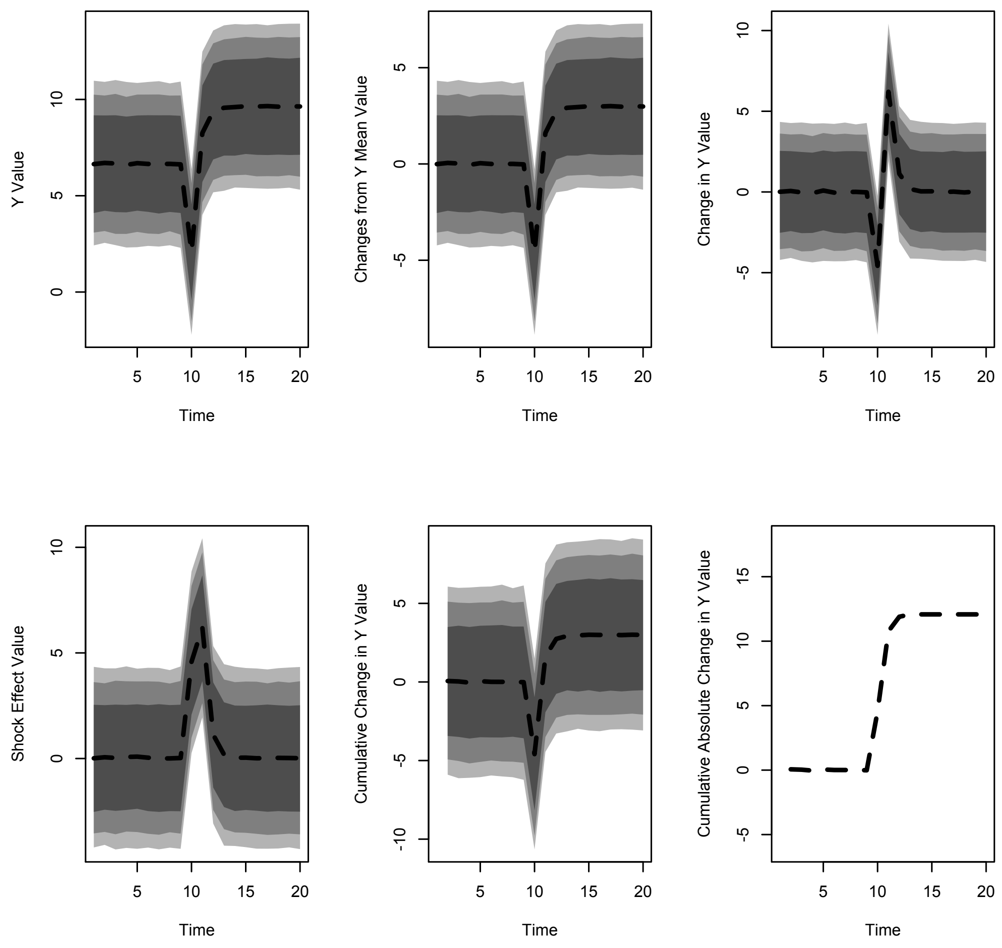

Figure 1: Six quantities of interest from the ARDL equation $\Delta y_t = -0.8y_{t-1} - 2\Delta x_t + x_{t-1} + u_t$.

Figure 1 plots these visualizations from a hypothetical ARDL model. Starting with the top-left plot and moving clockwise, users can easily understand how a dependent variable responds over time to a shock in an independent variable, how it changes away from its average pre-shock value, how the change itself changes over time, the cumulative absolute changes in the dependent variable, the cumulative nature of the changes in the dependent variable, and how the shock decays over time. All six graphical quantities of interest provide a richer set of information about our inferences. These require no additional mathematical burden on the user, regardless of the underlying model complexity.

Conclusion

Dynamic models are complicated since they have effects that do not lend themselves to straightforward interpretation. Stochastic simulations can move our interpretations forward, even in the face of increased model complexity. As we have shown, there are six quantities of interest that are of interest to users, since they provide us with a richer understanding of the dynamic process taking place. dynamac now includes the ability to obtain all of these visualizations described above in a single function: `dynardl.all.plots()`. This will help users expand the different types of quantities of interest they can use to assess statistical and substantive significance.

References

Danish, R. U. (2020). Linking biomass energy and CO2 emissions in China using dynamic autoregressive-distributed lag simulations. *Journal of Cleaner Production*, *250*, 119533. doi:10.1016/j.jclepro.2019.119533

Demirhan, H. (2020). *dLagM: Time series regression models with distributed lag models*. Retrieved from https://CRAN.R-project.org/package=dLagM

Jordan, S., & Philips, A. Q. (2018a). Cointegration testing and dynamic simulations of autoregressive distributed lag models. *The Stata Journal*, *18*(4), 902–923. doi:10.1177/1536867X1801800409

Jordan, S., & Philips, A. Q. (2018b). Dynamic simulation and testing for single-equation cointegrating and stationary autoregressive distributed lag models. *The R Journal*, *10*(2). doi:10.32614/RJ-2018-076

Jordan, S., & Philips, A. Q. (2020). *Dynamac: Dynamic simulation and testing for single-equation ARDL models*. Retrieved from https://CRAN.R-project.org/package=dynamac

Natsiopoulos, K., & Tzeremes, N. (2020). *ARDL: ARDL, ECM and bounds-test for cointegration*. Retrieved from https://CRAN.R-project.org/package=ARDL

Sharma, C. (2019). Testing the asymmetric effects of the economic policy uncertainty on the tourism demand in India. *Tourism Economics*. doi:10.1177/1354816619894080

Zeileis, A. (2019). *Dynlm: Dynamic linear models and time series regression*. Retrieved from https://CRAN.R-project.org/package=dynlm