

## **TAKING DYADS SERIOUSLY**

Abstract. International relations scholarship concerns dyads, yet standard modeling approaches fail to adequately capture the data generating process behind dyadic events and processes. As a result, they suffer from biased coefficients and poorly calibrated standard errors. We show how a regression-based approach, the Additive and Multiplicative Effects (AME) model, can be used to account for the inherent dependencies in dyadic data and glean substantive insights in the interrelations between actors. First, we conduct a simulation to highlight how the model captures dependencies and show that accounting for these processes improves our ability to conduct inference on dyadic data. Second, we compare the AME model to approaches used in three prominent studies from recent international relations scholarship. For each study, we find that compared to AME, the modeling approach used performs notably worse at capturing the data generating process. Further, conventional methods misstate the effect of key variables and the uncertainty in these effects. Finally, AME outperforms standard approaches in terms of out-of-sample fit. In sum, our work shows the consequences of failing to take the dependencies inherent to dyadic data seriously. Most importantly, by better modeling the data generating process underlying political phenomena, the AME framework improves scholars' ability to conduct inferential analyses on dyadic data.

**Word count:** 9,542

---

## Appendix A. AME Monte Carlo Markov Chain Algorithm

Given initial values of  $\{\boldsymbol{\beta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2\}$ , the algorithm proceeds as follows until convergence:<sup>1</sup>

- sample  $\boldsymbol{\theta} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)
- sample  $\boldsymbol{\beta} \mid \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)
- sample  $\mathbf{a}, \mathbf{b} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)
- sample  $\Sigma_{ab} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \rho, \text{ and } \sigma_\epsilon^2$  (Inverse-Wishart)
- update  $\rho$  using a Metropolis-Hastings step with proposal  $\rho^* \mid \rho \sim \text{truncated normal}_{[-1,1]}(\rho, \sigma_\epsilon^2)$
- sample  $\sigma_\epsilon^2 \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \text{ and } \rho$  (Inverse-Gamma)
- For each  $k \in K$ :
  - Sample  $\mathbf{U}_{[k]} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}_{[-k]}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)
  - Sample  $\mathbf{V}_{[k]} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}_{[-k]}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)
  - Sample  $\mathbf{D}_{[k,k]} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\theta}, \mathbf{a}, \mathbf{b}, \mathbf{U}, \mathbf{V}, \Sigma_{ab}, \rho, \text{ and } \sigma_\epsilon^2$  (Normal)<sup>2</sup>

---

<sup>1</sup>Further details on the full conditional distributions and estimation approach underlying AME can be found in Hoff (2021).

<sup>2</sup>Subsequent to estimation,  $\mathbf{D}$  matrix is absorbed into the calculation for  $\mathbf{V}$  as we iterate through  $K$ .

---

## Appendix B. Additional Replication Information

For each of the replications involving a binary dependent variable we provide a table of coefficient estimates that includes the original GLM estimation and the results from the AME model. We have adopted for presentation purposes only the use of significance testing on these observational data for facile comparison with the replicated studies. However, note that we do provide predictive heuristics for these models as well.

**Reiter & Stam (2003).** Additional information for the Reiter & Stam (2003) re-estimation.

Variable	GLM (Logit)	AME
Intercept	-4.784 (0.097)	-3.224 (0.076)
Pers/Democ Directed Dyad	1.026 (0.14)	0.252 (0.066)
Democ/Pers Directed Dyad	0.083 (0.191)	0.114 (0.089)
Personal	0.281 (0.265)	0.212 (0.11)
Military	-0.323 (0.574)	-0.005 (0.234)
Single	-0.677 (0.144)	-0.033 (0.081)
Democracy	-1.073 (0.194)	-0.247 (0.069)
Contiguous	2.912 (0.09)	1.311 (0.038)
Major Power	2.174 (0.101)	0.931 (0.107)
Ally	0.078 (0.086)	0.157 (0.042)
Higher/Lower Power Ratio	-0.316 (0.027)	-0.106 (0.012)
Economically Advanced	-0.175 (0.131)	0.057 (0.055)
Years Since Last Dispute	-0.381 (0.023)	-0.124 (0.008)
Cubic Spline 1	-0.004 (0.000)	-0.001 (0.000)
Cubic Spline 2	0.002 (0.000)	0.001 (0.000)
Cubic Spline 3	-0.001 (0.000)	0.000 (0.000)

**Table B1.** Parameter comparison for Reiter & Stam (2003). Standard errors in parentheses.

**Weeks (2012).** Additional information for the Weeks (2012) re-estimation.

Variable	GLM (Logit)	AME
(Intercept)	-3.784 (0.423)	-2.432 (0.122)
Machine	-0.459 (0.174)	-0.003 (0.041)
Junta	0.515 (0.169)	0.031 (0.045)
Boss	0.649 (0.153)	-0.04 (0.046)
Strongman	0.832 (0.132)	0.04 (0.044)
Other Type	0.147 (0.132)	-0.001 (0.034)
New/Unstable Regime	-0.312 (0.092)	-0.042 (0.033)
Democracy Target	0.185 (0.115)	0.025 (0.026)
Military Capabilities Initiator	5.234 (1.69)	0.044 (0.387)
Military Capabilities Target	6.34 (1.675)	-0.938 (0.477)
Low Trade Dependence	-24.794 (12.866)	-4.872 (3.126)
Both Major Powers	1.136 (0.547)	1.114 (0.232)
Minor/Major	0.772 (0.239)	0.506 (0.112)
Major/Minor	0.711 (0.225)	0.766 (0.153)
Contiguous	2.172 (0.32)	0.709 (0.063)
Log Dist. Between Capitals	-0.209 (0.038)	-0.129 (0.01)
Alliance Similarity Dyad	-0.999 (0.144)	-0.061 (0.062)
Alliance Similarity With System Leader Initiator	0.11 (0.24)	0.08 (0.061)
Alliance Similarity Leader Target	0.203 (0.244)	0.079 (0.057)
Time Since Last Conflict	-0.229 (0.018)	-0.067 (0.006)
Spline1	-0.001 (0.000)	0.000 (0.000)
Spline2	0.000 (0.000)	0.000 (0.000)
Spline3	0.000 (0.000)	0.000 (0.000)

**Table B2.** Parameter comparison for Weeks (2012)). Standard errors in parentheses.

---

**Gibler (2017).** Additional information for the Gibler (2017) re-estimation.

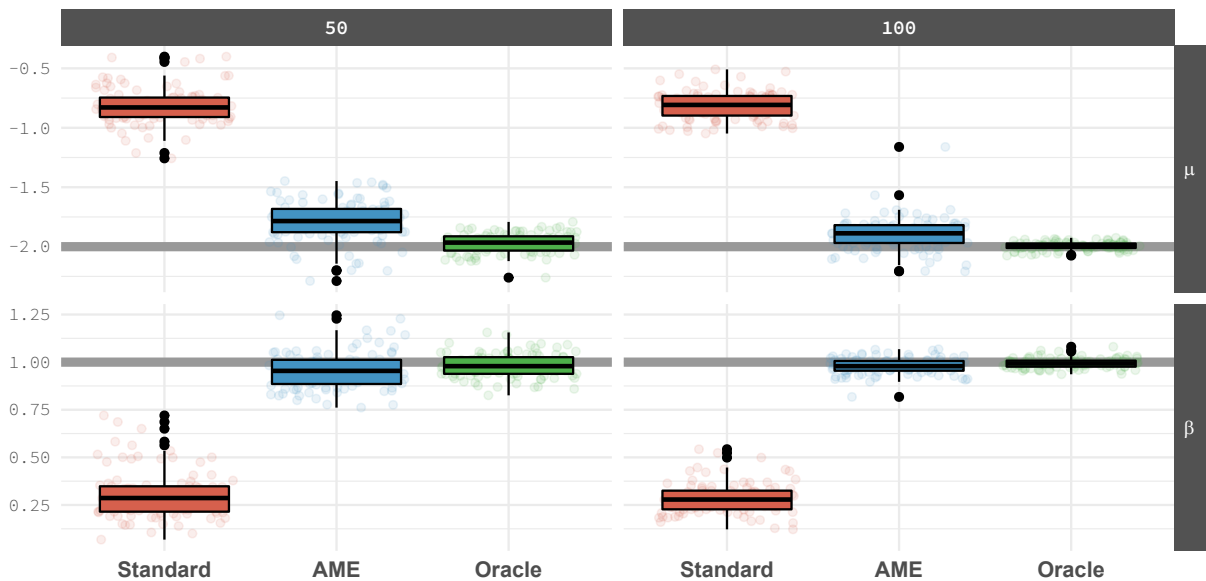
Variable	GLM (Logit)	AME
(Intercept)	-5.826 (0.366)	-4.056 (0.168)
Allied	0.133 (0.102)	0.058 (0.026)
Joint Democracy	-0.527 (0.099)	0.03 (0.023)
Peace Years	-0.261 (0.016)	-0.063 (0.004)
Spline 1	-0.001 (0.000)	0.000 (0.000)
Spline 2	0.000 (0.000)	0.000 (0.000)
Spline 3	0.000 (0.000)	0.000 (0.000)
Contiguity	2.427 (0.196)	0.594 (0.028)
Parity	-0.77 (0.551)	0.006 (0.085)
Parity at Entry Year	2.034 (0.617)	-0.221 (0.084)
Rivalry	2.034 (0.213)	0.679 (0.032)

**Table B3.** Parameter comparison for Gibler (2017). Standard errors in parentheses.

### Appendix C. Additional Simulations: Non-Linear Data

Aronow et al note that “fitting a linear approximation to mildly non-linear data” can be problematic for dyadic random effect models. We agree on the importance of testing this in the context of the simulation that we posed in the paper. Aronow et al incorporate non-linear misspecification by adding in squared version of the key dyadic homophily variable to the DGP for their simulations:  $Y_{ij} = \beta_0 + \beta_1 X_{ij} + \beta_2 X_{ij}^2 + \epsilon_{ij}$ . Here  $X$  is their representation of homophily and is considered observed, while  $X^2$  is their check for robustness to misspecification and is unobserved. We incorporate the same type of misspecification into our probit simulation setup:  $Z_{ij} = \mu + \beta X_{ij} + \gamma W_{ij} + \epsilon_{ij}$ , where  $W$  now instead of being drawn from a separate normal distribution is just the squared version of  $X$ . For the simulation we set values of  $\mu$ ,  $\beta$ , and  $\gamma$  to -2, 1, and 1, respectively. Results are shown below in Figures 1 and 2 below. As in the paper, along the x-axis we implement a standard model that does not take any steps to account for dependencies, AME, and the Oracle model, and as in the paper we find that the AME approach provides a number of benefits in terms of bias and coverage over the standard approach of not dealing with dependencies.

**Figure 1.** Regression parameter estimates for the standard, AME, and oracle models from 1,000 simulations. Summary statistics are presented through a traditional box plot, and the estimates from each simulation are visualized as well as points.



**Figure 2.** Proportion of times the true value fell within the estimated 95% confidence interval for the standard, AME, and oracle models from 1,000 simulations.





---

## Appendix D. Additional Simulations: Correlation with Omitted Variable

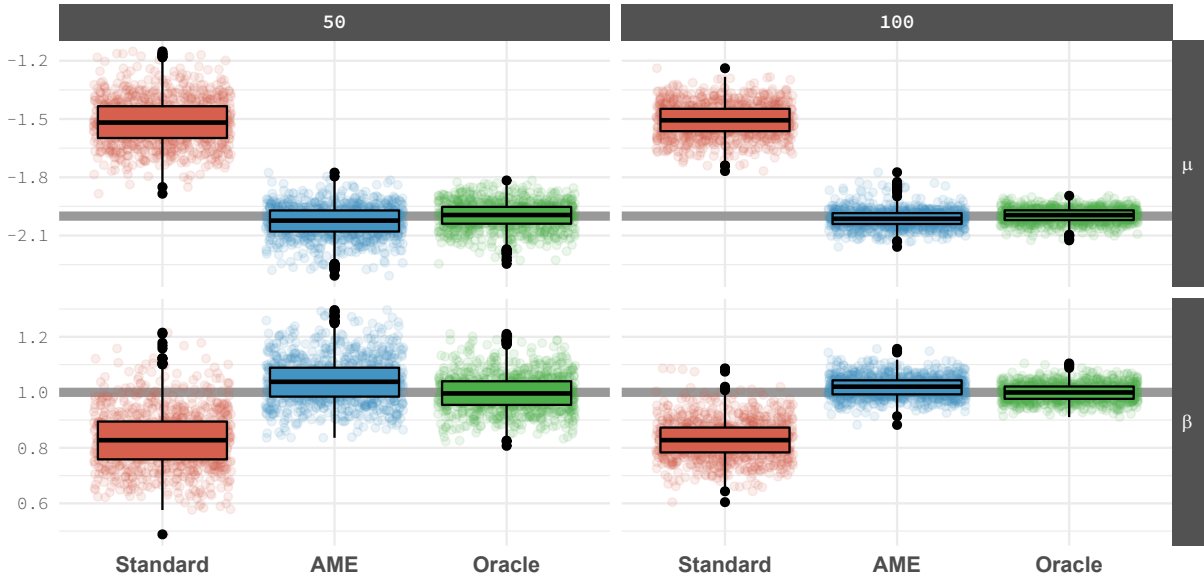
Estimation certainly gets more challenging when the omitted variable is correlated with the observed variable. In the case of normal linear regression, we can show that in general a regression coefficient is unbiased for its “unconditional effect”, which is the direct effect plus something related to the direct effects of any omitted variables that it is correlated with. For example, suppose in our simulation setup  $X_{i,j}$  is correlated with  $W_{i,j}$  via the relation  $W_{i,j} = \alpha X_{i,j} + \epsilon_{i,j}$ . Then:

$$\begin{aligned} Y_{i,j} &= \mu + \beta X_{i,j} + \gamma W_{i,j} + \epsilon_{i,j} \\ &= \mu + (\gamma\alpha + \beta)X_{i,j} + (\gamma\epsilon_{i,j} + \epsilon_{i,j}) \\ &= \mu + \tilde{\beta}X_{i,j} + \tilde{\epsilon}_{i,j} \end{aligned}$$

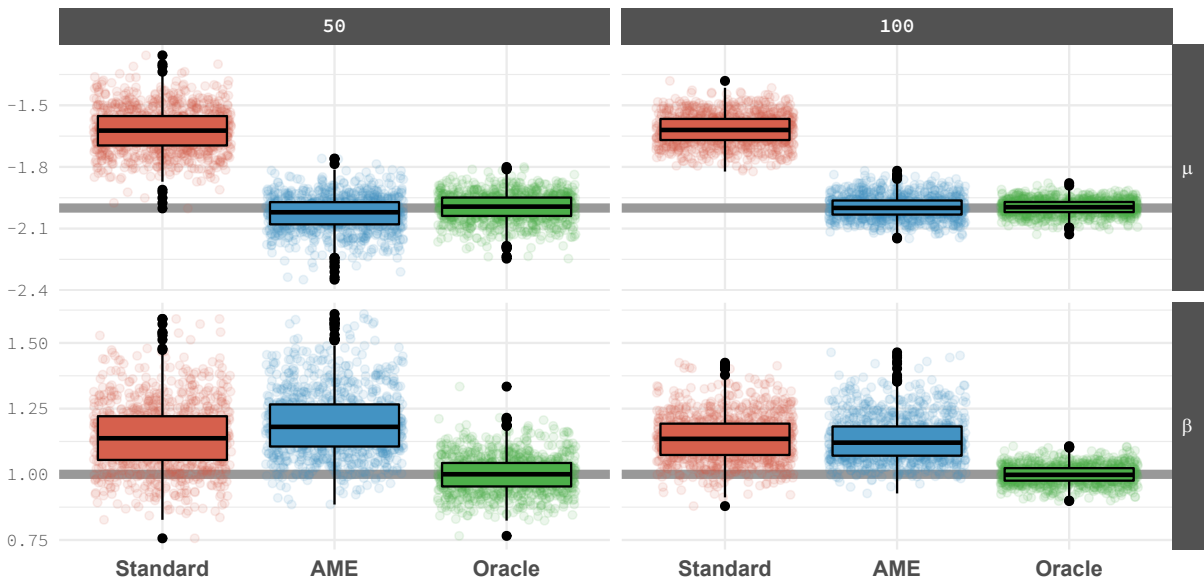
The naive model is in some sense still “correct” as it is an unbiased estimator of the unconditional (direct plus indirect) effect  $\gamma\alpha + \beta$  of  $X$  on  $Y$ . However, there is no way to eliminate this extra bias without making further assumptions about the nature of the omitted variable. Additionally, in the binary case with the probit setup, other sources of bias get introduced as well because of the nonlinearity of the population moments as a function of the parameters. An interesting question is whether or not we can reduce the bias in the network setting by assuming the omitted variable  $w_{i,j}$  is the product of latent node-specific factors. An idea related to this is discussed in Minhas *et al.* (2017).

For now though to show the amount of bias that gets introduced into our setup when  $X$  and  $W$  are correlated we repeat our simulation exercise but induce varying levels of correlation between  $X$  and  $W$ . Results with  $X$  and  $W$  correlated at 0.4 and then 0.7 are shown below, respectively.

**Figure 3.** Regression parameter estimates for the standard, AME, and oracle models from 1,000 simulations. Summary statistics are presented through a traditional box plot, and the estimates from each simulation are visualized as well as points.  $X$  and  $W$  are correlated at 0.4 in this case.



**Figure 4.** Regression parameter estimates for the standard, AME, and oracle models from 1,000 simulations. Summary statistics are presented through a traditional box plot, and the estimates from each simulation are visualized as well as points.  $X$  and  $W$  are correlated at 0.7 in this case.



---

## Appendix E. AME Tutorial

Using the AMEN function requires formatting data into a particular structure. The primary distinction in data formatting is whether the outcome of interest represents a directed or undirected network.

If undirected, the AMEN function has three main inputs:

- **Y**: a  $T$  length **list** of  $n \times n$  adjacency matrices, where  $T$  = number of years in the dataset and  $n$  = number of nodes in the network.
  - An adjacency matrix describes relationships between nodes in a particular year of data. For example, in an adjacency matrix of interstate MIDs, row  $i$  and column  $j$  takes a value of '1' if country  $i$  and country  $j$  had a MID in that year, and '0' otherwise. The diagonal in the adjacency matrix is typically missing. **Y** is a list of these adjacency matrices for the outcome variable, where each element in the list is a different year of data.
- **Xdyad**: a  $T$  length **list** of  $n \times n \times p$  arrays, where  $p$  = number of dyadic covariates in dataset.
  - An array is a data object in R that contains a series of 'stacked' matrices for each year of data. An array of dimension (2, 3, 4), for example, contains 4 rectangular matrices each with 2 rows and 3 columns. Each matrix in the array describes the relationship between nodes with respect to some covariate. For example, for interstate alliances, row  $i$  and column  $j$  takes a value of '1' if country  $i$  and country  $j$  had an alliance in that year, and '0' otherwise. An array contains a matrix of this kind for each covariate going into the model. **Xdyad** is a list of these arrays, where each element in the list is a different year of data.
- **Xrow**: a  $T$  length **list** of  $n \times p$  matrices, where  $p$  = number of monadic (nodal) covariates in dataset.
  - Each matrix in **Xrow** has the nodes running down the rows and the covariates in the dataset running along the columns. An entry in the matrix captures the value that a node takes on for each covariate in a given year. For example, if column  $j$  is GDP per capita and column  $k$  is population, then row  $i$ , column  $j$  measures country  $i$ 's GDP per capita and row  $i$ , column  $k$  measures country  $i$ 's population in a given year. **Xrow** is a list of these matrices, where each element in the list is a different year of data.

---

If directed, AMEN further requires:

- Xrow: a  $T$  length list of  $n \times p$  matrices, where  $p$  = number of sender (nodal) covariates in dataset.
  - Xrow here is nearly identical to the undirected case. The only difference is that rather than each matrix containing covariate data on all nodes, in the directed context each matrix only contains data on the nodes that are acting as senders in that particular year. Using interstate MIDs as an example, if in 1983 only two countries initiated MIDs, Xrow would only contain data for those two countries (i.e., Xrow would only have two rows).
- Xcol: a  $T$  length list of  $n \times p$  matrices, where  $p$  = number of receiver (nodal) covariates in dataset.
  - Xcol here is nearly identical to Xrow, except each matrix contains data on receiver nodes, not sender nodes. Using interstate MIDs as an example, if in 1983 only two countries had MIDs initiated against them, Xcol would only contain data for those two countries (i.e., Xcol would only have two rows).

Beyond the data inputs, the AMEN function requires additional specification:

- model: how to model the outcome variable, e.g., 'logit'
- symmetric: whether the input network is symmetric
- intercept: whether to estimate an intercept
- nscan: number of iterations of the Markov chain
- burn: burn-in period
- odens: thinning interval
- R: dimension of the multiplicative effect (referred to as  $K$  in the paper)
- gof: whether to calculate goodness of fit statistics

There is often little theoretical reason to choose a particular value of R (above). One strategy is to estimate models at different values of R and compare goodness of fit statistics across models. Given the computational time needed for parameter estimates to converge, parallelization strategies are recommended to speed up analysis. In addition, providing AMEN function with starting

---

values, either dictated by theory, previous research, or previous runs can also help speed up convergence time.

The code below presents an example of an AME model running in parallel across 4 different levels of R. Note also that the model is using starting values from a previous run, defined in *startVals*.

```
# running in parallel varying k
imps = 10000 ; brn = 25000 ; ods = 10 ; latDims = 0:3

# Run amen in parallel
library(doParallel) ; library(foreach) ; cl=makeCluster(4) ; registerDoParallel(cl)
foreach(ii = 1:length(latDims), .packages=c("amen")) %dopar% {

  # load previous model run
  load(prevModelFiles[ii])
  # extract start vals
  startValso = ameFit$startVals
  # dump rest
  rm(ameFit)

  ameFit = ame_repL(
    Y=yList , Xdyad=xDyadList , Xrow=NULL , Xcol=NULL ,
    model="bin" , symmetric=FALSE , intercept=TRUE , R=latDims[ii] ,
    nscan=imps , seed=1 , burn=brn , odens=ods ,
    plot=FALSE , print=FALSE , gof=TRUE , startVals=startValso ,
    periodicSave=TRUE )
  save(ameFit , file=pasteo('model_k' , latDims[ii] , '_v2.rda' ) )
}
stopCluster(cl)
```

---

## References

- Hoff, Peter. 2021. Additive and Multiplicative Effects Network Models. *Statistical Science*, **36**(1), 34-50.
- Minhas, Shahryar, Hoff, Peter D., & Ward, Michael D. 2017. Influence Networks in International Relations. *Working Paper*.