Bayesian forecasting of many count-valued time series – Supplementary Material–

Lindsay R. Berry & Mike West
Duke University

March 24, 2019

1 Discount Factor Model Specifications in DGLMs and DCMMs

Appendix A.1 of the paper summarizes the filtering and forecasting analysis of DGLMs. Some comments there in Appendix A.2 concern the use of discount factors to specify evolution variances matrices as well as the new random effects components in DCMMs. More technical and conceptual details are given here.

1.1 Traditional Component Discounting in DGLMs

Specification of evolution variance matrices $\mathbf{W}_t$ in eqn. (2) of the paper uses the standard method based on component discounting (West and Harrison, 1997, chapter 6). In most practical models the state vector is naturally partitioned into components representing different explanatory effects, such as trends (e.g., local level, local gradient), seasonality (time-varying seasonal factors or Fourier coefficients) and effects of independent predictor variables. That is, for some integer J we have $\boldsymbol{\theta}'_t = (\boldsymbol{\theta}'_{t1}, \ldots, \boldsymbol{\theta}'_{tJ})$. It is natural to define $\mathbf{W}_t$ to represent potentially differing degrees of stochastic variation in these components and this is enabled using separate discount factors $\delta_1, \ldots, \delta_J$, where each $\delta_j \in (0, 1]$. A high discount factor implies a low level of stochastic change in the corresponding elements of the state vector, and vice-versa (with $\delta_j = 1$ implying no stochastic noise at all–obviously desirable but rarely practically relevant). The definition of $\mathbf{W}_t$ is as follows.

Referring to Appendix A.1, part 2 of the paper, the time $t-1$ prior variance matrix of $\mathbf{G}_t\boldsymbol{\theta}_{t-1}$ is $\mathbf{P}_t = \mathbf{G}_t\mathbf{C}_{t-1}\mathbf{G}'_t$; this represents information levels about the state vector following the deterministic evolution via $\mathbf{G}_t$ but before the impact of the evolution noise that then simply adds $\mathbf{W}_t$. Write $\mathbf{P}_{tj}$ for the diagonal block of $\mathbf{P}_t$ corresponding to state subvector $\boldsymbol{\theta}_{tj}$ and set

$$\mathbf{W}_t = \text{block diag}[\mathbf{P}_{t1}(1-\delta_1)/\delta_1, \ldots, \mathbf{P}_{tJ}(1-\delta_J)/\delta_J].$$

The implied prior variance matrix of $\boldsymbol{\theta}_t$ following the evolution has corresponding diagonal blocks $\mathbf{R}_{tj} = \mathbf{P}_{tj}/\delta_j$ but maintains the off-diagonal blocks from $\mathbf{P}_t$. Thus, the stochastic evolution increases uncertainties about state vector elements in each subvector j by $100(1-\delta_j)/\delta_j\%$, maintains the correlations in $\mathbf{P}_{tj}$ for state elements within the subvector j, while reduces cross-correlations between elements in differing subvectors. In practice, high values of the δ_j are desirable and typical applications use values in the range 0.97–0.99 with, generally, robustness in terms of forecasting performance with respect to values in the range. Evaluation of forecast metrics on training data using different choices of discount factors is a basic strategy in model building and tuning.

1.2 Discount Factor Specifications for Random Effects Extensions of DGLMs

Consider now the random effects extension of DGLMs in Section 2.3 of the paper. The DCMM uses this for the shifted Poisson case for non-zero count series. As detailed in that section, this is enabled by extension of the state vector to include time-specific zero mean elements defining these random effects. Practically, this is defined using a random effects discount factor $\rho \in (0, 1]$ whose net impact on the DGLM analysis summarized in Appendix A.1 of the paper is simply to inflate the prior variance of the linear predictor $\lambda_t = \log(\mu_t)$. That is, in Appendix A.1, part 4 of the paper, the variance $q_t = \text{V}[\lambda_t \mid \mathcal{D}_{t-1}, \mathcal{I}_{t-1}]$ is modified to $q_t + v_t$ where $v_t = q_t(1-\rho)/\rho$, resulting in q_t/ρ.

As with the standard discounting on state vectors above, evaluation of forecast metrics on training data using different choices of ρ is a basic strategy for choosing values, and these will be specific to each time series.

More theoretical insight can be gained by considering the impact on the implied $1-$step forecast distributions. In the standard DGLM with no random effects, recall that the conditional prior for the Poisson mean μ_t is $Ga(\alpha_t, \beta_t)$ where the parameters are chosen to be consistent with the prior mean f_t and variance q_t of $\lambda_t = \log(\mu_t)$, namely $f_t = \psi(\alpha_t) - \log(\beta_t)$ and $q_t = \psi'(\alpha_t)$. Suppose we have a relatively precise prior– with q_t modestly high– so that the approximations $\psi(\alpha_t) \approx \log(\alpha_t)$ and $\psi'(\alpha_t) \approx 1/\alpha_t$ are valid (these are in fact very accurate approximations in many applications). Then $f_t \approx \log(\alpha_t/\beta_t)$ and $q_t \approx 1/\alpha_t$, resulting in $\alpha_t = 1/q_t$ and $\beta_t = \alpha_t \exp(-f_t)$. The implied $1-$step forecast distribution for y_t is negative binomial with mean $\alpha_t/\beta_t = \exp(f_t)$ and variance $\alpha_t/\beta_t + \alpha_t/\beta_t^2 = \exp(f_t)(1 + \exp(f_t)q_t)$.

Now consider the impact of the random effects extension. The practical impact of ρ is that q_t is inflated to q_t/ρ. The resulting negative binomial forecast distribution then has the same mean $\exp(f_t)$– not impacted at all by ρ– but now has variance $\exp(f_t)(1+\exp(f_t)q_t/\rho)$. This has the same base component $\exp(f_t)$ (the "Poisson" component) but the second term (considered the "extra-Poisson" variation in the negative binomial) increases by a factor of $1/\rho$. Note that the impact of the random effects extension is then to increase forecast variances more at higher levels of the series (higher f_t), consistent with the aim of improving forecasts for infrequent higher events.

2 Additional Forecast Comparisons

This section adds to the forecast evaluations and comparisons made in Section 5.3 of the paper. That study of forecasts of sales on four selected items included comparisons using the sMSE metric as detailed in Section 5.2 of the paper. The sMSE results are summarized here, to add to those for MAD and MRPS in Section 5.3 of the paper.

Comparisons under sMSE: Figure 1 shows the sMSE versus forecast horizon for each item from the best performing multi-scale and benchmark DCMMs. For each item, the multi-scale DCMM has lower sMSE across all of the forecast horizons. Comments by specific items are as follows.

A: The decreases in sMSE are similar across the forecast horizons for the multi-scale versus benchmark DCMM. Averaging across the forecast horizons, the overall percentage decrease in sMSE of the multi-scale DCMM versus the benchmark DCMM is 1.4%.

B: The decreases in sMSE are largest when forecasting 1–7 days ahead. When forecasting 1–3 days ahead, the percentage decrease in sMSE of the multi-scale DCMM versus the benchmark DCMM is greater than 3%. Averaged across all of the forecast horizons, the percentage decrease in sMSE is 1.8%.

C: The decreases in sMSE are similar across the forecast horizons. Across the forecast horizons, the average percentage decrease in sMSE of the multi-scale versus benchmark DCMM is 6.3%.

D: The decreases in sMSE are consistent across the forecast horizons. Compared to the benchmark DCMM, the multi-scale DCMM has an average 10.3% decrease in sMSE across the forecast horizons.

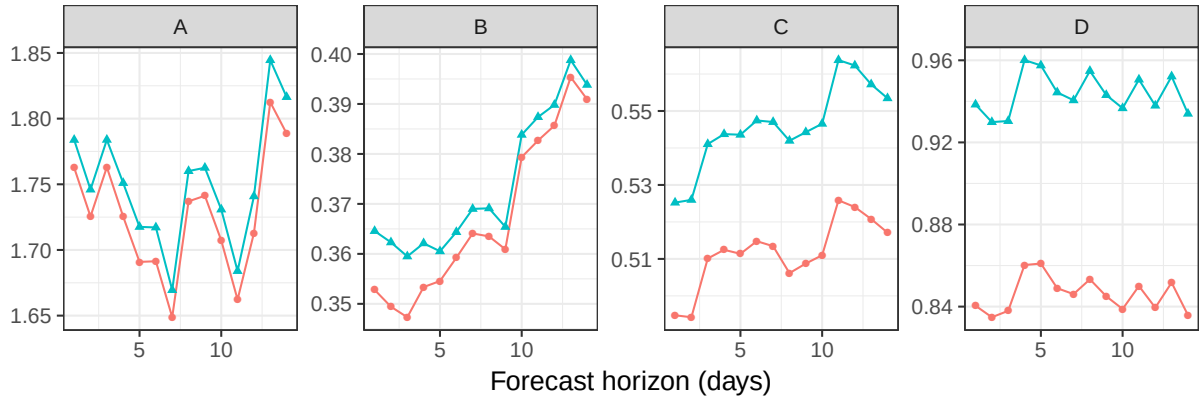

Figure 1: Scaled mean squared error (sMSE) vs forecast horizon (days) for items A–D from the multi-scale (orange circles) and benchmark (blue triangles) models.

3 Detailed Comparisons of DCMMs with Existing Models

3.1 Overview

This section provides detailed discussion of extensive comparisons noted and summarized under **Comparisons with Alternative Models** at the end of Section 5.3 of the paper.

The DCMM is compared with several existing models that could be used in the context of product demand forecasting. For each comparison method, we briefly describe the model, detail the availability of code/packages, and (if possible) summarize the performance of each model on items in our dataset. Given our focus on forecasting low-valued count data, we limit the scope of models considered here to those which produce coherent forecasts over the non-negative integers. To compare forecast results, we present various point forecast metrics as well as probabilistic forecast evaluation.

Each of the models considered here fall in the framework of observation-driven models with some flavor of autoregressive dependence structure between observations. Before describing model specific details, we detail a few key differences between DCMMs and the following methods:

- All of the following models are univariate, and we have proposed an *efficient* multi-scale framework for incorporating cross series dependence.
- In the following models, the ARMA, regression, and seasonal components are static while the DCMM can incorporate dynamic versions of these model components.
- Some of the models mentioned below account for over-dispersion– typically through an extension from a Poisson to Negative Binomial distribution. However, in our applied framework with potentially millions of individual series, it is not realistic to individually specify the appropriate model for each individual item. Additionally, the appropriate model may change over time if the demand for an item changes (e.g. seasonal products). The DCMM random effects extension we have proposed can *automatically* account for overdispersion in time series of counts.
- In the following models, there is no way to account for data with more/fewer zeros than expected under Poisson/Negative Binomial models. In the DCMM, the binary component is automatic and flexible enough to model any time series of counts (with fewer/excess zeros) without the need for individual item model customization.

3.2 INAR Model

Details of the integer valued autoregressive (INAR) model are given in Al-Osh and Alzaid (1987) and McKenzie (1988). An alternate name for the same model is the Poisson Autoregressive (PAR) model.

Model details. Let y_t denote a non-negative count valued time series observed over time $t = 1, \ldots, T$. The INAR model is defined in the following manner,

$$y_t = \alpha \circ y_{t-1} + \epsilon_t$$

where y_0 has a Poisson distribution with mean λ_0, and $\epsilon_t \sim Po(\lambda)$, with $\epsilon_t \perp y_k$ for all t, k. The binomial thinning operator $\circ$ is defined as follows: given X_{t-1}, $\alpha X_{t-1} = \sum_{i=1}^{X_{t-1}} B_{it}$, where $B_{1t}, \ldots, B_{X_{t-1},t}$ are iid Bernoulli random variables with success probability α.

Our comments. The intuition underlying the INAR model is that the count at time t is the sum of "survivors" from time $t-1$ and new arrivals between time $t-1$ and t. The survivors are represented by the binomial thinning, $\alpha \circ y_{t-1}$, and the new arrivals are represented by ϵ_t, generally a Poisson random variable. This simple intuition is appealing in many applications, and could be relevant when modeling the number of customers waiting in line at a store. However, when modeling the daily sales of a supermarket product, the birth-death understanding does not seem to be a natural way of understanding the data.

Additionally, it is not trivial to extend this model to incorporate covariate effects. Incorporating item level price, promotions, and holiday effects is a very important part of product sales forecasting. Finally, it is unclear how this model would incorporate negative dependence between observations. Both of these tasks are possible in the DCMM framework.

We spent many hours searching, but we were unable to find any code/packages to implement the INAR model. It is possible for us to code up the estimation procedure to produce MLE estimates of the model parameters ourselves. Given these MLEs, we could write code to produce $k-$step mean forecasts for the INAR model. However, given that our time series are low-valued counts, the mean forecast is not especially interesting, and our main interest is in entire forecast distributions. It is unclear if $k-$step predictive distributions are available in this model, and would be a separate research project to implement this ourselves. Given these experiences, we have focused on other comparison models which have similarities to the INAR model in that they are observation-driven and have an autoregressive dependence structure.

3.3 GLARMA model

The Generalized Linear ARMA model (GLARMA) is described in Dunsmuir (2015) and Dunsmuir and Scott (2015).

Model details. Let y_t denote a non-negative count valued time series observed over time $t = 1, \ldots, T$. Here we detail the Poisson GLARMA model, however there is also a Negative Binomial GLARMA for modeling overdispersed counts.

$$y_t \mid \mathcal{D}_{t-1} \sim Po(\mu_t) \tag{3.1}$$

$$\log(\mu_t) = W_t = \mathbf{x}_t'\boldsymbol{\beta} + Z_t \tag{3.2}$$

$$Z_t = \sum_{i=1}^{p} \phi_i(Z_{t-i} + e_{t-i}) + \sum_{i=1}^{q} \theta_i e_{t-i} \tag{3.3}$$

where

$$e_t = \frac{Y_t - \mu_t}{\nu_t}$$

and ν_t is some scaling factor. For the models considered here, we set e_t to be Pearson residuals with $\nu_t = \sqrt{\mu_t}$.

Our comments. The GLARMA model is implemented in the `glarma` R package. This package allows incorporation of static regression effects and static ARMA dependence structure. Multi-step forecasting is available through a simulation based approach– allowing access to the full predictive distribution k-steps ahead. Unlike the DCMM, the GLARMA model does not account for excess zeros. To account for overdispersion, this package allows both conditionally Poisson and Negative Binomial response distributions. Another important detail is that the GLARMA package does not handle time series with missing data which is straightforward in the DCMM framework.

To compare with the DCMM, we implement the GLARMA models with both Poisson and Negative Binomial response distributions. Each model includes the log price as a predictor and six dummy variables for the day of the week to incorporate the weekly seasonal effect. We also include a lag-1 autoregressive structure to incorporate time dependence in the model.

As in our paper, we fit the model on the first 100 days of data, and then began forecasting $1-14$ days ahead on each day during the next two years. In the GLARMA framework, this requires re-estimating the model on all of the observed data at each time t, and then forecasting over the next 1-14 days using the built in GLARMA simulation method. We run this analysis across the same 4 items (A–D) in the multiscale application in the paper. See Figure 2 for plots of the daily sales of items A–D over this time period. Figure 3 displays the MAD of forecasts under each model for items A–D. Figures 4 and 5 display probabilistic forecast metrics for the GLARMA models. We now discuss the specifics of the forecasting results for each item.

Item A. Here we only present results for the Poisson GLARMA model since the Negative Binomial GLARMA model encountered an error during estimation for item A. Figure 3 (top, left) shows the MAD over this two-year period versus the forecast horizon. The multi-scale DCMM has the lowest MAD for 12 of the 14 forecast horizons; however, the MAD of all three models are fairly similar. This similarity is most likely due to the fact that this low-selling item will have a median of zero or one sales on most days, so the average absolute deviation of the median will be very small on average.

The rPIT plot and coverage plots for the Poisson GLARMA model are shown in the top row of Figure 4. In the rPIT plot, there is a slight S-shape to the rPIT values, indicating that the forecast distribution is slightly too narrow.

Figure 6 shows binary calibration plots for $1-$day forecasts of non-zero sales of item A in the Poisson GLARMA model. Well calibrated binary forecasts means that the crosses or red bars should fall within the shaded gray region. For item A, we see that there are three bars that all fall below the gray shaded region. Furthermore, although the other bars fall within the gray regions, we do

notice a trend of the crosses falling under the diagonal line. This plot indicates that the Poisson GLARMA model overpredicted the probability of zero sales occurring.

Item B. Item B is a relatively high-selling item. Figure 3 (top, right) displays the MAD for each of the four models. We see that the DCMM models both have lower MAD than the GLARMA models. The decrease in MAD of the multi-scale DCMM versus the Negative Binomial GLARMA model ranges from 0.11 to 0.20. Over this two year period, this corresponds to between 86 and 144 units more accurate when using the multi-scale DCMM.

The rPIT plot and coverage plots for the Poisson GLARMA model are shown in the bottom row of Figure 4. In the rPIT plot, we see that the Poisson rPIT values deviate from uniformity. This indicates that the forecast distribution is too narrow in the Poisson GLARMA model. The Negative Binomial rPIT values appear to conform to uniformity. In the Poisson coverage plot, we see that there is undercoverage of forecast intervals. Overall, it appears that the Poisson GLARMA model does not sufficiently account for the overdispersion apparent in the data. In the Negative Binomial coverage plot, we see that the coverage lies along the 45-degree line. The Negative Binomial GLARMA model appears to perform well in terms of probabilistic forecasting.

Item C. The multi-scale DCMM has the lowest MAD for 13 of the 14 forecast horizons. When forecasting 14-days ahead, the multi-scale DCMM and Negative Binomial GLARMA model actually have the exact same MAD. The GLARMA models have lower MAD than the univariate DCMM across the forecast horizon. The largest differences in MAD occur for short term forecasting. For 1, 2, and $5-$step forecasting, the differences in MAD correspond to between 25 and 30 units more accurate for the multi-scale DCMM versus the Negative Binomial GLARMA model.

The rPIT plot and coverage plots for the Poisson GLARMA model are shown in the top row of Figure 5. In the rPIT plot, the Poisson rPIT values deviate from uniformity slightly. Based on the rPIT values, it appears that the predictive distribution is too light on the lower end, and slightly too narrow on the higher end. The Negative Binomial rPIT values are closer to uniformity, however there is a slight deviation on the upper end. This may indicate that the upper tail of the predictive distribution is too long.

The Poisson coverage plot shows very slight undercoverage for intervals above 80%. The Negative Binomial coverage plot shows slight overcoverage for the forecast intervals.

Item D. The multi-scale DCMM has the lowest MAD for all of the forecast horizons. However, the GLARMA models have lower MAD than the univariate DCMM. The differences in MAD are consistent across forecast horizons. Across the two-year period, this corresponds with between 46 and 64 units of accuracy gained by using the multi-scale DCMM versus the Negative Binomial GLARMA model. Similarly, this corresponds with between 23 and 43 units of accuracy gained by using the multi-scale DCMM versus the Poisson GLARMA model.

The rPIT plot and coverage plots for the Poisson GLARMA model are shown in the bottom row of Figure 5. In the rPIT plot, we see that the Poisson rPIT values slightly deviate from uniformity. These values indicate that the forecast distribution in the Poisson GLARMA model is too narrow on

the upper and lower tail. The Negative Binomial rPIT values are closer to uniformity, but do not conform exactly to the $45-$degree line.

The Poisson coverage plot shows very slight undercoverage. The Negative Binomial coverage plot is very close to the $45-$degree line.

Item E. The daily sales of item E are shown in Figure 7. We attempted to run the forecasting analysis on this item for the Poisson and Negative Binomial GLARMA models. However, both of these models encountered errors during the time period we were evaluating. This item has a high probability of zero sales, but also frequent bursts of large sales. It appears that the GLARMA models are not appropriate for this time series.

Discussion of results. Overall, the performance of the GLARMA models is good for the items discussed here. The performance of the GLARMA models under MAD is quite competitive with the DCMMs for items A, C, and D. However, both of the DCMMs outperform the GLARMA models for item B. We believe one of the underlying reasons for this is that the daily sales of item B are more nonstationary and more variable over time than the other three items. This theory is supported by the fact that the GLARMA models encountered errors for item E, which displays quite non-stationary demand.

For probabilistic forecasting, the Poisson GLARMA model consistently produces forecast distributions that are just too narrow. The performance of the Negative Binomial GLARMA model was often improved.

One important consideration for the GLARMA models is that there is no separate model for zero sales like in the DCMM. One result of this is that the binary forecasting of zero-versus-non-zero sales appears to be miscalibrated for item A.

While the GLARMA models performed well in these comparisons, we believe that the multi-scale DCMM is the more appropriate choice for our application to product sales forecasting. The main reason for this is that the DCMM is more automatically flexible and robust to handling the common characteristics of time series of counts. The DCMM can automatically handle time series with excess zeros, overdispersion, and non-stationary model components. The GLARMA models require individual item customization by fitting either a Poisson or Negative Binomial model, do not have a separate binary component, and appear to struggle with time series with apparent non-stationarities. Another key issue with the GLARMA framework is that it cannot be used to model and forecast for every item. This would require further customization for individual items which is not realistic in our context.

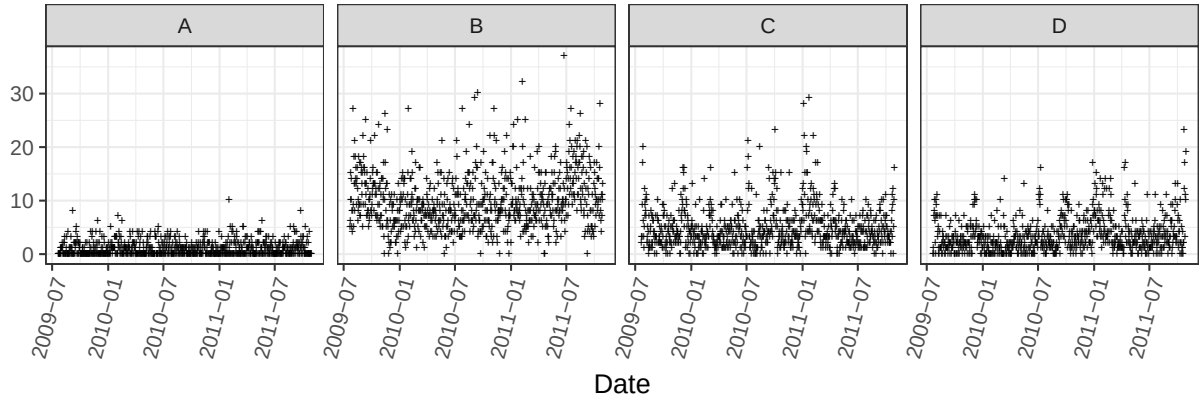

Figure 2: Daily unit sales (in counts per day) of four spaghetti items A–D in one store from July 22nd 2009 to October 29th 2011.

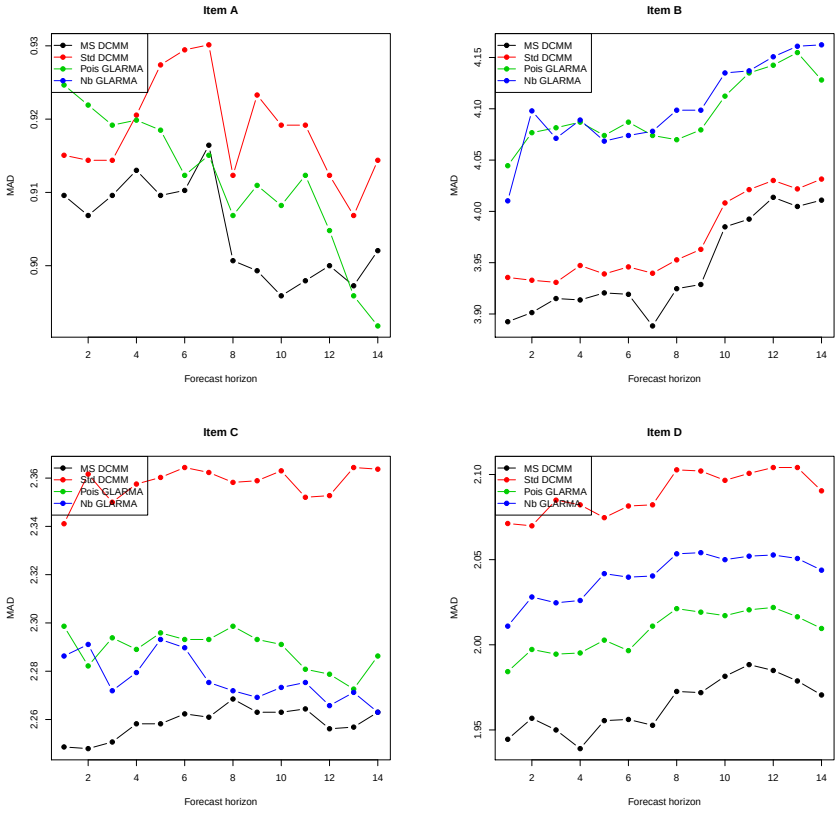

Figure 3: Forecast comparison in terms of MAD of items A–D of multi-scale (black) and univariate (red) DCMMs to the Poisson (green) and Negative Binomial (blue) GLARMAs.

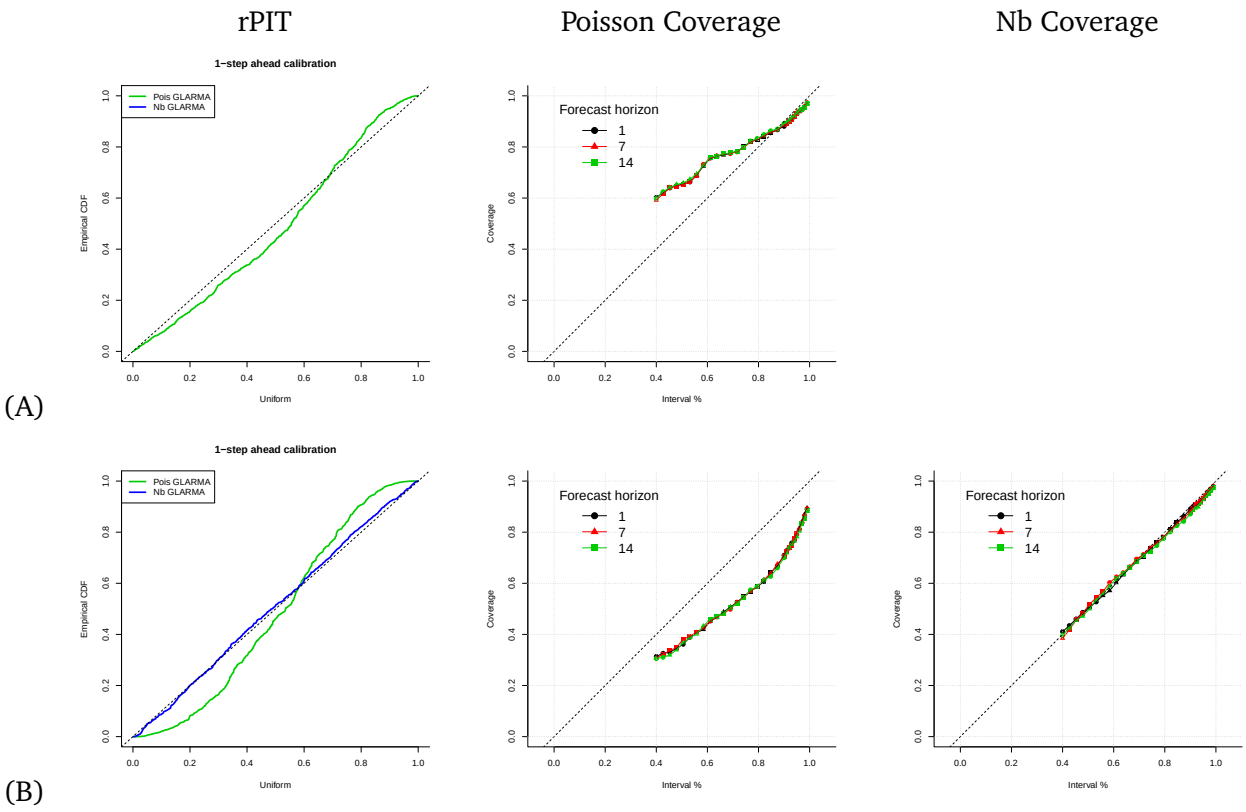

Figure 4: Probabilistic forecast evaluations for items A (top row) and B (bottom row). The left column shows the rPIT plots for the Poisson (green) and Negative Binomial (blue) GLARMA models. The middle column shows the coverage of HPD intervals for the Poisson GLARMA model. The right column shows the coverage of HPD intervals for the Negative Binomial GLARMA model.

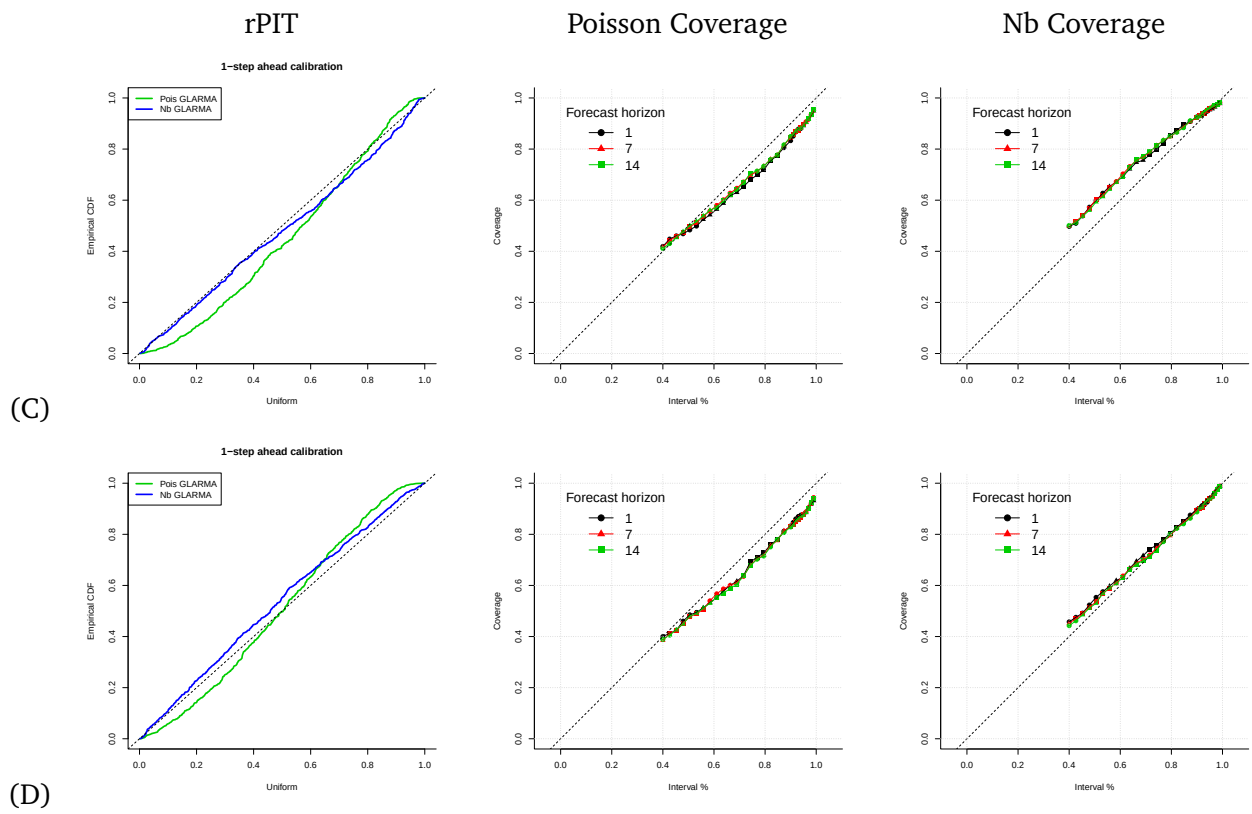

Figure 5: Probabilistic forecast evaluations for items C (top row) and D (bottom row), with details as in Figure 4.

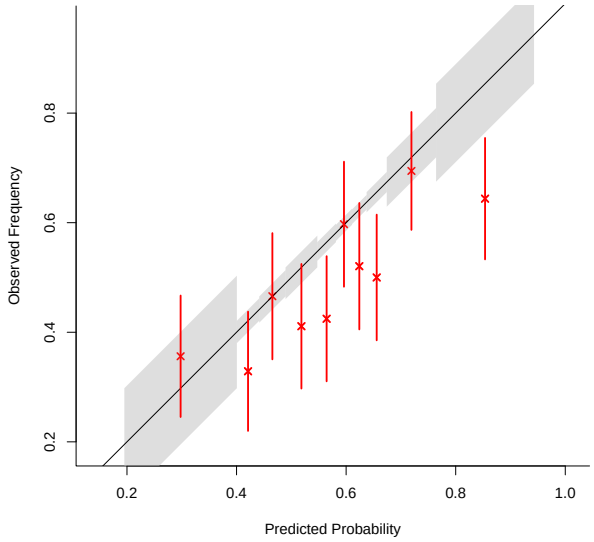

Figure 6: Binary calibration plots from 1-day ahead forecasting of non-zero sales of item A in the Poisson GLARMA model. Crosses mark observed frequencies in each bin, horizontal grey shading indicates variation of forecasts in each bin, and vertical bars indicate binomial variation based on the number of days in each bin.

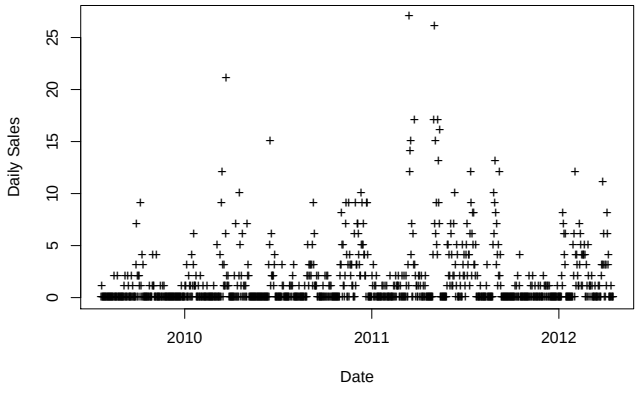

Figure 7: Daily sales of item E from mid 2009 to early 2012.

3.4 ACP or INGARCH Model

An alternate name for this model is the integer-valued GARCH model (INGARCH) of order p and q. Details about these models are available in Heinen (2003), Ferland et al. (2006), and Fokianos et al. (2009).

Model details. Let y_t denote a non-negative count valued time series observed over time $t = 1, \ldots, T$. The ACP model has counts follow a Poisson distribution with an autoregressive mean.

$$y_t \mid \mathcal{D}_{t-1} \sim Po(\mu_t) \tag{3.4}$$

$$E[y_t \mid \mathcal{D}_{t-1}] = \mu_t = \omega + \sum_{j=1}^{p} \alpha_j y_{t-j} + \sum_{j=1}^{q} \beta_j \mu_{t-j} \tag{3.5}$$

for positive $\alpha_j, \beta_j, \omega$. Here, $\mathcal{D}_{t-1} = \{y_1, \ldots, y_{t-1}\}$. In this model, the order p describes the number of lagged observations that affect the mean at time t. The order q represents the number of lagged values of Poisson mean that appear in the model. Since the Poisson mean μ_t is positive, the values of $\alpha_j, \beta_j, \omega$ are constrained to positive values. The most commonly used form of this model is the ACP(1,1).

Our comments. The ACP(p,q) model is implemented in the `acp` R package. Built-in functions in this package estimate the ACP(p,q) model with covariates, and provide static forecast means. In the ACP(p,q) model, time dependence is achieved through the latent ARMA structure, but the ARMA coefficients and the covariate effects are static in time. This model accounts for overdispersion through the autoregressive Poisson mean. There is no simple way to account for excess/fewer zeroes, and there is no multivariate extension implemented. To apply this model in an on-line setting, we refit the model to all of the observed data at each time $t = 1, \ldots, T$.

For this comparison with the DCMM, we attempt to implement the ACP(1,0) and ACP(1,1) models. Each time we tried to run the ACP(1,0) model, the code returned an error– it appears that this model form is not supported by the current software. We focused our comparison on the ACP(1,1) model instead.

The built-in function `predict.acp` produces a static forecast over the next k days. At time t, when forecasting time $t+k$, this function will use the parameter estimates at time t, and requires the observed data $y_{t+1}, \ldots, y_{t+k-1}$. In our application, we are interested in multi-step ahead predictive distributions. In order to compare results with the DCMM, we have written our own function to produce simulations from the k-step joint forecast distribution. After fitting the ACP(1,1) model on data from time 1 to t, we use the `predict.acp` function to forecast the $1-$step forecast mean for time $t+1$. Conditioning on the mean, we simulate a Poisson valued prediction y^*_{t+1} from the implied forecast distribution for y_{t+1}. Treating this simulated y^*_{t+1} as synthetic data, we use the `predict.acp` function to forecast the mean at time $t+2$, and then draw a value of y^*_{t+2}. We repeat this procedure up to time $t+14$ to produce a single joint draw form the $1{:}14$ step forecast distribution.

After developing this forecasting procedure, we refit the ACP(1,1) model at each day in the two year time period of interest. On each day, we use the described procedure to forecast k-steps ahead.

However, for each of the four items A–D in the paper, the ACP package encountered an error at some point during this two year period.

Another R package `tscount` implements the INGARCH model. We were able to implement the INGARCH model for some items with this package. The built-in function `predict.tsglm` produces $1{:}k$ step forecast means, medians, and prediction intervals. It is only possible to extract prediction intervals rather than the entire forecast distributions. Here, we focus on the coverage of 95% prediction intervals.

Item A. The MAD for the INGARCH model for item A is shown in Figure 8. The multi-scale DCMM has lower MAD across all forecast horizons. The decrease in MAD varies between 0.06 and 0.10. Across the two-year period, these decreases correspond with between 48 and 70 units of accuracy gained under the multi-scale DCMM.

The 95% prediction intervals for each forecast horizon show slight overcoverage of the observed data. Across the forecast horizons, the 95% intervals contain over 98% of the observed data.

Item B. The MAD for item B is shown in Figure 8. The multi-scale DCMM has lower MAD across all 14 forecast horizons. The decrease in MAD varies from 0.11 to 0.20 for the multi-scale DCMM versus the INGARCH model. Across the two-year period, these decreases in MAD corresponded to between 84-119 units more accurate under the multi-scale DCMM.

The 95% prediction intervals at each forecast horizon showed undercoverage of the observed data. The empirical coverage across the forecast horizon varied from 79:81%.

Item C. The MAD results for item C are shown in Figure 8. The multi-scale DCMM has lower MAD across all 14 forecast horizons. The decrease in MAD ranges from 0.37 to 0.52 for the multi-scale DCMM versus the INGARCH model. Across the two-year period, these decreases correspond to between 271 to 382 units of accuracy gained by using the multi-scale DCMM versus the INGARCH model.

The 95% prediction intervals at each forecast horizon are very close to the nominal coverage. Empirical coverage varies from 95% to 98% across the forecast horizons.

Item D. The MAD results for item D are shown in Figure 8. The multi-scale DCMM has lower MAD across all 14 forecast horizons. The decrease in MAD ranges from 0.28 to 0.40 for the multi-scale DCMM versus the INGARCH model. Across the two-year period, these decreases correspond to between 206 and 295 units of accuracy gained by using the multi-scale DCMM versus the INGARCH model.

The 95% prediction intervals at each forecast horizon are very close to the nominal coverage.

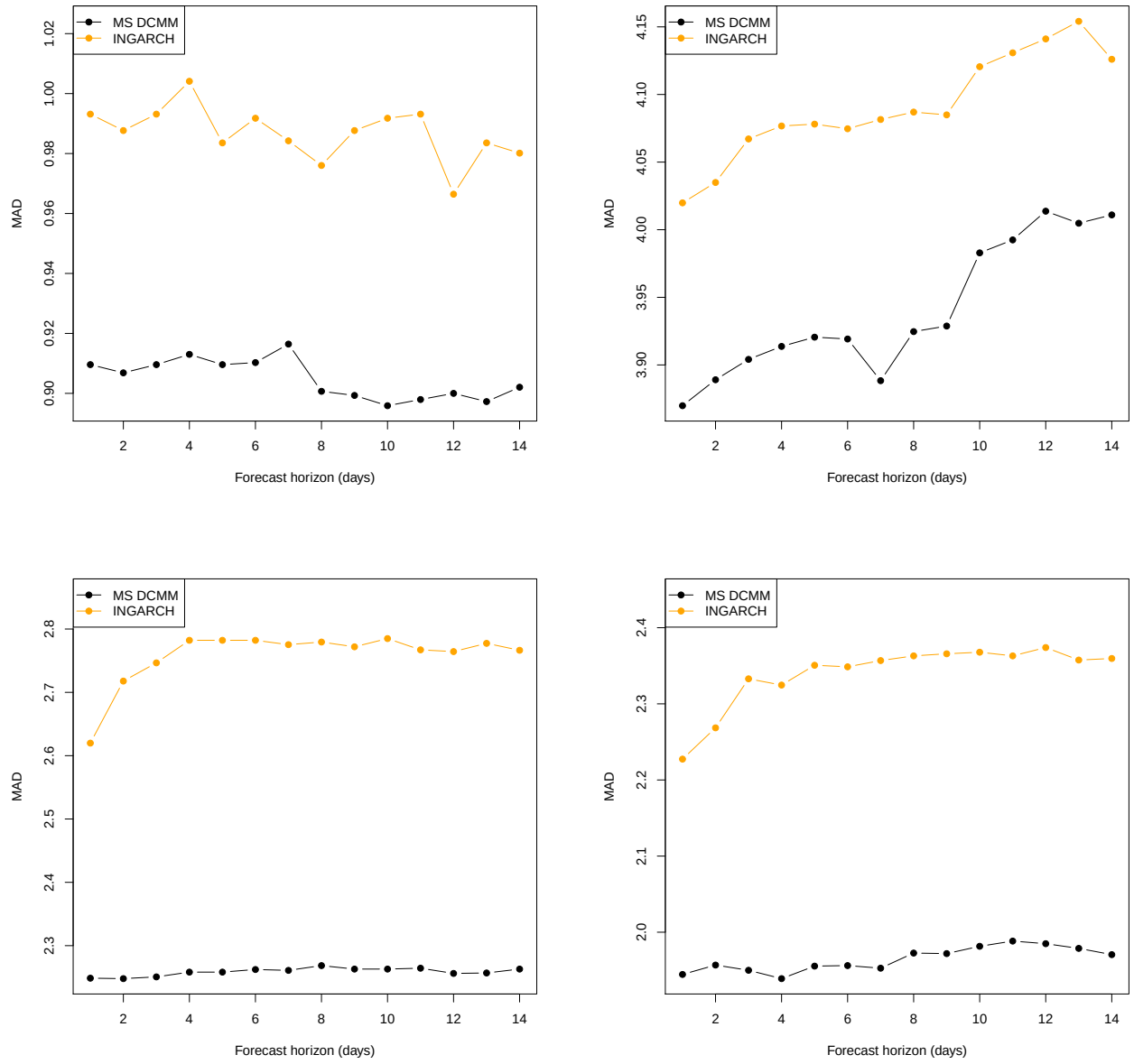

Figure 8: Forecast comparison in terms of MAD for item A,B,C,D for the multi-scale DCMM versus the INGARCH(1,0) model.

References

Al-Osh, M. A. and A. A. Alzaid (1987). First-order integer valued autoregressive (INAR(1)) process. *Journal of Time Series Analysis 8*, 261–275. 4

Dunsmuir, W. T. M. (2015). Generalized linear autoregressive moving average models. In R. Davis, S. Holan, R. Lund, and N. Ravishanker (Eds.), *Handbook of Discrete-Valued Time Series*, Chapter 3, pp. 51–57. CRC Monographs. 5

Dunsmuir, W. T. M. and D. J. Scott (2015). The glarma package for observation-driven time series regression of counts. *Journal of Statistical Software 67*. 5

Ferland, R., A. Latour, and D. Oraichi (2006). Integer-valued GARCH process. *Journal of Time Series Analysis 27*, 923–942. 13

Fokianos, K., A. Rahbek, and D. Tjøstheim (2009). Poisson autoregression. *Journal of the American Statistical Association 104*, 1430–1439. 13

Heinen, A. (2003). Modelling time series count data: An autoregressive conditional Poisson model. *Munich Personal RePEc Archive 8113*. Electronic publication. 13

McKenzie, E. (1988). Some ARMA models for dependent sequences of Poisson coutns. *Advances in Applied Probability 20*, 822–835. 4

West, M. and J. Harrison (1997). *Bayesian Forecasting and Dynamic Models* (2nd ed.). Springer-Verlag. 1