



The **glarma** Package for Observation-Driven Time Series Regression of Counts

William T. M. Dunsmuir
University of New South Wales

David J. Scott
University of Auckland

Abstract

We review the theory and application of generalized linear autoregressive moving average observation-driven models for time series of counts with explanatory variables and describe the estimation of these models using the R package **glarma**. Forecasting, diagnostic and graphical methods are also illustrated by several examples.

Keywords: observation-driven count time series, generalized linear ARMA models, **glarma**, R.

1. Introduction

Increasingly, in fields as diverse as financial econometrics, public policy assessment, environmental science, modeling of disease incidence and musicology, there is a need for modeling discrete response time series in terms of covariates. In many of these applications the counts are often quite small, at least for a significant portion of the observation times, so as to render the use of traditional Gaussian continuous-valued response time series regression methods inappropriate. In response to this, over the past twenty or so years there has been substantial development of models for discrete-valued time series. For regression models, where interest is focused on making correct inferences about the impact of covariates on the response series, accounting for serial dependence is critical, particularly for obtaining correct standard errors for regression coefficient estimates. Assessing and modeling dependence when the outcomes are discrete random variables can be challenging since traditional methods for detecting series dependence in regression residuals are often not effective. In this paper we consider the GLARMA (generalized linear autoregressive moving average) subclass of observation-driven models in detail. Compared to the parameter driven class of models (see Section 1.1) GLARMA models are relatively easy to fit and provide an accessible and rapid way to detect and account for serial dependence in regression modeling of time series.

The R (R Core Team 2015) **glarma** package (Dunsmuir, Li, and Scott 2015) provides functionality for estimating regression relationships between a vector of regressors (covariates, predictors) and a discrete-valued response and is available from the Comprehensive R Archive Network (CRAN) at <http://CRAN.R-project.org/package=glarma>. In time series regression modeling it is typically the case that there is serial dependence. This package models the serial dependence using the GLARMA class of observation-driven models and provides valid inference for the regression model components. Forecasting for this class of models is straightforward using simulation.

Dunsmuir (2015) provides a review of GLARMA models and their applications and extensions to analysis of multiple independent time series. GLARMA models have found applications in many disciplines including: financial modeling, epidemiological assessments, clinical management, analysis of crime statistics, and primate behavior. Other areas of applications and examples where GLARMA models could readily be applied include those mentioned in the references cited in Sections 2 where we review observation-driven models more widely.

1.1. Generalized state space models

The GLARMA models considered here form a subclass of generalized state space models for non-Gaussian time series described in Davis, Dunsmuir, and Wang (1999), Brockwell and Davis (2010) and Durbin and Koopman (2012) for example. A generalized state space model for a time series $\{Y_t\}$ consists of an observation variable and a state variable. The model is expressed in terms of conditional probability distributions for the observation and state variables. Following Cox (1981) such models can be loosely characterized as either parameter driven or observation driven. The observation specification is the same for both models.

For parameter driven models the state equation commonly consists of a regression component and a latent, usually stationary, time series that cannot be observed directly and which evolves independently of past and present values of the observed responses or the covariates. On the other hand, as the name implies, in observation driven models, the random component of W_t depends on past observations $\{Y_s, s < t\}$ as well as other covariates.

Davis and Dunsmuir (2015) review estimation for parameter driven models including generalized methods of moments, maximum likelihood and composite likelihood methods. In particular, maximum likelihood estimation requires very high dimensional integrals to be evaluated or approximated using asymptotic expansions, simulation methods, numerical integration or all three. Because of this they can be difficult to fit and for routine model building, in which many potential regressors need to be considered and evaluated for significance, the parameter driven models for count time series are not yet ready for general use.

On the other hand, the observation-driven models considered here are much easier to fit because the likelihood is conditionally specified as a product of conditional distributions which belong to the exponential family and for which the natural parameter is readily calculated via recursion. As a result they are relatively straightforward to apply in practical settings with numerous regressors and long time series.

The outline of the remainder of the paper is as follows. Section 2 briefly reviews observation-driven models in order to place the GLARMA models into context. Section 3 provides the necessary theoretical background for GLARMA models. It describes the various combinations of model elements (response distributions, dependence structures and predictive residuals) that are currently supported in the **glarma** package. Options for initializing and obtaining

maximum likelihood estimates using a form of Fisher scoring or Newton-Raphson iterations are described. Issues of parameter identifiability and convergence properties of GLARMA models and the associated maximum likelihood estimates are also reviewed to guide users in the specification of these models. The forecasting methodology for GLARMA models is also described. Section 4 describes the various modeling functions available in the package. Section 5 describes the built-in model diagnostic procedures and plotting functions. Section 6 provides several examples illustrating the use of the package on real data sets. Section 7 reviews other R packages that are available for fitting observation-driven time series models and Section 8 concludes with discussion of some potential future enhancements to the package.

2. Observation-driven models

Particularly over the past twenty or so years, following the early contributions of [Jacobs and Lewis \(1978a,b\)](#), [Cox \(1981\)](#), [Zeger and Qaqish \(1988\)](#) and [McKenzie \(1988\)](#) for instance, there has been substantial development of a range of observation-driven models for time series. A general reference is the book by [Kedem and Fokianos \(2005\)](#). Without aiming to be exhaustive we mention some of the main models that have been developed. For count responses, often modeled using a Poisson or negative binomial distribution conditional on the past of the process and regression variables, reviews can be found in [Davis *et al.* \(1999\)](#), [Jung and Tremayne \(2011\)](#), [Jung, Kukuk, and Liesenfeld \(2006\)](#) and [Fokianos and Tjøstheim \(2011\)](#) for example. The main models discussed include the autoregressive conditional Poisson (ACP) model, the integer-valued autoregressive model (INAR), the integer-valued generalized autoregressive conditional heteroscedastic model (INGARCH), the conditional linear autoregressive process, and the dynamic ordered probit model. For binary responses the autologistic model proposed by [Cox \(1958\)](#) has been extended to the binary autoregressive model (BARMA) proposed by [Li \(1994\)](#).

There is no single model class that covers all of these observation driven models. However [Benjamin, Rigby, and Stasinopoulos \(2003\)](#) propose a quite general class of models, called generalized autoregressive moving average (GARMA) models which we briefly summarize as follows. Let there be n consecutive times at which the response and regressor series are observed. The response series is comprised of observations on the random variables $\{Y_t : t = 1, \dots, n\}$ and associated with these are K -dimensional vectors of regressors x_t also observed for $t = 1, \dots, n$. We let $\mathcal{F}_t = \{Y_s : s = 1, \dots, t-1; x_s : s = 1, \dots, t\}$ and note that this summarizes, relative to any time point t , the past responses and the past and current regressors. The distribution of the current observed response Y_t conditional on \mathcal{F}_t is assumed to be a member of the exponential family of distributions ([McCullagh and Nelder 1989](#)) with density

$$f(y_t|W_t) = \exp \left\{ \frac{y_t W_t - b(W_t)}{\kappa} + c(y_t, \kappa) \right\}, \quad (1)$$

where W_t is the canonical parameter, which we call the “state variable”, κ is the scale parameter, and the functions $b(\cdot)$ and $c(\cdot)$ define the specific member of the exponential family of interest. Then, see [McCullagh and Nelder \(1989\)](#), $\mu_t = \mathbb{E}(Y_t|\mathcal{F}_t) = \dot{b}(W_t)$ and $\text{VAR}(Y_t|\mathcal{F}_t) = \kappa \ddot{b}(W_t)$ are the conditional mean and variance of Y_t given \mathcal{F}_t . Here \dot{b} and \ddot{b} refer to first and second derivatives of $b(\cdot)$ with respect to its argument. A link function, $g(\cdot)$, is used to relate the conditional mean to W_t so that $g(\mu_t) = W_t$. In the GARMA model (and in the GLARMA model of this paper) focus is on the case where the state variable in (1) is

of the general form

$$W_t = x_t^\top \beta + Z_t. \quad (2)$$

If Z_t were not present in W_t , the above model would be the standard generalized linear model (GLM) of [McCullagh and Nelder \(1989\)](#) in which, given the regressors, observations Y_t are independent. In the most general form of the GARMA model [Benjamin *et al.* \(2003\)](#) define Z_t as

$$Z_t = \sum_{j=1}^p \phi_j \mathcal{A}(y_{t-j}, x_{t-j}, \beta) + \sum_{j=1}^q \theta_j \mathcal{M}(y_{t-j}, \mu_{t-j}), \quad (3)$$

where \mathcal{A} and \mathcal{M} are functions for autoregression and moving average components with unknown parameters (in addition to the regression parameters β) ϕ for the autoregressive and θ for the moving average components. Choices for \mathcal{M} suggested in [Benjamin *et al.* \(2003\)](#) are deviance residuals, Pearson residuals, residuals on the original scale ($y_t - \mu_t$) or on the predictor scale ($g(y_t) - W_t$).

[Benjamin *et al.* \(2003\)](#) state that the GARMA model (3) “is too general for practical application” and concentrate on a simplified version of it. In particular they define their GARMA(p, q) model as a model which satisfies (1) and

$$W_t = x_t^\top \beta + \sum_{j=1}^p \phi_j [g(y_{t-j}) - x_t^\top \beta] + \sum_{j=1}^q \theta_j [g(y_{t-j}) - W_{t-j}]. \quad (4)$$

Note that the GARMA class of models consists of models on the predictor scale. [Benjamin *et al.* \(2003\)](#) then present a series of examples such as the Poisson GARMA model and the binomial logistic GARMA model amongst others. However, because the observed responses enter into (4) after transformation by the link function there is a need to introduce an additional arbitrary threshold quantity c to cater for $y_t = 0$ in the Poisson or binomial cases and $y_t = 1$ in the binary case so that $g(y_t)$ is properly defined. For example, for Poisson responses with log link, $g(y_t) = \log(y_t)$ is not defined when $y_t = 0$ which arises often in low count series, which corresponds to the situation for which these types of models are most required. The threshold quantity c is not a parameter that is estimated along with the other parameters and this is a major drawback of the model with the state variable evolving according to (4) when the canonical link is used. One alternative discussed in [Benjamin *et al.* \(2003\)](#) is the identity link in which case the arbitrary threshold parameter c is not needed. However, care must be taken with scaling the residuals in the moving average part of (4) as well as the regression variables – see [Davis *et al.* \(1999\)](#) for details. Some R packages exist for estimation of some GARMA models and these are reviewed in Section 7.

A variant of the GARMA and GLARMA models for binary series is the binary autoregressive moving average (BARMA) model suggested by [Li \(1994\)](#), and discussed more recently in [Wang and Li \(2011\)](#). This has a state variable defined by

$$W_t = x_t^\top \beta + \sum_{i=1}^p y_{t-i} + \sum_{i=1}^q (y_{t-i} - \mu_{t-i}), \quad (5)$$

where μ_t is the conditional probability of a success at time point t given \mathcal{F}_t . For binary time series good reviews of modeling approaches are provided in [Startz \(2008\)](#) and [Kauppi and Saikkonen \(2008\)](#). Their focus is on predicting U.S. quarterly recessions and they demonstrate the utility of dynamics in the model for W_t , a feature of GARMA and GLARMA

models. Models of this type without a moving average term could be fit using standard GLM procedures by defining the regressors x_t to include lagged values of the response series without mean adjustment or scaling. Cross product terms between y_{t-j} and y_{t-k} or x_{t-k} can be included also – see [Startz \(2008\)](#) or [Kauppi and Saikkonen \(2008\)](#) for details. Models of this type have also been considered by [Rydberg and Shephard \(2003\)](#) as a component of a within day stock price movement model but they suggest that these are numerically unstable and recommend using a GLARMA specification instead. The **glarma** package can be used to fit BARMA models (see below). Note that the model considered by [Kauppi and Saikkonen \(2008\)](#) in their Equation 4 is not a BARMA model because the recursions for W_t include the regression term whereas in GLARMA and BARMA models the regression term and the dynamics are additively combined.

3. Theory of GLARMA models

GLARMA models are similar in structure to GARMA models with some important differences which will be apparent once we give a complete description of the former. In general the conditional distribution of Y_t given \mathcal{F}_t can be a member of the exponential family (1). However, to describe the GLARMA model as currently implemented in **glarma** we define a simplified exponential family form in which the dispersion parameter $\kappa = 1$ and, in order to accommodate the binomial distribution in a simple way we define the b function to be scaled by a factor a_t . The simplified version of (1) is

$$f(y_t|W_t) = \exp \{y_t W_t - a_t b(W_t) + c_t\}, \quad (6)$$

where a_t and c_t are sequences of constants possibly depending on the observations y_t . Details of the specific distributions currently available in **glarma** are provided in Section 3.1.

Note that (6) is not the fully general form of the exponential family (see [McCullagh and Nelder 1989](#)) in that it does not include an over-dispersion parameter and the canonical link is used. It follows from (6), as before for the GARMA specification, that the conditional means and variances of the responses are $\mu_t := E(Y_t|W_t) = a_t \dot{b}(W_t)$ and $\sigma_t^2 := \text{VAR}(Y_t|W_t) = a_t \ddot{b}(W_t)$. The negative binomial case is not a member of the exponential family when the scale parameter is not fixed and therefore needs to be treated in a special way – see below.

For GLARMA models as implemented in the **glarma** package the state variable is defined as

$$W_t = x_t^\top \beta + O_t + Z_t, \quad (7)$$

in which there is an additional offset term O_t included because for many applications this may be important. For example in modeling motor vehicle driver deaths, disease counts or crime counts using a Poisson distribution, inclusion of the logarithm of population as an offset term is important and provides a model for the incidence per unit of population. The offset term enters the model without regression parameters.

3.1. Response distributions

Response distributions currently supported in the **glarma** package are Poisson and binomial (which includes Bernoulli as a special case), both of which can be put in the form of (6), and

	Poisson	Binomial	Negative Binomial
$f(y W_t)$	$\frac{\mu_t^y e^{-\mu_t}}{y!}$	$\binom{m_t}{y} \pi_t^y (1 - \pi_t)^{m_t - y}$	see (8)
a_t	1	m_t	-
c_t	$-\log y_t!$	$\log \binom{m_t}{y_t}$	-
$b(W_t)$	e^{W_t}	$\log(1 + e^{W_t})$	-
μ_t	e^{W_t}	$m_t \pi_t$	e^{W_t}
σ_t^2	μ_t	$m_t \pi_t (1 - \pi_t)$	$\mu_t + \mu_t^2 / \alpha^2$
Link	log	logit	log

Table 1: Distributions and associated link functions supported in the **glarma** package.

the negative binomial distribution with density in the form given by

$$f(y_t|W_t, \alpha) = \frac{\Gamma(\alpha + y_t)}{\Gamma(\alpha)\Gamma(y_t + 1)} \left[\frac{\alpha}{\alpha + \mu_t} \right]^\alpha \left[\frac{\mu_t}{\alpha + \mu_t} \right]^{y_t}, \quad (8)$$

where $\mu_t = \exp(W_t)$. This cannot be put into the one parameter exponential family form when α is not known, which, however, is the case of most practical interest. As $\alpha \rightarrow \infty$ the negative binomial density converges to the Poisson density. Table 1 specifies the distributions and associated link functions currently supported in the **glarma** package.

3.2. GLARMA dependence structure

Serial dependence in the response process W_t given by (7) is, in a similar vein to Benjamin *et al.* (2003), introduced by specifying Z_t as an autoregressive moving average recursion of the form

$$Z_t = \sum_{i=1}^p \phi_i (Z_{t-i} + e_{t-i}) + \sum_{i=1}^q \theta_i e_{t-i}, \quad (9)$$

in which the predictive residuals are defined as

$$e_t = \frac{Y_t - \mu_t}{\nu_t} \quad (10)$$

for some scaling sequence $\{\nu_t\}$ – see Section 3.3 for choices currently supported.

This specification allows recursive calculation (in t) of the state equation. The model is referred to as a GLARMA model in Davis, Dunsmuir, and Streett (2003). Shephard (1995) provides the first example of such a model class. Rydberg and Shephard (2003) define the GLARMA model in the following form

$$Z_t = \sum_{i=1}^p \phi_i Z_{t-i} + \sigma e_{t-1} + \sigma \sum_{i=1}^q \theta_i e_{t-1-i}, \quad (11)$$

which can be made equivalent to (9) by suitable redefinition of the degrees p and q and the autoregressive and moving average coefficients as shown in Section 3.4.

The name GLARMA, which we prefer to GARMA, was originally used in Shephard (1995), an unpublished paper, and subsequently in Rydberg and Shephard (2003) and Davis *et al.* (2003). This terminology captures the essence of the models we consider which are generalizations of both generalized linear (GL) and autoregressive moving average (ARMA) models.

We can also write Z_t given by (9) using linear combinations of past predictive residuals e_t (Davis *et al.* 1999, 2003), i.e.,

$$Z_t = \sum_{j=1}^{\infty} \gamma_j e_{t-j}, \quad (12)$$

where γ_j are given as coefficients in the power series

$$\sum_{j=1}^{\infty} \gamma_j \zeta^j = \theta(\zeta)/\phi(\zeta) - 1,$$

with $\phi(\zeta) = 1 - \phi_1\zeta - \dots - \phi_p\zeta^p$ and $\theta(\zeta) = 1 + \theta_1\zeta + \dots + \theta_q\zeta^q$ being the respective autoregressive and moving average polynomials of the ARMA filter, each having all zeros outside the unit circle. It follows that the $\{Z_t\}$ defined in this way can be thought of as the best linear predictor of a stationary invertible ARMA process with driving noise specified by the sequence $\{e_t\}$ of scaled deviations of count responses from their conditional mean given the past responses and the past and current regressors.

3.3. Types of GLARMA residuals

GLARMA predictive residuals are of the form (10) where $\nu(W_t)$ is a scaling function. Currently several choices for this are supported.

Pearson residuals: These are scaled by the standard deviation, σ_t (defined in Table 1) of the predictive distribution $\nu_t = \nu_{P,t} = \sigma_t$.

Score-type residuals: These replace the conditional standard deviation by the conditional variance $\nu_t = \nu_{S,t} = \sigma_t^2$ as suggested in Creal, Koopman, and Lucas (2008).

Identity residuals: These use no scaling so that $\nu_t = \nu_{I,t} = 1$.

The use of identity residuals allows BARMA models considered in Wang and Li (2011) to be fit using the **glarma** package. For the Poisson response distribution GLARMA model, failure to scale by the variance or standard deviation function will lead to unstable Poisson means (that diverge to infinity or collapse to zero as an absorbing state for instance) and existence of stationary and ergodic solutions to the recursive state equation is not assured – see Davis *et al.* (1999), Davis *et al.* (2003) and Davis, Dunsmuir, and Streett (2005) for details. For the binomial situation this lack of scaling should not necessarily lead to instability in the success probability as time evolves since the success probabilities, π_t , and observed responses, Y_t , are both bounded between 0 and 1. Thus degeneracy can only arise if the regressors x_t become unbounded from below or above. As recommended in Davis *et al.* (1999) temporal trend regressors should be scaled using a factor relating to sample size n .

Note that for the distributions specified in Table 1 and provided that the regressors are such that the choice of scaling $\nu_t \neq 0$, then the e_t have zero mean and are uncorrelated (Davis *et al.* 2003). When ν_t is set to the conditional standard deviation of Y_t , the e_t also have unit variance, hence are weakly stationary white noise.

Note on BARMA models These can be put into the GLARMA framework by including lagged values of the response variable in the regression matrix as additional columns and specifying moving average terms with identity residuals. [Startz \(2008\)](#) cautions that in BARMA models it is possible to get parameter estimates such that $\pi_t = 0$ or $\pi_t = 1$ numerically in which case the information matrix for the model is singular and estimates of parameter covariances are not defined.

The above residuals provide a reasonable range of possible choices. [Dunsmuir, Leung, and Liu \(2004\)](#) investigate the use of parametric scaling for the Poisson response case where $e_t = (y_t - \mu_t)/\mu_t^\lambda$ which gives Pearson residuals when $\lambda = 0.5$ and score residuals when $\lambda = 1$. They estimate λ by profiling the GLARMA likelihood. For the Polio data of [Zeger \(1988\)](#) they obtain the estimate $\hat{\lambda} = 1.14$ which is not significantly different from 1 (score residuals) but is significantly different from 0.5 (Pearson residuals). This method has not been implemented in the **glarma** package because it is cumbersome to control convergence and has limited application (mainly to Poisson responses). However, the models using Pearson and score residuals could be fitted and their likelihoods compared to choose between $\lambda = 0.5$ and $\lambda = 1$. Another alternative is to use Anscombe residuals which are claimed to produce residuals which, although discrete, may be better matched moment-wise to the normal distribution particularly in regard to skewness. These have been discussed in [Davis *et al.* \(1999\)](#) for developing methods for testing for serial dependence. Including them in the **glarma** package would require only a modest amount of work for the Poisson and negative binomial response distributions but it would be substantially more involved for the binary or binomial response, since in these cases, the Anscombe residuals require a computation using the incomplete beta function which is defined in terms of the model parameters through the conditional probability of success. Hence the computation of derivatives of the likelihood would be quite challenging.

3.4. Parameter identifiability

The GLARMA component Z_t of the state variable given in (9) can be rewritten as

$$Z_t = \sum_{i=1}^p \phi_i Z_{t-i} + \sum_{i=1}^{\tilde{q}} \tilde{\theta}_i e_{t-i}, \quad (13)$$

where $\tilde{q} = \max(p, q)$ and:

1. If $p \leq q$, $\tilde{\theta}_j = \theta_j + \phi_j$ for $j = 1, \dots, p$ and $-\tilde{\theta}_j = \theta_j$ for $j = p+1, \dots, q$.
2. If $p > q$, $\tilde{\theta}_j = \theta_j + \phi_j$ for $j = 1, \dots, q$ and $\tilde{\theta}_j = \phi_j$ for $j = q+1, \dots, p$.

When pre-observation period values are set to zero (that is $Z_t = 0$ for $t \leq 0$ and $e_t = 0$ for $t \leq 0$) then, if and only if $\tilde{\theta}_j = 0$ for $j = 1, \dots, \tilde{q}$, the recursion (13) would result in $Z_t = 0$ for all t and hence there is no serial dependence in the GLARMA model. This is equivalent to $\phi_j = -\theta_j$ for $j = 1, \dots, p$ and $\theta_j = 0$ for $j = p+1, \dots, \tilde{q}$.

Consequently the null hypothesis of no serial dependence requires only these constraints on the θ and ϕ parameters. In some situations this means that under the null hypothesis of no serial dependence there are nuisance parameters which cannot be estimated. This has implications for the convergence behavior, i.e., the number of iterations required to optimize the likelihood, and on testing that there is no serial dependence in the observations (other than that induced by the regression component $x_t^\top \beta$).

When $p > 0$ and $q = 0$ (equivalent to an ARMA(p, p) specification with constraint $\theta_j = \phi_j$ or a pure MA with $p = 0$ and $q > 0$), then identification issues do not arise and the hypothesis of no serial dependence corresponds to the hypothesis that $\phi_j = 0$ for $j = 1, \dots, p$ in the first case and $\theta_j = 0$ for $j = 1, \dots, q$ in the second case. The likelihood ratio and Wald tests (see Section 5.1 for further details) provided in the **glarma** package will have an asymptotic chi-squared distribution with correct degrees of freedom.

In cases where $p > 0$ and $q > 0$ some caution is advised when fitting models and testing that serial dependence is not present. To simplify the discussion we focus on the case where $p = q$.

1. If there is no serial dependence in the observations but $p = q > 0$ is specified then there is a strong possibility that the likelihood optimization for this overspecified model will not converge because the likelihood surface will be “ridge-like” along the line where $\phi_j = -\theta_j$. This issue is classical for standard ARMA models. Similarly if the degree of serial dependence is of lower order than that specified for the GLARMA model, identifiability issues and lack of convergence of the likelihood optimizing recursions are likely to occur. Lack of identifiability typically manifests itself in the matrix of second derivatives D_{NR} or the approximate Fisher scoring version D_{FS} becoming close to singular or even non-positive definite. The state variable W_t can also degenerate to $\pm\infty$ in which case an error code in the output of the `glarma()` call is provided.
2. The likelihood ratio test, that there is no serial dependence, versus the alternative, that there is GLARMA-like serial dependence with $p = q > 0$, will not have a standard chi-squared distribution, because the parameters ϕ_j for $j = 1, \dots, p$ are nuisance parameters which cannot be estimated under the null hypothesis. Testing methods such as proposed in Hansen (1996) for example need to be developed for this situation.

3.5. The GLARMA likelihood

Given n successive observations $\{y_t : t = 1, \dots, n\}$ of the response series, the likelihood, conditional on initial values for the recursions to calculate Z_t , is constructed as the product of conditional densities of Y_t given \mathcal{F}_t . The state vector W_t at each time point embodies these conditioning variables as well as the unknown parameters $\delta = (\beta, \phi, \theta)$ and in order to stress this we will use the notation $W_t(\delta)$ where appropriate. Then, the log-likelihood is given by

$$l(\delta) = \sum_{t=1}^n \log f(y_t | W_t(\delta)). \quad (14)$$

Strictly speaking this is a conditional (on initial values) log-likelihood but for brevity we will henceforth use the term log-likelihood. For the Poisson and binomial response distributions the log-likelihood (14) is

$$l(\delta) = \sum_{t=1}^n \{y_t W_t(\delta) - a_t b(W_t(\delta)) + c_t\}. \quad (15)$$

For the negative binomial response distribution the log-likelihood is based on (8) and the shape parameter α also has to be estimated along with β , ϕ and θ . We then let $\delta = (\beta, \phi, \theta, \alpha)$.

Note that the e_t in (10) and the Z_t in (9) are also functions of the unknown parameter δ and hence need to be recomputed for each iteration of the likelihood optimization. Thus

in order to calculate the likelihood and its derivatives, recursive expressions are needed to calculate e_t , Z_t and W_t as well as their first and second partial derivatives with respect to δ . Expressions for these recursive formulae are available in [Davis *et al.* \(2005\)](#) for the Poisson case. Corresponding formulae for the binomial case were derived in [Lu \(2002\)](#) and for the negative binomial case in [Wang \(2004\)](#). The essential computational cost is in the recursions for Z_t and W_t and their first and second derivatives with respect to δ . Fortunately, these require identical code for the various response distributions and definitions of predictive residuals e_t .

The likelihood is maximized from a suitable starting value of the parameter δ using a version of Fisher scoring iteration or by Newton-Raphson iteration. For a given value of δ let the vector of first derivatives with respect to δ of the log-likelihood (14) be

$$d(\delta) = \frac{\partial l(\delta)}{\partial \delta}$$

and the second derivative matrix be

$$D_{NR}(\delta) = \frac{\partial^2 l(\delta)}{\partial \delta \partial \delta^\top}, \quad (16)$$

where the matrix of second derivatives of the log-likelihood is (in the Poisson and binomial response cases) given by

$$D_{NR}(\delta) = \sum_{t=1}^n [y_t - a_t \dot{b}(W_t)] \frac{\partial^2 W_t}{\partial \delta \partial \delta^\top} - \sum_{t=1}^n a_t \ddot{b}(W_t) \frac{\partial W_t}{\partial \delta} \frac{\partial W_t}{\partial \delta^\top}. \quad (17)$$

Using the fact that, at the true parameter value δ , $\mathbb{E}[y_t - a_t \dot{b}(W_t) | \mathcal{F}_t] = 0$, the expected value of the first summation in (17) is zero and hence the expected value of the matrix of second derivatives is $\mathbb{E}[D_{FS}(\delta)]$, where

$$D_{FS}(\delta) = - \sum_{t=1}^n a_t \ddot{b}(W_t) \frac{\partial W_t}{\partial \delta} \frac{\partial W_t}{\partial \delta^\top}. \quad (18)$$

Since $\mathbb{E}(y_t - a_t \dot{b}(W_t) | \mathcal{F}_t) = 0$ it follows that $\mathbb{E}[D_{NR}(\delta)] = -\mathbb{E}[d(\delta)d(\delta)^\top]$. While these expectations cannot be computed in closed form, expression (18) requires first derivatives only and is used in the **glarma** package as the basis for the approximate Fisher scoring method. Thus, if $\delta^{(k)}$ is the parameter vector at the current iterate k , the Newton-Raphson updates proceed using

$$\delta^{(k+1)} = \delta^{(k)} - D_{NR}(\delta^{(k)})^{-1} d(\delta^{(k)}) \quad (19)$$

and the approximate Fisher scoring updates use D_{FS} in place of D_{NR} . Given a specified tolerance TOL , iterations continue until the largest gradient of the log-likelihood satisfies $\max_i |d_i(\delta^{(k)})| \leq TOL$ or a maximum number of iterations $MAXITER$ is surpassed. At termination we let $\hat{\delta} = \delta^{(k+1)}$ and call this the ‘‘maximum likelihood estimate’’ of δ .

Selecting initial values of the state variable For calculation of the Z_t in (9), initializing conditions for the recursions must be used. The current implementation in **glarma** is to set $e_t = 0$ and $Z_t = 0$ for $t \leq 0$ ensuring that the conditional and unconditional expected values of e_t are zero for all t . These would seem to be natural specifications for GLARMA models

and are used in Rydberg and Shephard (2003) and Davis *et al.* (2003). In BARMA models (5) the recursions for W_t are in terms of past Y_t and past unscaled deviations $Y_t - \mu_t$. While setting pre-period values of these latter to zero may be natural, there remains the question of how best to initialize Y_t for $t \leq 0$. For example Kauppi and Saikkonen (2008) suggest using a method that can be “interpreted as estimates of the unconditional mean” of W_t and Startz (2008) suggests setting μ_t to zero, or the sample mean of Y_t , or $\pi_t = 0.5$ or initial values of $y_t - \mu_t$ to zero. Currently to use the **glarma** package to fit BARMA models $y_t - \mu_t = 0, t \leq 0$ is assumed and the users are free to specify values y_t for $t \leq 0$ in whatever way they wish.

Choosing lags for the ARMA components Selection of appropriate lags for the AR and MA components in the GLARMA model is usually considerably more difficult than it is for Gaussian series where residuals from least squares model fits can provide quite useful information about the serial dependence structure through use of the autocorrelation and partial autocorrelation functions. Unlike residuals from least squares regression, which often provide useful information about serial dependence model structure for continuous responses, in the discrete-valued response setting, residuals from the GLM fit often do not give good guidance on choosing p and q needed to specify the GLARMA model, particularly if the dependence is weak or moderate. We would recommend that the autocorrelation and partial autocorrelation functions be estimated using GLM residuals and, if strong patterns are observed, that these be reflected in the choice of p and q . However, following the discussion on parameter identifiability in Section 3.4 it is also highly recommended that users start with low orders for p and q and initially avoid specifying them to be equal. Once stability of estimation is achieved for a lower order specification increasing the values of p or q could be attempted.

Initializing the MLE recursions By default, the iterations in (19) are initialized using the GLM estimates for β and zero initial values for the autoregressive moving average terms. For the negative binomial case, β and α are initialized using a call to `glm.nb()` from package **MASS** (see Venables and Ripley 2002). While this is often a successful strategy, for which convergence to the MLE is typically rapid, it does not always lead to convergence or can take quite a few iterations. In some cases, using the Fisher scoring method can be more robust to starting values. Thus, this can be used to initialize the iterative algorithm and to then switch to Newton-Raphson once the procedure is reasonably stable, because Newton-Raphson is typically faster once the iterations are close to the optimum. Users may optionally specify initial parameter values of their own choice. This can be important particularly when strong serial dependence is required to be modeled or when the model may be overspecified and hence poorly identified because the likelihood surface is flat in some directions in parameter space. In the former case, the autocorrelation function of the residuals from an initial GLM fit can guide selection of AR and MA terms and initial values for these. Alternatively the autocorrelation of the randomized probability integral transformation residuals might provide guidance. When the model is poorly identified it is particularly difficult to obtain convergence of the likelihood iterations as discussed in more detail in Section 3.4. Our recommendation is to start with moving average terms at small order lags and seasonal lags (if appropriate) and build up a sequence of models with the moving average order, q , increasing. Then, examination of the moving average coefficient estimates, $\hat{\theta}_j$ can sometimes suggest a simpler autoregressive or mixed autoregressive moving average specification.

3.6. Stochastic properties of GLARMA models

Sufficient conditions for the mean and variance of the predictive residuals to exist are, for example, that the autoregressive polynomial $\phi(\zeta)$ has all zeros outside the unit circle, the regressors x_t are such that $0 < \sigma_t < \infty$ and the recursions in (9) are initialized with $Z_t = e_t = 0$ for $t \leq 0$. Assuming then that $\sigma_t < \infty$ in the Poisson and negative binomial case and that $0 < \pi_t < 1$ and $m_t < \infty$ in the binomial case the three types of residuals considered will have finite mean and variance at any time point t . In that case they form a martingale difference sequence (Rydberg and Shephard 2003; Davis *et al.* 2003) and as a result of that are uncorrelated. Further the Pearson residuals (and only those) have unit variance and therefore are zero mean, unit variance weakly stationary white noise. Means, variances and autocovariances for the state process $\{W_t\}$ can be readily derived using the definition of Z_t in (12) – see (Davis *et al.* 1999). For the Poisson response case the corresponding means, variance and autocovariances for the count response series $\{Y_t\}$ can be derived approximately. Additionally an approximate interpretation of the regression coefficients β can be given – see (Davis *et al.* 2003). Similar results could be derived for the negative binomial response case. For binomial and Bernoulli responses, calculation of means, variances, autocovariances for the response series and interpretation of regression coefficients is not straightforward. This is a typical issue for interpretation of random effects models and transition models in the binomial or Bernoulli case – see Diggle, Heagerty, Liang, and Zeger (2002) for example.

To date the stationarity and ergodicity properties of the GLARMA model are only partially understood. These properties are important in order to ensure that the process is capable of generating sample paths that do not degenerate to zero or do not explode as time progresses, as well as for establishing the large sample distributional properties of estimates of the parameters. Davis *et al.* (2003) provide partial results for perhaps the simplest of all possible models for Poisson responses specified with $p = 0$, $q = 1$ and $x_t^\top \beta = \beta$. Results for simple examples of the stationary Bernoulli case are given in Streett (2000). Recently Davis and Liu (2012) have considered a general class of observation-driven models for one parameter exponential family response distributions. As they point out, the GLARMA processes do not necessarily satisfy the contraction condition required for the existence of a stationary distribution except in simple cases of Davis *et al.* (2003). These cases exclude regression terms in W_t .

Increasingly, asymptotic results for various types of observation driven models without covariates are becoming available. Tjøstheim (2012) reviews the ergodic and stationarity properties for some observation-driven models primarily in the Poisson response context. However these results do not apply to the specific form of GLARMA models that are presented here. Wang and Li (2011) discuss the BARMA model and present some asymptotic results for that case. However the state equation for the BARMA model differs structurally from that for the binary GLARMA model, because past observations of Y_t also enter in the autoregressive terms without scaling or centering, whereas in the GLARMA model they only enter through the residuals. Woodard, Matteson, and Henderson (2011) present some general results on stationarity and ergodicity for the form of the GARMA model of Benjamin *et al.* (2003). However these results are not directly applicable to the GLARMA model form considered here because the state equation recursions again involve applying the link function to both the responses and the mean responses. None of these recent results consider the case of covariates and hence are not, as yet, applicable to likelihood estimation for regression models for discrete outcome time series.

3.7. Fitted values

There are two concepts of fitted values currently supported for the GLARMA model. The first is defined as the estimated conditional mean function $\hat{\mu}_t$ at time point t calculated using the maximum likelihood estimates $\hat{\delta}$. Thus

$$\hat{\mu}_t = a_t \dot{b}(x_t^\top \hat{\beta} + \hat{Z}_t), \quad (20)$$

where \hat{Z}_t are calculated using $\hat{\delta}$ in (9). These fitted values combine the regression fit (fixed effects) together with the contribution from weighted sums of past estimated predictive residuals.

Because for GLARMA models the unconditional mean function is difficult to obtain exactly in all cases an estimated unconditional mean function of t is not provided. Instead, for the second concept of fitted values, the fitted value from the regression term only is suggested as a guide to the fit without the effect of random variation due to Z_t . This is defined to be

$$\tilde{\mu}_t = a_t \dot{b}(x_t^\top \hat{\beta}). \quad (21)$$

We refer to this as the “fixed effects fit” in plotting functions below. Note that this is not an estimate of the unconditional mean even in the Poisson case (arguably the most tractable for this calculation). The theoretical unconditional mean for this case is approximated by $\exp(x_t^\top \beta + \nu^2/2)$ where $\nu^2 = \sum_{l=1}^{\infty} \gamma_l^2$ – see [Davis *et al.* \(2003\)](#) for details. A similar calculation for the binomial case is not available. Hence, in view of these theoretical constraints, the use of the fixed effects fit seems a simple and sensible alternative to the conditional mean $\hat{\mu}_t$ given by (20).

3.8. Distribution theory for likelihood estimation

For inference in the GLARMA model it is assumed that the central limit theorem holds so that

$$\hat{\delta} \stackrel{d}{\approx} N(\delta, \hat{\Omega}), \quad (22)$$

where the approximate covariance matrix is estimated by

$$\hat{\Omega} = -D_{NR}(\hat{\delta})^{-1} \quad (23)$$

in the case of Newton-Raphson and similarly with D_{NR} replaced by D_{FS} in the case of Fisher scoring. Thus a standard error for the maximum likelihood estimates of the i th component of δ is computed using $\hat{\Omega}_{ii}^{1/2}$.

There have been a number of claims in the literature concerning a central limit theorem for models of this type. However all of these make assumptions concerning convergence of key quantities which require ergodicity to be established which has not been done in generality yet. The central limit theorem for the maximum likelihood parameter estimates is rigorously established only in the stationary Poisson response case in [Davis *et al.* \(2003\)](#) and in the Bernoulli stationary case in [Streett \(2000\)](#). Simulation results are also reported in [Davis *et al.* \(1999, 2003\)](#) for non-stationary Poisson models. Other simulations not reported in the literature support the supposition that the estimates $\hat{\delta}$ have a multivariate normal distribution for large samples for a range of regression designs and for the various response distributions considered here. Hence, while a central limit theorem for the maximum likelihood estimators

is currently not available for the general model it seems plausible, since for these models the log-likelihood is a sum of elements in a triangular array of martingale differences.

For nested models likelihood ratio test statistics and Wald statistics can be calculated. Whenever the central limit theorem (22) holds an approximate chi-squared distribution with appropriate degrees of freedom can be used to assess significance. Let $\delta^{(1)}$ specify a subset of δ that is hypothesized to take a specific value $\delta_0^{(1)}$. The Wald test is constructed as

$$W^2 = [\hat{\delta}^{(1)} - \delta_0^{(1)}]^\top [\hat{\Omega}^{(1)}]^{-1} [\hat{\delta}^{(1)} - \delta_0^{(1)}], \quad (24)$$

where $\hat{\Omega}^{(1)}$ is the submatrix corresponding to $\delta_0^{(1)}$ of the estimated asymptotic covariance matrix of (23). Further details on implementation of these tests in **glarma** are given in Section 5.1.

Associated with testing of the need for GLARMA terms in the model is the use of residuals to investigate serial dependence not accounted for by the model fitted so far. If the residuals from a GLM fit (without GLARMA terms) are used to calculate estimated autocorrelations and partial autocorrelations in the usual way then it is straightforward to show that, under the conditions on the regressors as given in Davis and Wu (2009), these will be asymptotically normally distributed with standard error given by the usual $1/\sqrt{n}$. This provides the usual 95% intervals in the diagnostic plots provided in **glarma**. If the residuals are calculated after fitting GLARMA terms in the model, then the asymptotic theory is not yet available to establish the corresponding result. However the first author's experience with simulations suggests that the autocorrelations estimated using residuals from the GLARMA model fit are asymptotically normal with the usual standard error. Because the residuals from an adequately specified GLARMA model are martingale differences their autocorrelations at the true parameters are uncorrelated with standard error $1/\sqrt{n}$ as usual. Asymptotic normality of the estimated autocorrelations is also likely. Experience with simulations suggests that, for example, the usual Box-Ljung portmanteau test for serial dependence has the correct asymptotic chi-squared distribution.

3.9. Forecasting GLARMA models

Methods for forecasting future values of the observed time series using observation-driven models are not as well developed as they are for the traditional ARMA or ARMAX models for continuous responses. The conditional specification of the observation-driven model means that one-step ahead forecasts are simple. However for multi-step ahead forecasts the conditional specification means that all possible future sample paths over the forecast horizon need to be considered either theoretically or via simulation. We discuss these aspects here.

For one-step ahead forecasts of Y_{n+1} from the last observation at time point n , provided values of the regressors x_{n+1} at time point $n+1$ are available either by knowledge or separate forecasting, then the conditional distribution of Y_{n+1} may be forecast simply by forecasting the state variable W_{n+1} . However, this can be done using the parameter estimates in the observation-driven model. We illustrate this for the GLARMA model with similar formulae applying for GARMA and BARMA models.

$$\begin{aligned} \hat{W}_{n+1} &= x_{n+1}^\top \hat{\beta} + \hat{Z}_{n+1}, \\ \hat{Z}_{n+1} &= \sum_{j=1}^p \hat{\phi}_j (\hat{Z}_{n+1-j} + \hat{e}_{n+1-j}) + \sum_{j=1}^q \hat{\theta}_j \hat{e}_{n+1-j}. \end{aligned} \quad (25)$$

Note that \hat{Z}_{n+1} can be completely determined using values of \hat{Z}_t and \hat{e}_t for $t \leq n$. Then the predictive distribution for Y_{n+1} is estimated to be $f(y|\hat{W}_{n+1})$ where the density f is given by (6). Obviously this estimated density provides a complete description of the forecast and could be presented as a bar chart or features such as its mean, mode or median for a single point forecast could be used. Additionally one could determine prediction intervals but due to the discreteness of the distribution these will not correspond to commonly selected prediction coverage probabilities such as 95% for example.

Multi-step ahead forecasts are more involved as has been noted by several authors. [Rydberg and Shephard \(2003\)](#) give the following general formula which applies for lead times $L > 1$.

$$f(y_{n+L}|\mathcal{F}_n) = \sum_{y_{n+1}} \cdots \sum_{y_{n+L-1}} \prod_{j=1}^L f(Y_{n+j}|\mathcal{F}_{n+j}). \quad (26)$$

Here the information sets \mathcal{F}_{n+j} are defined to combine the actual information in \mathcal{F}_n which is available, knowledge of the regressors x_{n+1}, \dots, x_{n+j} and the values of $y_{n+1}, \dots, y_{n+j-1}$ used in the summations on the right hand side of (26). The summation then is over all possible projected sample paths over $n+1, \dots, n+L-1$, the number of which will grow at least exponentially with L . The state variables W_{n+j} required to specify the conditional densities in (26) need to be updated recursively. However the GLARMA updating equations are rapid to compute. [Rydberg and Shephard \(2003\)](#) suggest that complete enumeration of the sums and products is impossible and suggest either doing this by truncating the state space for responses Y_t (relevant in the Poisson and negative binomial cases) or by simulation.

The binomial, and particularly the binary case may be amenable to complete enumeration and, indeed, such an approach is suggested in [Kauppi and Saikkonen \(2008\)](#). In the binary case [Startz \(2008\)](#), [Klingenberg \(2008\)](#), and in an appendix [Kauppi and Saikkonen \(2008\)](#), discuss implementation of (26) for binary dynamic models including the BARMA type structure and provide formulae for analogous terms on the right hand side by summing over all possible paths of zeros and ones over the forecast period. For binary data this will provide non-simulation based forecasts of the future probabilities of success conditional on available data. Extension of this method to binomial, Poisson or negative binomial distributions will be computationally and algebraically complex. Because of this, simulation is the approach that has been implemented in the **glarma** package for all distributions currently supported. [Benjamin et al. \(2003\)](#) also utilize the simulation approach and suggest that this is feasible for short-range forecasting. Again they mention that the regressors need to be known in the future or, if stochastic, are capable of being forecast.

Other recent work on forecasting includes [McCabe, Martin, and Harris \(2011\)](#), who study multi-period ahead forecasts within the context of the integer autoregressive class of models, and [Freeland and McCabe \(2004\)](#), for the Poisson autoregression model. These models are not in the GLARMA class of models.

4. Modeling functions in glarma

There are seven modeling functions for fitting GLARMA models, falling into three groups:

Poisson: `glarmaPoissonPearson()` and `glarmaPoissonScore()`.

Binomial: `glarmaBinomialIdentity()`, `glarmaBinomialPearson()` and `glarmaBinomialScore()`.

Negative binomial: `glarmaNegBinPearson()` and `glarmaNegBinScore()`.

The second component of the name indicates the distribution used for the counts and the third component the residuals used in the fitting routine. A call to `glarma()` results in a call to the appropriate fitting routine, as determined by the values of the arguments `type` and `residuals` supplied to the `glarma()` call. Pearson residuals are used by default. Two iterative methods are available for the optimization of the log-likelihood, Fisher scoring (`method = "FS"`) and Newton-Raphson (`method = "NR"`), the default method being Fisher scoring. The object returned by any of the fitting routines is of class ‘`glarma`’.

To specify the model in a call to `glarma()`, the response variable is given by the argument `y`, and the matrix of predictors for the regression part of the model is given by the argument `X`. The matrix `X` must include a column of ones to enable the fitting of a mean term in the regression component of the model. If required for W_t in (7), an offset term O_t must be specified also. Initial values can be given for the coefficients in the regression component using the argument `beta`. If no initial values are provided, a call is made to the corresponding GLM to obtain initial regression coefficient values.

The ARMA component of the model is specified using the arguments `phiLags` and `phiInit` (for the AR terms) and `thetaLags` and `thetaInit` (for the MA terms). For both the AR and MA terms, the first argument of the pair of arguments specifies the orders of the lags which are to be included in the model, and the second argument the initial values of the coefficients for those lags.

When the counts are modeled using the negative binomial distribution, there is an additional parameter, the shape parameter of the negative binomial, designated as α in the GLARMA model. This parameter is called θ in the function `glm.nb()` from package **MASS**, but for GLARMA models θ refers to the moving average terms in the ARMA component of the model. An initial value for α can be provided using the argument `alphaInit`. If no initial value is provided, a call is made to `glm.nb()` from **MASS**. An initial value for the call to `glm.nb()` can be supplied by giving a value to the argument `alpha` of `glarma()`. The default value for `alpha` is 1.

Because the GLARMA model is fitted using numerical non-linear optimization, it is possible that non-convergence occurs. Two error codes are included in the object returned by `glarma()` to alert users to numerical problems which occurred during fitting. If the Fisher scoring or Newton-Raphson iterations fail to converge, `errorCode` will be set to 1. This can result from non-identifiability of the ARMA component of the model such as when the degrees and lags of both the AR and MA components are specified to be the same, as discussed in Section 3.4. It is possible that for certain values of the ARMA parameters the recursions calculating $\{W_t\}$ diverge to $\pm\infty$. In that case the value of `wError` will be set to 1 allowing the user to check for this condition when the likelihood optimization fails to converge.

Once a fitted model object has been obtained, there are accessor functions available using S3 methods to extract the coefficients (`coef()`, or the alias `coefficients()`), the fitted values (`fitted()` or the alias `fitted.values()`), the residuals (`residuals()` or the alias `resid()`), the model frame (`model.frame()`), the number of observations (`nobs()`), the log-likelihood (`logLik()`), and the AIC (`extractAIC()`). These are standard implementations of the meth-

ods with the exception of `coef()`. This method takes an argument `types` which allows the extraction of the ARMA coefficients (`types = "ARMA"`), or the regression coefficients (`types = "beta"`), or both sets of coefficients (`types = "all"`), the default.

For an object of class ‘`glarma`’ the package includes S3 `print`, `summary`, and `plot` methods as well as a `print` method for the object returned by `summary`.

5. Diagnostics

5.1. Likelihood ratio and Wald tests

In `glarma`, the likelihood ratio test and the Wald test both test that the serial dependence parameters $\psi = (\phi^\top, \theta^\top)^\top$ are all equal to zero (that is $H_0 : \psi = 0$ versus $H_a : \psi \neq 0$). These tests are provided by the function `likTests()`, which operates on an object of class ‘`glarma`’. The likelihood ratio test compares the likelihood of the fitted GLARMA model with the likelihood of the GLM model with the same regression structure. The same null hypothesis applies to the Wald test, which is based on the Wald statistic defined in (24). Values of both statistics are compared to the chi-squared distribution with degrees of freedom given by the number of ARMA parameters. These degrees of freedom and associated chi-squared p values are correct under the situations discussed in Section 3.4.

Package users may also construct their own tailor-made likelihood ratio tests by using the reported log-likelihood (`logLik()`) for the two models under comparison and Wald tests W^2 in (24) using the appropriate submatrix of the reported estimated covariance matrix in (23) available as `glarmamod$cov`.

5.2. Probability integral transformation

To examine the validity of the assumed distribution in the GLARMA model a number of authors have suggested the use of the probability integral transformation (PIT), see for example [Czado, Gneiting, and Held \(2009\)](#). Although PIT applies to continuous distributions and the distributions in GLARMA models are discrete, [Czado *et al.* \(2009\)](#) have provided a non-randomized approach which has been implemented in the `glarma` package. There are four functions involved: `glarmaPredProb()` calculates conditional predictive probabilities; `glarmaPIT()` calculates the non-randomized PIT; `histPIT()` plots a histogram of the PIT; and `qqPIT()` draws a Q-Q plot of the PIT. If the distribution selected for the model is correct, then the histogram and Q-Q plot should resemble the histogram and Q-Q plot obtained when sampling from the uniform distribution on $[0, 1]$. Of the two plots, the histogram is generally more revealing. Deviations from the expected form of the Q-Q plot are often difficult to discern.

To calculate the conditional predictive probabilities and the PIT the following formulae from [Czado *et al.* \(2009\)](#) are used. Given the counts $\{y_t\}$, the conditional predictive probability function $F^{(t)}(\cdot|y_t)$ is given by

$$F^{(t)}(u|y_t) = \begin{cases} 0, & u \leq F(y_t - 1), \\ \frac{u - F(y_t - 1)}{F(y_t) - F(y_t - 1)}, & F(y_t - 1) \leq u \leq F(y_t), \\ 1, & u > F(y_t). \end{cases} \quad (27)$$

Here $F(y_t)$ and $F(y_t - 1)$ are the upper and lower conditional predictive probabilities respectively.

Then the non-randomized PIT is defined as

$$\bar{F}(u) = \frac{1}{n-1} \sum_{t=2}^n F^{(t)}(u|y_t). \quad (28)$$

To draw the PIT histogram, the number of bins, I , is chosen, then the height of the i th bin is

$$f_i = \bar{F}\left(\frac{i}{I}\right) - \bar{F}\left(\frac{i-1}{I}\right). \quad (29)$$

The default number of bins in `histPIT()` is 10. To help with assessment of the distribution, a horizontal line is drawn on the histogram at height 1, representing the density function of the uniform distribution on $[0, 1]$.

The Q-Q plot of the PIT plots $\bar{F}(u)$ against u , for $u \in [0, 1]$. The quantile function of the uniform distribution on $[0, 1]$ is also drawn on the plot for reference.

Jung and Tremayne (2011) employ the above diagnostics as well as the randomized version of PIT residuals to compare alternative competing count time series models for several data sets.

One can also define the normalized conditional (randomized) quantile residuals of Dunn and Smyth (1996) as $r_t = \Phi^{-1}(u_t)$ where Φ^{-1} is the inverse standard normal distribution function and u_t is a uniform random variable on the interval $[F(y_t - 1), F(Y_t)]$ defined as in (27). Benjamin *et al.* (2003) advocate the use of autocorrelation and partial autocorrelation function and associated portmanteau statistics (e.g., the Box-Ljung statistic) of these r_t to assess the adequacy of the serial dependence components in a GARMA model. This idea can easily be implemented using `glarmaPredProb()` described above. Earlier, Berkowitz (2001) suggested using a likelihood ratio test that the r_t are mean zero, variance one and independent in particular and, more generally, for other ways in which dependence may not have been modeled adequately.

5.3. Plots

The plot method for objects of class ‘`glarma`’ produces six plots by default: a time series plot with the observed values of the dependent variable, the fixed effects fit, and the GLARMA fit; a plot of the residuals against time; a histogram of the uniform PIT values; a histogram of the normal randomized residuals; a Q-Q plot of the normal randomized residuals; and the autocorrelation of the normal randomized residuals. Additional four plots can be produced: the autocorrelation of the residuals; a Q-Q plot of the residuals; a Q-Q plot of the uniform PIT values; and the partial autocorrelation of the normal randomized residuals. Any subset of these ten plots can be obtained using the `which` argument. For example the default value of `which` is set to `which = c(1L, 3L, 5L, 7L, 8L, 9L)`. Arguments to the plot method are also provided to change properties of the lines in these plots, namely line types, widths, and colors.

6. Examples

There are several example data sets included in the **glarma** package which cover binary, binomial, Poisson and negative binomial responses. Sample analyses for all these data sets are provided in either the help pages for the data sets or for the `glarma()` function.

GLARMA models with Poisson counts have appeared previously in the literature, however analyses using the binomial and negative binomial distributions are novel, so we concentrate on those cases in this section.

6.1. Asthma data

This data set arose from a single hospital (at Campbelltown), as part of a larger study into the relationship between atmospheric pollution and the number of asthma cases presenting at emergency departments in the South West region of Sydney, Australia, see [Davis *et al.* \(2003\)](#). A description of the columns in the data set is given in Table 2.

We fit a model with a moving average term at lag 7 with negative binomial counts. The initial values of the regression coefficients are found by fitting the corresponding GLM model, and the initial value of the shape parameter, α , of the negative binomial distribution is taken as 0. Pearson residuals are used and fitting is by Newton-Raphson.

```
R> library("glarma")
R> data("Asthma", package = "glarma")
R> y <- Asthma[, 1]
R> X <- as.matrix(Asthma[, 2:16])
R> glarmamod <- glarma(y, X, thetaLags = 7, type = "NegBin", method = "NR",
+   alphaInit = 0, maxit = 100, grad = 1e-6)
R> glarmamod
```

```
Call: glarma(y = y, X = X, type = "NegBin", method = "NR", thetaLags = 7,
  alphaInit = 0, maxit = 100, grad = 1e-06)
```

Negative Binomial Parameter:

```
alpha
37.19
```

Column	Variable	Description
1	Count	Daily asthma counts.
2	Intercept	Vector of 1s.
3	Sunday	Dummy variable for Sundays.
4	Monday	Dummy variable for Mondays.
5	CosAnnual	$\cos(2\pi t/365)$, annual cosine term.
6	SinAnnual	$\sin(2\pi t/365)$, annual sine term.
7	H7	Scaled, lagged and smoothed humidity.
8	NO2max	Maximum daily nitrogen oxide.
9–16	T1.1990–T2.1993	Smooth shapes to capture school terms in each year.

Table 2: The asthma data set.

GLARMA Coefficients:

theta_7
0.04392

Linear Model Coefficients:

Intercept	Sunday	Monday	CosAnnual	SinAnnual
0.58397	0.19455	0.22999	-0.21450	0.17728
H7	NO2max	T1.1990	T2.1990	T1.1991
0.16843	-0.10404	0.19903	0.13087	0.08587
T2.1991	T1.1992	T2.1992	T1.1993	T2.1993
0.17082	0.25276	0.30572	0.43607	0.11412

Degrees of Freedom: 1460 Total (i.e. Null); 1444 Residual

Null Deviance: 1990

Residual Deviance: 1443

AIC: 4874

R> summary(glarmamod)

Call: *glarma*(y = y, X = X, type = "NegBin", method = "NR", thetaLags = 7,
alphaInit = 0, maxit = 100, grad = 1e-06)

Pearson Residuals:

Min	1Q	Median	3Q	Max
-1.849	-0.741	-0.175	0.609	6.178

Negative Binomial Parameter:

	Estimate	Std.Error	z-ratio	Pr(> z)
alpha	37.2	25.4	1.46	0.14

GLARMA Coefficients:

	Estimate	Std.Error	z-ratio	Pr(> z)
theta_7	0.0439	0.0194	2.27	0.023 *

Linear Model Coefficients:

	Estimate	Std.Error	z-ratio	Pr(> z)
Intercept	0.5840	0.0633	9.22	< 2e-16 ***
Sunday	0.1946	0.0576	3.38	0.00073 ***
Monday	0.2300	0.0564	4.08	4.6e-05 ***
CosAnnual	-0.2145	0.0397	-5.41	6.3e-08 ***
SinAnnual	0.1773	0.0415	4.27	2.0e-05 ***
H7	0.1684	0.0563	2.99	0.00279 **
NO2max	-0.1040	0.0339	-3.07	0.00216 **
T1.1990	0.1990	0.0585	3.41	0.00066 ***
T2.1990	0.1309	0.0590	2.22	0.02648 *
T1.1991	0.0859	0.0675	1.27	0.20306

```

T2.1991      0.1708      0.0595      2.87  0.00410 **
T1.1992      0.2528      0.0567      4.46  8.2e-06 ***
T2.1992      0.3057      0.0510      5.99  2.1e-09 ***
T1.1993      0.4361      0.0523      8.33  < 2e-16 ***
T2.1993      0.1141      0.0627      1.82  0.06868 .

```

```

Null deviance: 1989.9 on 1460 degrees of freedom
Residual deviance: 1442.6 on 1444 degrees of freedom
AIC: 4874

```

```
Number of Newton Raphson iterations: 6
```

```
LRT and Wald Test:
```

```
Alternative hypothesis: model is a GLARMA process
```

```
Null hypothesis: model is a GLM with the same regression structure
```

```

          Statistic p-value
LR Test          7.05 0.0079 **
Wald Test         5.15 0.0233 *

```

```
---
```

```
Signif. codes:
```

```
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We note that virtually all the regression terms in the model are significant, most being highly significant. The moving average term is significant and both tests indicate that there is a need to fit a GLARMA model rather than a simple GLM. The value of α is quite large, suggesting that a Poisson model might provide adequate fit.

The plot method for an object of class 'glarma' shows six plots by default, as explained previously. As an example, in Figure 1, we show just four of these plots. Since the default title for the PIT histogram is too long for the available space we use the `titles` argument to abbreviate it.

```

R> plot(glarmamod, which = c(1, 2, 3, 5),
+       titles = list(NULL, NULL, NULL, "PIT for GLARMA (Neg. Binomial)"))

```

The ACF plot indicates that the model has dealt adequately with any serial correlation present, and the PIT histogram suggests that the negative binomial distribution provides a suitable model for the counts.

6.2. Court conviction data

This data set records monthly counts of charges laid and convictions made in Local Court and Higher Court in armed robbery in New South Wales, Australia, from 1995–2007, see [Dunsmuir, Tran, and Weatherburn \(2008\)](#). A description of the columns in the data set is given in Table 3.

The first step is to set up dummy variables for months.

```

R> data("RobberyConvict", package = "glarma")
R> datalen <- dim(RobberyConvict)[1]

```

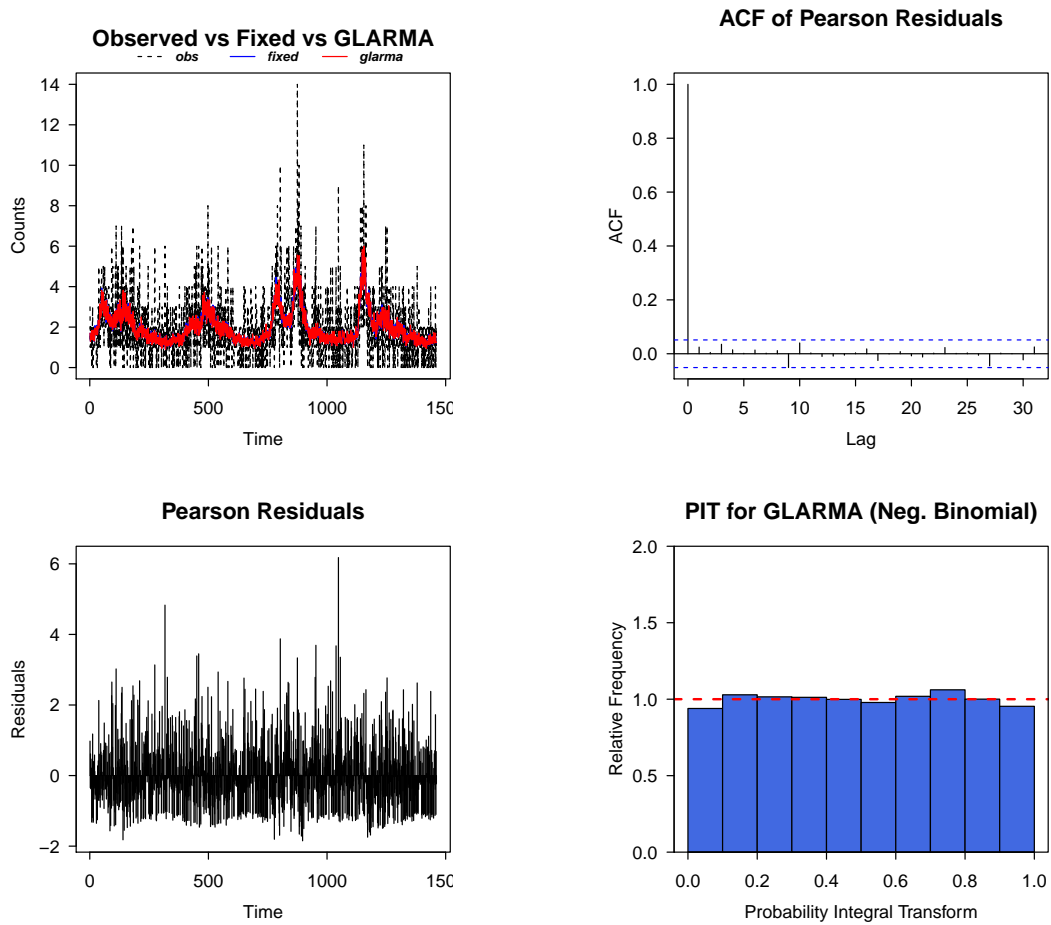


Figure 1: Diagnostic plots for the asthma model.

Column	Variable	Description
1	Date	Date in month/year format.
2	Incpt	Vector of 1s.
3	Trend	Scaled time trend.
4	Step.2001	Step change from 2001 onwards.
5	Trend.2001	Change in trend from 2001 onwards.
6	HC.N	Monthly number of cases, Higher Court.
7	HC.Y	Monthly number of convictions, Higher Court.
8	HC.P	Monthly proportion of convictions, Higher Court.
9	LC.N	Monthly number of cases, Lower Court.
10	LC.Y	Monthly number of convictions, Lower Court.
11	LC.P	Monthly proportion of convictions, Lower Court.

Table 3: The court conviction data set.

```
R> monthmat <- matrix(0, nrow = datalen, ncol = 12)
R> months <- unique(months(strptime(RobberyConvict$Date,
+   format = "%m/%d/%Y"), abbreviate = TRUE))
```

```
R> dimnames(monthmat) <- list(NULL, months)
R> for (j in 1:12) {
+   monthmat[months(strptime(RobberyConvict$Date, "%m/%d/%Y"),
+   abbreviate = TRUE) == months[j], j] <-1
+ }
R> RobberyConvict <- cbind(rep(1, datalen), RobberyConvict, monthmat)
```

Similar analyses can be carried out for both the Lower Court and the Higher Court data. Here we consider only the Lower Court data. The ARIMA component of the model is chosen to be AR(1) and the model for the conviction counts is binomial. A GLM is fitted first to obtain an initial value for the regression coefficients. The initial value of the AR parameter is set at 0. Pearson residuals are used with Newton-Raphson iteration.

First of all the data is prepared for fitting a binomial and the initial GLM fit is obtained.

```
R> y1 <- RobberyConvict$LC.Y; n1 <- RobberyConvict$LC.N
R> Y <- cbind(y1, n1 - y1)
R> head(Y, 5)
```

```
      y1
[1,]  3 9
[2,]  3 8
[3,]  6 9
[4,]  6 9
[5,]  6 5
```

```
R> glm.LCRobbery <- glm(Y ~ Step.2001 +
+   I(Feb + Mar + Apr + May + Jun + Jul) + I(Aug + Sep + Oct + Nov + Dec),
+   data = RobberyConvict, family = binomial(link = logit),
+   na.action = na.omit, x = TRUE)
R> summary(glm.LCRobbery, corr = FALSE)
```

Call:

```
glm(formula = Y ~ Step.2001 + I(Feb + Mar + Apr + May + Jun +
  Jul) + I(Aug + Sep + Oct + Nov + Dec), family = binomial(link = logit),
  data = RobberyConvict, na.action = na.omit, x = TRUE)
```

Deviance Residuals:

```
      Min       1Q   Median       3Q      Max
-2.543  -0.898   0.168   0.801   2.650
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.2568	0.1561	-1.65	0.0998 .
Step.2001	0.8232	0.0813	10.12	<2e-16 ***
I(Feb + Mar + Apr + May + Jun + Jul)	-0.3723	0.1619	-2.30	0.0215 *
I(Aug + Sep + Oct + Nov + Dec)	-0.5007	0.1655	-3.03	0.0025 **

Signif. codes:

0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 327.48 on 149 degrees of freedom
Residual deviance: 212.12 on 146 degrees of freedom
AIC: 684.8

Number of Fisher Scoring iterations: 4

Then the GLARMA model is fitted.

```
R> X <- glm.LCRobbery$x
R> colnames(X)[3:4] <- c("Feb-Jul", "Aug-Dec")
R> head(X, 5)
```

	(Intercept)	Step.2001	Feb-Jul	Aug-Dec
1	1	0	0	0
2	1	0	1	0
3	1	0	1	0
4	1	0	1	0
5	1	0	1	0

```
R> glarmamod <- glarma(Y, X, phiLags = 1, type = "Bin", method = "NR",
+   maxit = 100, grad = 1e-6)
R> summary(glarmamod)
```

```
Call: glarma(y = Y, X = X, type = "Bin", method = "NR", phiLags = 1,
  maxit = 100, grad = 1e-06)
```

Pearson Residuals:

	Min	1Q	Median	3Q	Max
	-2.446	-0.816	0.134	0.730	2.480

GLARMA Coefficients:

	Estimate	Std.Error	z-ratio	Pr(> z)
phi_1	0.0818	0.0330	2.48	0.013 *

Linear Model Coefficients:

	Estimate	Std.Error	z-ratio	Pr(> z)
(Intercept)	-0.2747	0.1571	-1.75	0.0804 .
Step.2001	0.8220	0.0957	8.59	<2e-16 ***
Feb-Jul	-0.3568	0.1598	-2.23	0.0256 *
Aug-Dec	-0.5004	0.1633	-3.06	0.0022 **

Null deviance: 327.48 on 149 degrees of freedom


```
Residual deviance: 198.91 on 145 degrees of freedom
AIC: 680.7
```

```
Number of Newton Raphson iterations: 4
```

```
LRT and Wald Test:
```

```
Alternative hypothesis: model is a GLARMA process
```

```
Null hypothesis: model is a GLM with the same regression structure
```

```
      Statistic p-value
LR Test      6.11  0.013 *
Wald Test    6.14  0.013 *
```

```
---
```

```
Signif. codes:
```

```
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We observe that the regression coefficients for the GLARMA model are quite similar to those for the GLM model. In particular, the step change in 2001 is highly significant. The likelihood ratio and Wald tests both suggest the need to deal with autocorrelation. Figure 2 is produced using `plot(glarmamod)`.

In these diagnostic plots, the ACF plot of the randomized residuals shows little residual autocorrelation, and the Q-Q plot of the randomized residuals shows reasonable conformity with normality. However the PIT histogram suggests that the binomial model for the counts is not appropriate for this data.

6.3. Example of diagnostics based on normalized randomized residuals

We illustrate the normalized randomized quantile residuals on the court conviction data below. The function `normRandPIT()` produces the randomized PIT residuals.

Figure 3 shows the utility of these residuals when the serial dependence term is omitted from the model. We first fit the model and obtain the randomized residuals.

```
R> glarmamod <- glarma(Y, X, type = "Bin", method = "NR", maxit = 100,
+   grad = 1e-6)
R> rt <- normRandPIT(glarmamod)$rt
```

Figure 3 is produced using `plot(glarmamod, which = 7:10)`.

The Box-Ljung portmanteau test results are:

```
R> Box.test(rt, lag = 12, type = "Ljung-Box", fitdf = 0)
```

```
Box-Ljung test
```

```
data: rt
```

```
X-squared = 24.498, df = 12, p-value = 0.01739
```

```
R> Box.test(glarmamod$residuals, lag = 12, type = "Ljung-Box", fitdf = 0)
```

```
Box-Ljung test
```

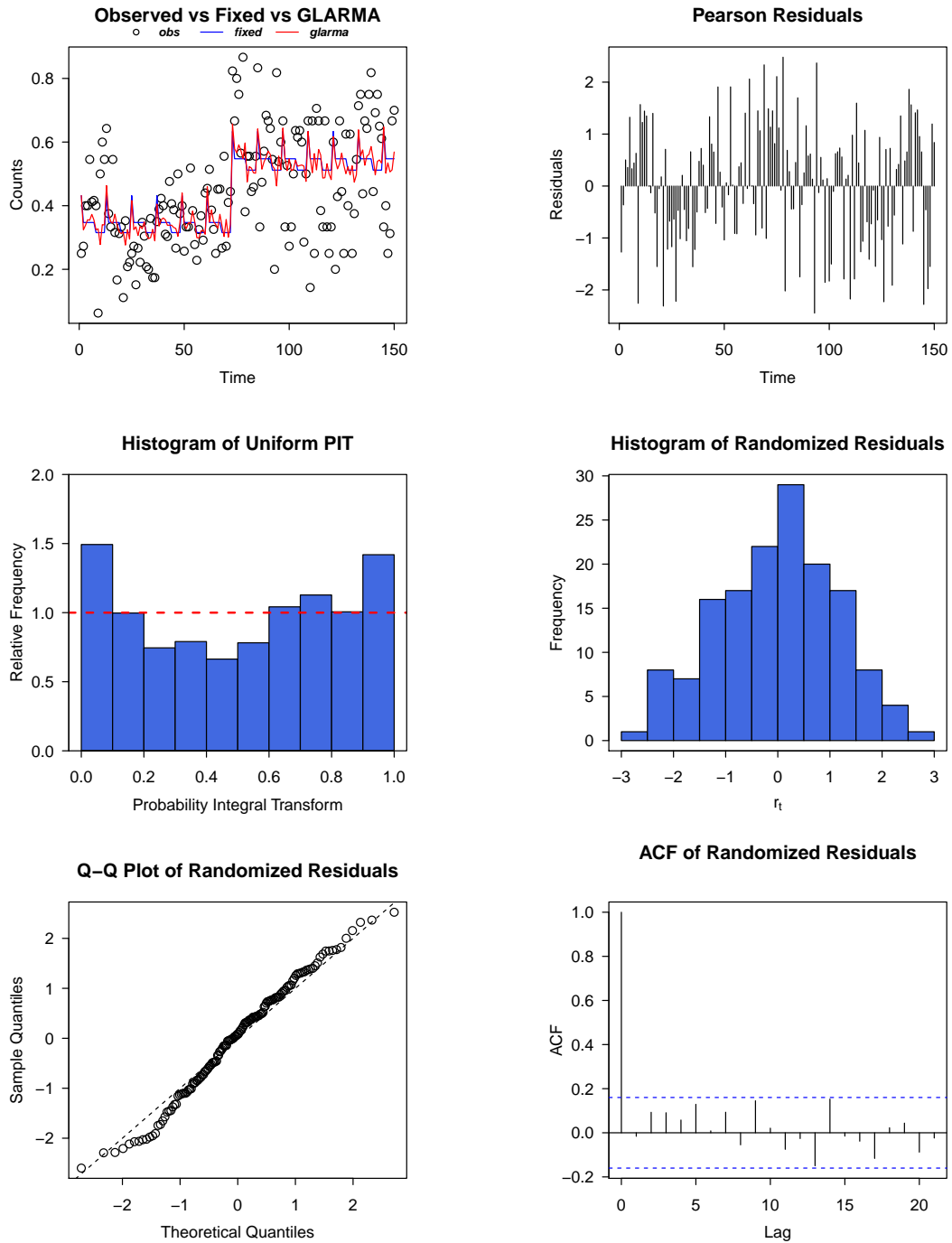


Figure 2: Diagnostic plots for the court conviction model.

```
data: glarmamod$residuals
X-squared = 23.16, df = 12, p-value = 0.02636
```

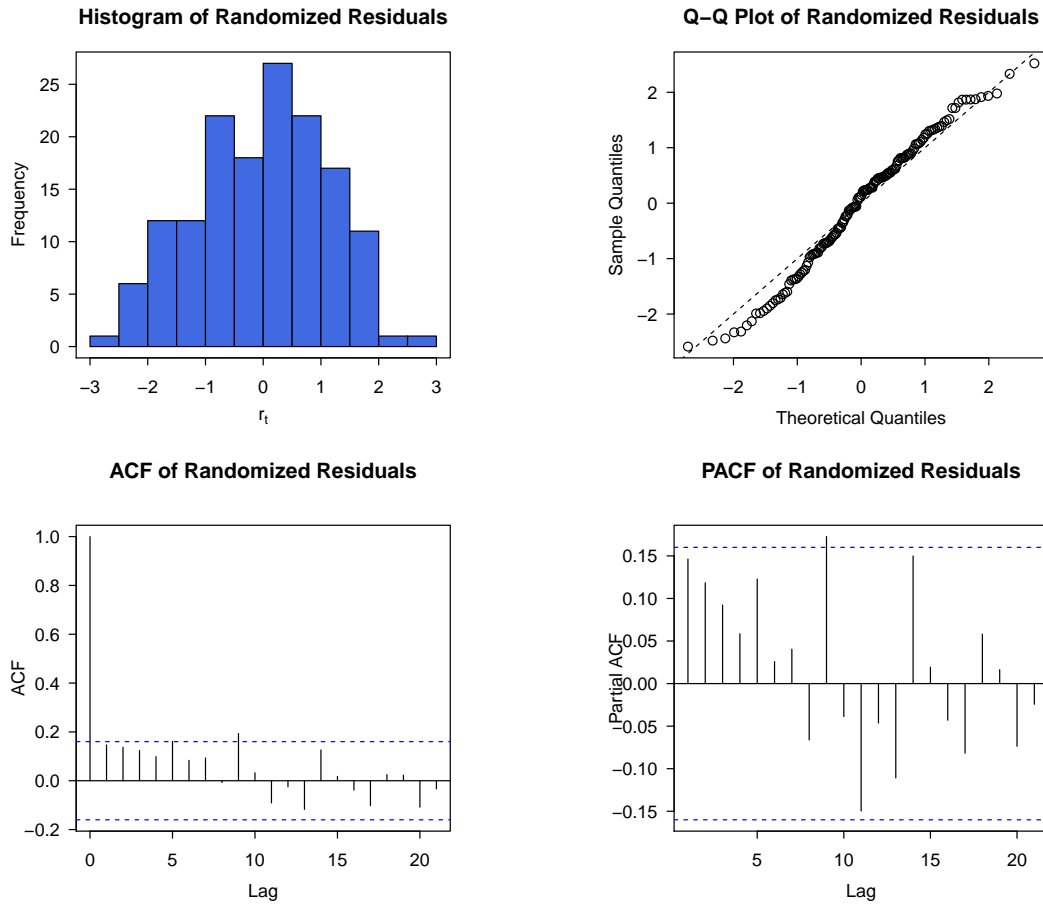


Figure 3: Diagnostic plots for the court conviction model.

6.4. Example of forecasting with GLARMA models

Forecasting for GLARMA models is implemented in the **glarma** package via the `forecast` method for 'glarma' objects. In the following example we first produce a series of one-step ahead forecasts of μ_t and plot them along with the estimated values of μ_t obtained by using all the data. We first obtain the estimates and predicted values.

```
R> library("zoo")
```

```
Loading required package: zoo
```

```
Attaching package: "zoo"
```

```
The following objects are masked from "package:base":
```

```
as.Date, as.Date.numeric
```

```
R> data("DriverDeaths", package = "glarma")
```

```

R> y <- DriverDeaths[, "Deaths"]
R> X <- as.matrix(DriverDeaths[, 2:5])
R> Population <- DriverDeaths[, "Population"]
R> glarmamod <- glarma(y, X, offset = log(Population/100000),
+   phiLags = 12, thetaLags = 1, type = "Poi", method = "FS",
+   residuals = "Pearson", maxit = 100, grad = 1e-6)
R> allX <- X
R> allFits <- fitted(glarmamod)
R> ally <- y
R> forecasts <- numeric(72)
R> for (i in 62:71) {
+   y <- DriverDeaths[1:i, "Deaths"]
+   X <- as.matrix(DriverDeaths[1:i, 2:5])
+   Population <- DriverDeaths[1:i, "Population"]
+   glarmamod <- glarma(y, X, offset = log(Population/100000),
+     phiLags = 12, thetaLags = 1, type = "Poi", method = "FS",
+     residuals = "Pearson", maxit = 100, grad = 1e-6)
+   XT1 <- matrix(allX[i + 1, ], nrow = 1)
+   offsetT1 <- log(DriverDeaths$Population[i + 1]/100000)
+   mu <- forecast(glarmamod, 1, XT1, offsetT1)$mu
+   if (i == 62) {
+     forecasts[1:62] <- fitted(glarmamod)
+   }
+   forecasts[i+1] <- mu
+ }

```

Next we produce the plot shown in Figure 4. The code to produce the plot is not shown, but can be found in the supplementary material.

For forecasting more than one-step ahead, sampling is required as explained previously. We continue the example above by producing a sample of possible values of μ_t for the last time period obtained by predicting two steps ahead.

```

R> glarmamod <- glarma(y[1:70], X[1:70, ],
+   offset = log(Population/100000)[1:70], phiLags = 12, thetaLags = 1,
+   type = "Poi", method = "FS", residuals = "Pearson", maxit = 100,
+   grad = 1e-6)
R> nObs <- NROW(X)
R> n.ahead <- 2
R> XT1 <- as.matrix(X[(nObs - n.ahead + 1):nObs, ])
R> offsetT1 <- log(Population/100000)[(nObs - n.ahead + 1):nObs]
R> nSims <- 500
R> forecastMu <- forecastY <- matrix(ncol = n.ahead, nrow = nSims)
R> for (i in 1:nSims) {
+   temp <- forecast(glarmamod, n.ahead, XT1, offsetT1)
+   forecastY[i, ] <- temp$Y
+   forecastMu[i, ] <- temp$mu
+ }

```

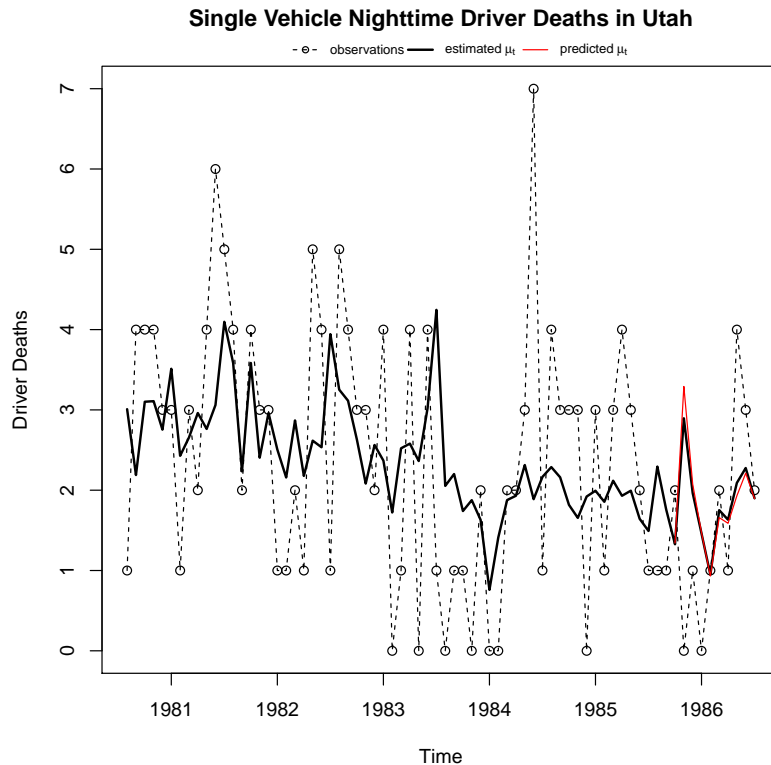


Figure 4: Successive one-step ahead predictions for driver deaths.

```
R> table(forecastY[, 2])
```

0	1	2	3	4	5	6	7	8
54	129	131	93	53	22	12	5	1

The distribution of the sample values is shown in Figure 5. The code to produce the plot is not shown.

7. Other packages for observation-driven models

A number of other packages on CRAN are available which use various approaches to fit observation-driven models for the type of data modeled using the **glarma** package. We here discuss the packages detected using searches for functions via the package **sos** [Graves, Dorai-Raj, and Francois \(2013\)](#). The keywords used for these searches were “GLARMA”, “GARMA” and “polio”, the last of these because the polio data set of [Zeger \(1988\)](#) is the classical example considered by most researchers concerned with analyzing this type of data.

Searching using the keyword “GLARMA” only produced references to the **glarma** package. Searching using the keyword “GARMA” produced references to the packages **VGAM** ([Yee and Wild 1996](#); [Yee 2010, 2015](#)), **gamlss.util** ([Stasinopoulos, Rigby, and Eilers 2015](#)) and **gsarima** ([Briet 2014](#)).

VGAM has a function `garma()` which fits a GARMA model. The help file indicates that the current version of the function is only a preliminary version, viz “Currently, this function

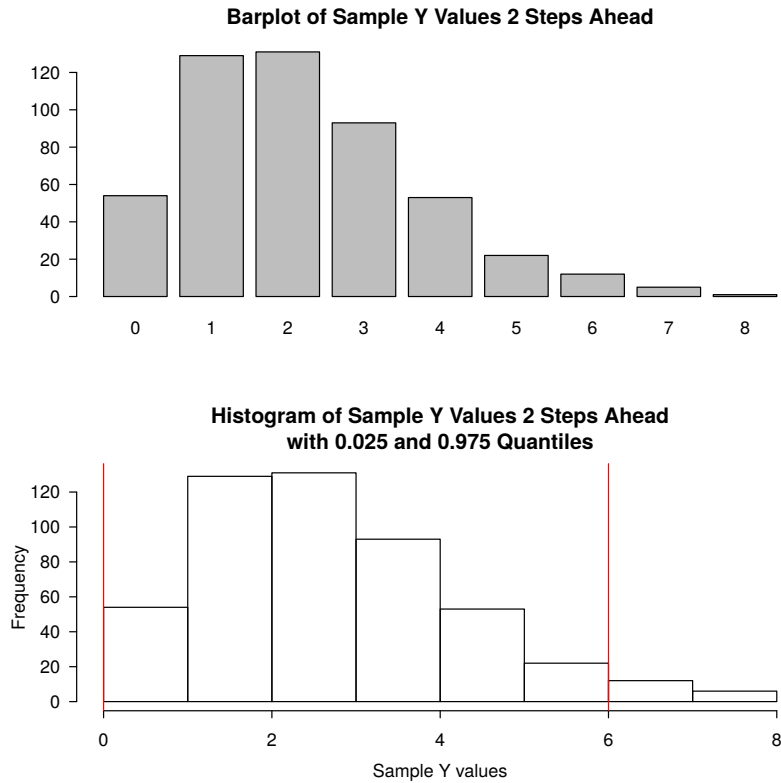


Figure 5: Distribution of sample values from forecasting two steps ahead.

is rudimentary and can handle only certain continuous, count and binary responses only”. Also “This function is unpolished and is (sic) requires lots of improvements. In particular, initialization is very poor. Results appear very sensitive to quality of initial values.” Fitting is by Fisher scoring, and Pearson residuals are produced. There is no plot method. One example is given, the interspike data from [Zeger and Qaqish \(1988\)](#) and the model fitted is an AR(2), with no regression component other than the overall mean.

The package `gamlss.util`, which is an extension to `gamlss` ([Rigby and Stasinopoulos 2005](#); [Stasinopoulos and Rigby 2007](#)) has a `garmaFit()` function. As an example the Polio data of [Zeger \(1988\)](#) is modeled. The word “GARMA” does not appear in the main references to the `gamlss` package, so further information about the function must be sought in [Benjamin *et al.* \(2003\)](#) and the code of the function itself. The work in [Benjamin *et al.* \(2003\)](#) has already been discussed.

The `garmaFit()` function has no `offset` argument, although it seems that an offset can be included using the formula specification of the model. The ARMA specification only allows for specification of the maximum AR and MA orders, so specifying that the model has MA terms 1, 2 and 5 only (as in the GLARMA polio analysis) does not appear possible. The range of possible response distributions is large and different link functions can also be used. The residuals used are identity residuals. There are `print`, `summary`, `fitted`, `coef`, `residuals`, `update`, `plot`, `deviance`, and `formula` methods for the object returned by `garmaFit()`. There is a `predict` method, but the help page states it is under development. Overall the function `garmaFit()` and associated methods provide a useful modeling capability, although we have

misgivings about the methodology, as we have outlined in Section 3.

The package **gsarima** only provides code for simulating some GLMs which have an autoregressive component.

Searching on “polio” produced two further packages on CRAN which analyze the polio data: **acp** (autoregressive conditional Poisson, Vasileios 2015); and **gcmr** (Gaussian copula marginal regression, Masarotto and Varin 2015). The theory behind these two packages may be found in Heinen (2003) and Masarotto and Varin (2012), respectively. A further package found in this search was **mr** (marginal regression, Masarotto and Varin 2010). This package is archived, but appears to have been superseded by **gcmr**.

Heinen (2003) only deals with the ARMA(1,1) structure for modeling the lack of independence. Besides the Poisson distribution, two versions of the double Poisson distribution are considered for the marginal distribution of the counts. Heinen (2003) also examines models where the variance has an ARMA(1,1) structure. There are examples in the paper for these models, but package **acp** only fits the ACP model, not the double Poisson or other enhancements.

Masarotto and Varin (2012) use multivariate Gaussian copula regression to model GLMs with time series errors. They use maximum likelihood to fit the parameters of their models but due to the complexity of their approach, a Monte Carlo approximation of the likelihood with importance sampling is used for model fitting. In the package **gcmr** fitting uses the function `gcmr()`. There are diagnostics for the model available, including examination of the residuals. Analysis of the polio data is carried out as an example for the package capability. The model specification only allows for all lag orders to be included when errors with given orders are specified. Marginal distributions can be beta, binomial, gamma, normal, negative binomial, Poisson or Weibull.

8. Future enhancements and applications

The **glarma** package can be further developed to expand the range of response distributions and forms of state equation. In relation to the state equation the development of transfer function type models (along the lines of Box, Jenkins, and Reinsel 2013) and the inclusion of intervention models would require replacing the linear regression component $x_t^T \beta$ by a transfer function which is non-linear in the parameters. Such an enhancement would create useful models for policy assessment in fields such as criminology.

As regards increasing the range of response distributions available three suggest themselves as being of considerable practical utility. The first is the gamma response (for continuous outcomes) or the generalized gamma. These distributions would allow modeling durations between events as arise in financial modeling. We have developed the GLARMA methods for these which, however, are currently not implemented in the **glarma** package but could be easily enough. The second is inclusion of zero-inflated distributions. This could currently be accommodated to a limited extent by modeling the probability of a zero occurring with a binary process and then modeling the probability of a positive response with a separate count distribution. The model could be conditionally specified in a similar vein to that used for model transaction level stock price movements as in Rydberg and Shephard (2003). A more general model would require both the zero-inflated component and the count response component to be modeled jointly in terms of possibly coupled state equations or bivariate

state equations. This would require a substantial additional development of the current **glarma** package.

The third is the multinomial response distribution which would be useful in financial modeling with, for example, the hurdle model used in Liesenfeld, Nolte, and Pohlmeier (2006) and in modeling listener responses to music as in Dean, Bailes, and Dunsmuir (2014). Again development of this in a complete way would require a multivariate state equation to be used and this would, as in the zero-inflated case, require substantial development.

The **glarma** package has already been utilized in modeling multiple independent time series of binary responses (Dean *et al.* 2014) and in modeling the impact of changes in legal blood alcohol levels on driver deaths (Dunsmuir 2015). Data sets of this type, which we call “long longitudinal data”, are increasingly needed to be analyzed. Dunsmuir (2015) outlines a fixed effects approach and a random effects approach (random effects being used to explain differences between regression parameters in the individual series). Currently to implement these additional R code is required, but it is planned to make this available in future releases of the **glarma** package.

Acknowledgments

The authors thank the two referees for their insightful and helpful suggestions for improving the **glarma** package and this paper.

Previous students Haolan Lu and Bo Wang contributed to modules in the **glarma** package and Daniel Drescher and Cuong Tran to early package design. Cenanning Li was responsible for coding a number of functions in the **glarma** package, for much of the package documentation and assembling the first version of the package.

References

- Benjamin M, Rigby R, Stasinopoulos D (2003). “Generalized Autoregressive Moving Average Models.” *Journal of the American Statistical Association*, **98**(461), 214–223. doi:10.1198/016214503388619238.
- Berkowitz J (2001). “Testing Density Forecasts, with Applications to Risk Management.” *Journal of Business & Economic Statistics*, **19**(4), 465–474. doi:10.1198/07350010152596718.
- Box G, Jenkins G, Reinsel G (2013). *Time Series Analysis: Forecasting and Control*. John Wiley & Sons.
- Briet O (2014). **gsarima**: *Two Functions for Generalized SARIMA Time Series Simulation*. R package version 0.1-4, URL <http://CRAN.R-project.org/package=gsarima>.
- Brockwell P, Davis R (2010). *Introduction to Time Series and Forecasting*. 2nd edition. Springer-Verlag, New York.
- Cox D (1958). “The Regression Analysis of Binary Sequences.” *Journal of the Royal Statistical Society B*, **20**(2), 215–242.

- Cox D (1981). “Statistical Analysis of Time Series: Some Recent Developments.” *Scandinavian Journal of Statistics*, **8**(2), 93–115.
- Creal D, Koopman S, Lucas A (2008). “A General Framework for Observation Driven Time-Varying Parameter Models.” *Discussion paper*, Tinbergen Institute.
- Czado C, Gneiting T, Held L (2009). “Predictive Model Assessment for Count Data.” *Biometrics*, **65**(4), 1254–1261. doi:10.1111/j.1541-0420.2009.01191.x.
- Davis R, Dunsmuir W (2015). “State Space Models for Count Time Series.” In R Davis, S Holan, R Lund, N Ravishanker (eds.), *Handbook of Discrete-Valued Time Series*. CRC Monographs.
- Davis R, Dunsmuir W, Streett S (2003). “Observation-Driven Models for Poisson Counts.” *Biometrika*, **90**(4), 777–790. doi:10.1093/biomet/90.4.777.
- Davis R, Dunsmuir W, Streett S (2005). “Maximum Likelihood Estimation for an Observation Driven Model for Poisson Counts.” *Methodology and Computing in Applied Probability*, **7**(2), 149–159. doi:10.1007/s11009-005-1480-4.
- Davis R, Dunsmuir W, Wang Y (1999). “Modeling Time Series of Count Data.” In S Ghosh (ed.), *Asymptotics, Nonparametrics, and Time Series*, volume 158 of *Statistics Textbooks and Monographs*, pp. 63–114. Marcel Dekker, New York, NY.
- Davis R, Liu H (2012). “Theory and Inference for a Class of Observation-Driven Models with Application to Time Series of Counts.” arXiv:1204.3915 [math.ST], URL <http://arxiv.org/abs/1204.3915>.
- Davis R, Wu R (2009). “A Negative Binomial Model for Time Series of Counts.” *Biometrika*, **96**(3), 735–749. doi:10.1093/biomet/asp029.
- Dean R, Bailes F, Dunsmuir W (2014). “Time Series Analysis of Real-Time Music Perception: Approaches to the Assessment of Individual and Expertise Differences in Perception of Expressed Affect.” *Journal of Mathematics and Music*, **8**(3), 183–205. doi:10.1080/17459737.2014.928752.
- Diggle P, Heagerty P, Liang KY, Zeger S (2002). *Analysis of Longitudinal Data*. 2nd edition. Oxford University Press, Oxford. doi:10.1002/pst.112.
- Dunn P, Smyth G (1996). “Randomized Quantile Residuals.” *Journal of Computational and Graphical Statistics*, **5**(3), 236–244. doi:10.1080/10618600.1996.10474708.
- Dunsmuir W (2015). “Generalized Linear Autoregressive Moving Average Models.” In R Davis, S Holan, R Lund, N Ravishanker (eds.), *Handbook of Discrete-Valued Time Series*. CRC Monographs.
- Dunsmuir W, Leung J, Liu X (2004). “Extensions of Observation Driven Models for Time Series of Counts.” In B Silva, N Mukhopadhyay (eds.), *Proceedings of the International Sri Lankan Statistical Conference: Visions of Futuristic Methodologies*. RMIT University and University of Peradeniy.

- Dunsmuir W, Li C, Scott D (2015). *glarma: Generalized Linear Autoregressive Moving Average Models*. R package version 1.4-0, URL <http://CRAN.R-project.org/package=glarma>.
- Dunsmuir W, Tran C, Weatherburn D (2008). *Assessing the Impact of Mandatory DNA Testing of Prison Inmates in NSW on Clearance, Charge and Conviction Rates for Selected Crime Categories*. NSW Bureau of Crime Statistics and Research.
- Durbin J, Koopman S (2012). *Time Series Analysis by State Space Methods*. Oxford University Press, Oxford. doi:10.1093/acprof:oso/9780199641178.001.0001.
- Fokianos K, Tjøstheim D (2011). “Log-Linear Poisson Autoregression.” *Journal of Multivariate Analysis*, **102**(3), 563–578. doi:10.1016/j.jmva.2010.11.002.
- Freeland R, McCabe B (2004). “Forecasting Discrete Valued Low Count Time Series.” *International Journal of Forecasting*, **20**(3), 427–434. doi:10.1016/s0169-2070(03)00014-1.
- Graves S, Dorai-Raj S, Francois R (2013). *sos: Search Contributed R Packages, Sort by Package*. R package version 1.3-8, URL <http://CRAN.R-project.org/package=sos>.
- Hansen B (1996). “Inference When a Nuisance Parameter Is Not Identified under the Null Hypothesis.” *Econometrica*, **64**(2), 413–430. doi:10.2307/2171789.
- Heinen A (2003). “Modelling Time Series Count Data: An Autoregressive Conditional Poisson Model.” <http://mpira.ub.uni-muenchen.de/8113>.
- Jacobs P, Lewis P (1978a). “Discrete Time Series Generated by Mixtures. I: Correlational and Runs Properties.” *Journal of the Royal Statistical Society B*, **40**(1), 94–105.
- Jacobs P, Lewis P (1978b). “Discrete Time Series Generated by Mixtures II: Asymptotic Properties.” *Journal of the Royal Statistical Society B*, **40**(2), 222–228.
- Jung R, Kukuk M, Liesenfeld R (2006). “Time Series of Count Data: Modeling, Estimation and Diagnostics.” *Computational Statistics & Data Analysis*, **51**(4), 2350–2364. doi:10.1016/j.csda.2006.08.001.
- Jung R, Tremayne A (2011). “Useful Models for Time Series of Counts or Simply Wrong Ones?” *Advances in Statistical Analysis*, **95**(1), 59–91. doi:10.1007/s10182-010-0139-9.
- Kauppi H, Saikkonen P (2008). “Predicting US Recessions with Dynamic Binary Response Models.” *The Review of Economics and Statistics*, **90**(4), 777–791. doi:10.1162/rest.90.4.777.
- Kedem B, Fokianos K (2005). *Regression Models for Time Series Analysis*, volume 488. John Wiley & Sons. doi:10.1002/0471266981.
- Klingenberg B (2008). “Regression Models for Binary Time Series with Gaps.” *Computational Statistics & Data Analysis*, **52**(8), 4076–4090. doi:10.1016/j.csda.2008.01.019.
- Li W (1994). “Time Series Models Based on Generalized Linear Models: Some Further Results.” *Biometrics*, **50**(2), 506–511. doi:10.2307/2533393.

- Liesenfeld R, Nolte I, Pohlmeier W (2006). “Modelling Financial Transaction Price Movements: A Dynamic Integer Count Data Model.” *Empirical Economics*, **30**(4), 795–825. doi:10.1007/s00181-005-0001-1.
- Lu H (2002). *Observation Driven and Parameter Driven Models for Time Series of Counts*. Project report, School of Public Health, University of Minnesota, Minneapolis, MN, USA.
- Masarotto G, Varin C (2010). “**mr**: Marginal Regression Models for Dependent Data.” URL <http://CRAN.R-project.org/src/contrib/Archive/mr/>.
- Masarotto G, Varin C (2012). “Gaussian Copula Marginal Regression.” *Electronic Journal of Statistics*, **6**, 1517–1549. doi:10.1214/12-ejs721.
- Masarotto G, Varin C (2015). **gcmr**: *Gaussian Copula Marginal Regression*. R package version 0.7-5, URL <http://CRAN.R-project.org/package=gcmr>.
- McCabe B, Martin G, Harris D (2011). “Efficient Probabilistic Forecasts for Counts.” *Journal of the Royal Statistical Society B*, **73**(2), 253–272. doi:10.1111/j.1467-9868.2010.00762.x.
- McCullagh P, Nelder J (1989). *Generalized Linear Models*. Chapman and Hall, London.
- McKenzie E (1988). “Some ARMA Models for Dependent Sequences of Poisson Counts.” *Advances in Applied Probability*, **20**(4), 822–835. doi:10.2307/1427362.
- R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.
- Rigby R, Stasinopoulos D (2005). “Generalized Additive Models for Location, Scale and Shape.” *Journal of the Royal Statistical Society C*, **54**(3), 507–554. doi:10.1111/j.1467-9876.2005.00510.x.
- Rydberg T, Shephard N (2003). “Dynamics of Trade-By-Trade Price Movements: Decomposition and Models.” *Journal of Financial Econometrics*, **1**(1), 2–25. doi:10.1093/jjfinec/nbg002.
- Shephard N (1995). “Generalized Linear Autoregressions.” *Technical report*, Nuffield College, Oxford University.
- Startz R (2008). “Binomial Autoregressive Moving Average Models with an Application to U.S. Recessions.” *Journal of Business & Economic Statistics*, **26**(1), 1–8. doi:10.1198/073500107000000151.
- Stasinopoulos DM, Rigby RA (2007). “Generalized Additive Models for Location Scale and Shape (GAMLSS) in R.” *Journal of Statistical Software*, **23**(7), 1–46. doi:10.18637/jss.v023.i07.
- Stasinopoulos M, Rigby B, Eilers P (2015). **gamlss.util**: *GAMLSS Utilities*. R package version 4.3-2, URL <http://CRAN.R-project.org/package=gamlss.util>.
- Streett S (2000). *Some Observation Driven Models for Time Series of Counts*. Ph.D. thesis, Colorado State University, Department of Statistics, Fort Collins, Colorado.

- Tjøstheim D (2012). “Some Recent Theory for Autoregressive Count Time Series.” *Test*, **21**(3), 413–438. doi:10.1007/s11749-012-0296-0.
- Vasileios S (2015). *acp: Autoregressive Conditional Poisson*. R package version 2.0, URL <http://CRAN.R-project.org/package=acp>.
- Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. 4th edition. Springer-Verlag, New York.
- Wang B (2004). *GLARMA Models and Stock Price Dynamics*. Project report, School of Mathematics and Statistics, University of New South Wales, Sydney, NSW, 2051, Australia.
- Wang C, Li W (2011). “On the Autopersistence Functions and the Autopersistence Graphs of Binary Autoregressive Time Series.” *Journal of Time Series Analysis*, **32**(6), 639–646. doi:10.1111/j.1467-9892.2011.00721.x.
- Woodard D, Matteson D, Henderson S (2011). “Stationarity of Generalized Autoregressive Moving Average Models.” *Electronic Journal of Statistics*, **5**, 800–828. doi:10.1214/11-ejs627.
- Yee T (2010). “The **VGAM** Package for Categorical Data Analysis.” *Journal of Statistical Software*, **32**(10), 1–34. doi:10.18637/jss.v032.i10.
- Yee T (2015). *VGAM: Vector Generalized Linear and Additive Models*. R package version 0.9-8, URL <http://CRAN.R-project.org/package=VGAM>.
- Yee T, Wild C (1996). “Vector Generalized Additive Models.” *Journal of Royal Statistical Society B*, **58**(3), 481–493.
- Zeger S (1988). “A Regression Model for Time Series of Counts.” *Biometrika*, **75**(4), 621–629. doi:10.1093/biomet/75.4.621.
- Zeger S, Qaqish B (1988). “Markov Regression Models for Time Series: A Quasi-Likelihood Approach.” *Biometrics*, **44**(4), 1019–1031. doi:10.2307/2531732.

Affiliation:

William T. M. Dunsmuir
 Department of Statistics
 School of Mathematics and Statistics
 University of New South Wales
 Sydney, NSW, 2052, Australia
 E-mail: W.Dunsmuir@unsw.edu.au
 URL: <http://web.maths.unsw.edu.au/~dunsmuir/>

Journal of Statistical Software

published by the Foundation for Open Access Statistics

October 2015, Volume 67, Issue 7

doi:10.18637/jss.v067.i07

<http://www.jstatsoft.org/>

<http://www.foastat.org/>

Submitted: 2013-01-29

Accepted: 2014-11-03
