A Class of Models for Aggregated Traffic Volume Time Series

A.E. Brockwell, N.H. Chan and P.K. Lee

June 10, 2002

Abstract

Two time series are considered in this paper: one is the volume of hard disk activity, aggregated into half-hour periods, measured on a workstation, and the other is the volume of internet requests made to a workstation. Both of these time series exhibit features typical of network traffic data, namely, strong seasonal components and highly non-Gaussian distributions. For these time series, a particular class of nonlinear state-space models is proposed, and practical techniques for model-fitting and forecasting are demonstrated.

Keywords: censored time series, traffic volume, general state-space models, Tobit models, nonlinear dynamic models, Gibbs sampler.

1 Introduction

In this paper we consider two time series of traffic volume, one of the volume in bytes of hard disk access requests, and the other of the volume in bytes of internet server access requests, each aggregrated over half-hour time intervals. These time series exhibit strong seasonal components, and are also non-Gaussian, with a clear "saturation" effect at a lower bound.

In the engineering literature, authors such as R.H.R. Riedi and Baraniuk (1999) have used multifractal wavelet models to study similar time series. Although popular, wavelet models often fail to capture the saturation effect and the autocorrelation structure of internet traffic data. We model the time series using a class of so-called *Tobit models*, which, although well-studied in econometrics, have received much less attention in other domains. The Tobit model is a special case of a state-space model (For details on general state-space models, also known as "dynamic models", see, e.g. West and Harrison, 1997; Kitagawa and Gersch, 1996; Brockwell and Davis, 1991; Shumway and Stoffer, 2000), in which the state process is linear and Gaussian, and the observation equation simply "truncates" a linear combination of the elements of the state vector at a certain point. Because of their ability to capture saturation effects and their flexibility in capturing autocorrelation structure, Tobit models are ideal candidates for modelling traffic volume time series.

For estimation of Tobit model parameters, Lee (1999) has proposed a technique which relies on the use of simulation to approximate likelihoods. Zeger and Brookmeyer (1986) have also

developed nice methods of computing likelihoods or approximate likelihoods for the specific case of censored Gaussian autoregressions (this is a sub-class of the family of Tobit models). To avoid the problems associated with using approximate likelihoods, and to bypass the restriction of working only with autoregressive processes, we adopt a Gibbs sampling approach, quite similar to that proposed in Carlin et al. (1992), for purposes of both parameter estimation and forecasting. This approach, at a slightly higher computational cost than simulated likelihood methods, allows us to obtain better parameter estimates and forecasts.

The methods developed are readily extended to handle multivariate Tobit models, and hence could potentially be used to analyze traffic volume data collected from multiple interconnected servers.

2 The Data

The first time series was collected from a hard disk of a Hewlett Packard workstation (named *hplajw* and described in Ruemmler and Wilkes, 1993). During the time period from April 18, 1992 to Jun 27, 1992 (63 days), the size in bytes of each job read from the hard disk was recorded, together with its arrival time, transferral time and the storage location (sector). The job sizes in successive 30 minute intervals were then aggregated to form the time series shown in Figure 1. Clearly, there are occasional periods of high usage, and the volume can never be negative, effectively skewing the data to the right. The sample autocovariance function of the data shows that seasonal components are present. The sinusoidal pattern with period 48 indicates a high correlation between points that are one day apart (48 lags $\times$ 30 minutes = 1 day), and another sinusoidal pattern with period 336 indicates high correlations between points a week apart.

The second time series represents volume of HTTP (internet) requests to the WWW server in the Computer Science Department at the University of Calgary, for the period from October 30, 1994 to April 2, 1995. Again, total request volumes in bytes in successive 30 minute intervals were aggregated to form a time series. For clearer illustration, we add one to the time series and take logs. The resulting series is shown in Figure 3 and its histogram and sample autocorrelation function are given in Figure 4. The histogram shows that there is a roughly Gaussian cluster of points, mixed in with a lot of additional points hitting the minimal value of zero. The sample autocovariance indicates significant correlation structure in the data. Again there appear to be strong day-apart and week-apart correlations.

The disk-trace and HTTP-request time series are clearly non-Gaussian. They hit a minimal value frequently. At the same time, the sample autocorrelation functions exhibit significant periodic structure. In the next sections, we introduce models for these two time series which capture both of these properties.

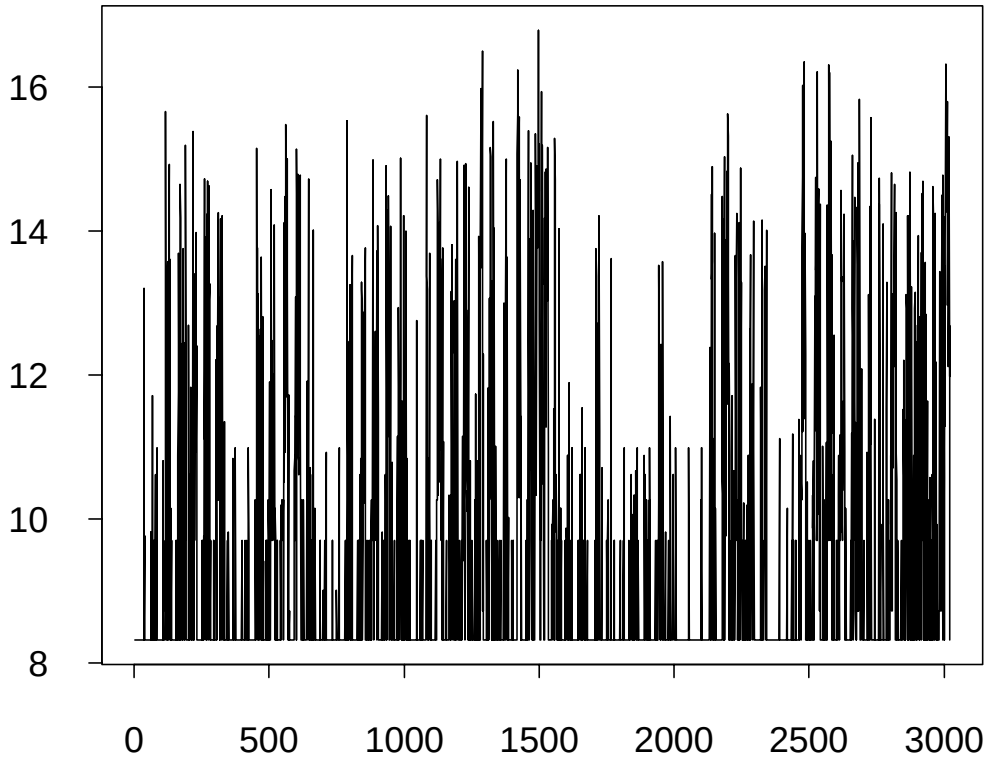

Figure 1: Log of the HP disk-trace data, aggregated in half-hour blocks.

3 The Class of Models

Motivated by the need to capture the saturation effect in the data, and also to retain a relatively large class of possible autocovariance structures, we consider the following class of models (known as Tobit models).

Let $\{X_t \in \mathbb{R}^p\}$ be a linear, causal stationary Gaussian time series with mean zero and autocovariance function $\gamma_X(h) = \mathbf{E}\left[X_t X_{t+h}^T\right]$, satisfying the state equation

$$X_{t+1} = FX_t + Z_t, \quad \{Z_t\} \sim \text{IIDN}(0, \Sigma), \tag{1}$$

where F is some $p \times p$ matrix and $\{Z_t\}$ is an independent and identically distributed sequence of multivariate normal random variables with mean (vector) zero and covariance matrix Σ. Next let

$$V_t = \sigma h^T (h^T \Lambda h)^{-1/2} X_t + \mu, \tag{2}$$

where h is a p-dimensional vector, Λ is the variance of X_t, that is, the covariance matrix satisfying $\Lambda = F\Lambda F^T + \Sigma$, and μ is a constant. V_t is constructed in this manner so that it has

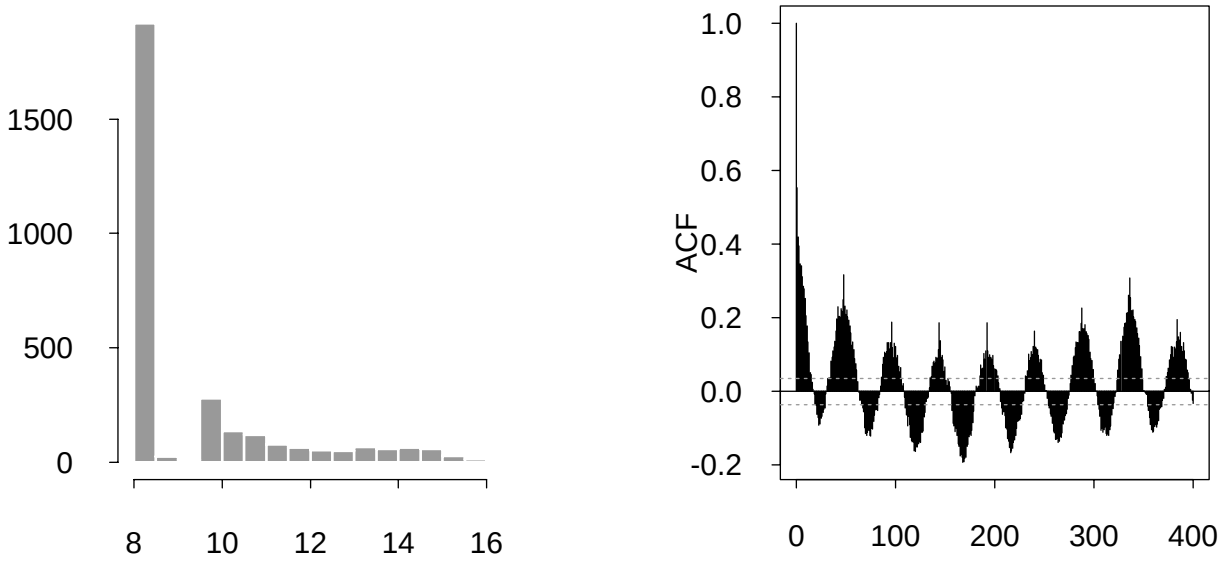

Figure 2: Log HP disk trace data: histogram and sample autocorrelation function.

mean μ and variance σ^2. Let the process $\{Y_t\}$ be defined by

$$Y_t = g(V_t), \tag{3}$$

where

$$g(x) = \begin{cases} x, & x > c \\ c, & x \leq c, \end{cases} \tag{4}$$

for some constant c. Thus each Y_t is a scaled and shifted linear combinations of the components of X_t, passed through the "censoring" function $g(\cdot)$.

The processes $\{X_t\}$ and $\{V_t\}$ are unobserved, while the process $\{Y_t\}$ is observed.

The model (1,2,3) effectively represents a generalization of the class of univariate linear Gaussian time series, since as c approaches $-\infty$, $g(\cdot)$ approaches the identity function, and its argument in (3) is simply a shifted and rescaled linear combination of the elements of the state vector. Loosely speaking, the structure of the state equation (1) determines the autocorrelation structure of the time series and the observation equation (3) determines the marginal distribution of the data. The obvious advantage of this model is its ability to capture the kind of saturation exhibited in the time series shown in Figures 1 and 3. In the remainder of this paper we show how parameter estimation and forecasting can be carried out for this class of models, using the disk-trace and HTTP-request time series data to illustrate the techniques.

In what follows, it will be convenient to define a class of "censored Gaussian" (abbreviated as CG) distributions as follows. Suppose that V is a normally distributed random variable with mean μ and variance σ^2. Then we will say that the random variable $W = g(V)$ (recall the definition (4)) has a censored Gaussian distribution with parameters μ, σ^2 and c, and we

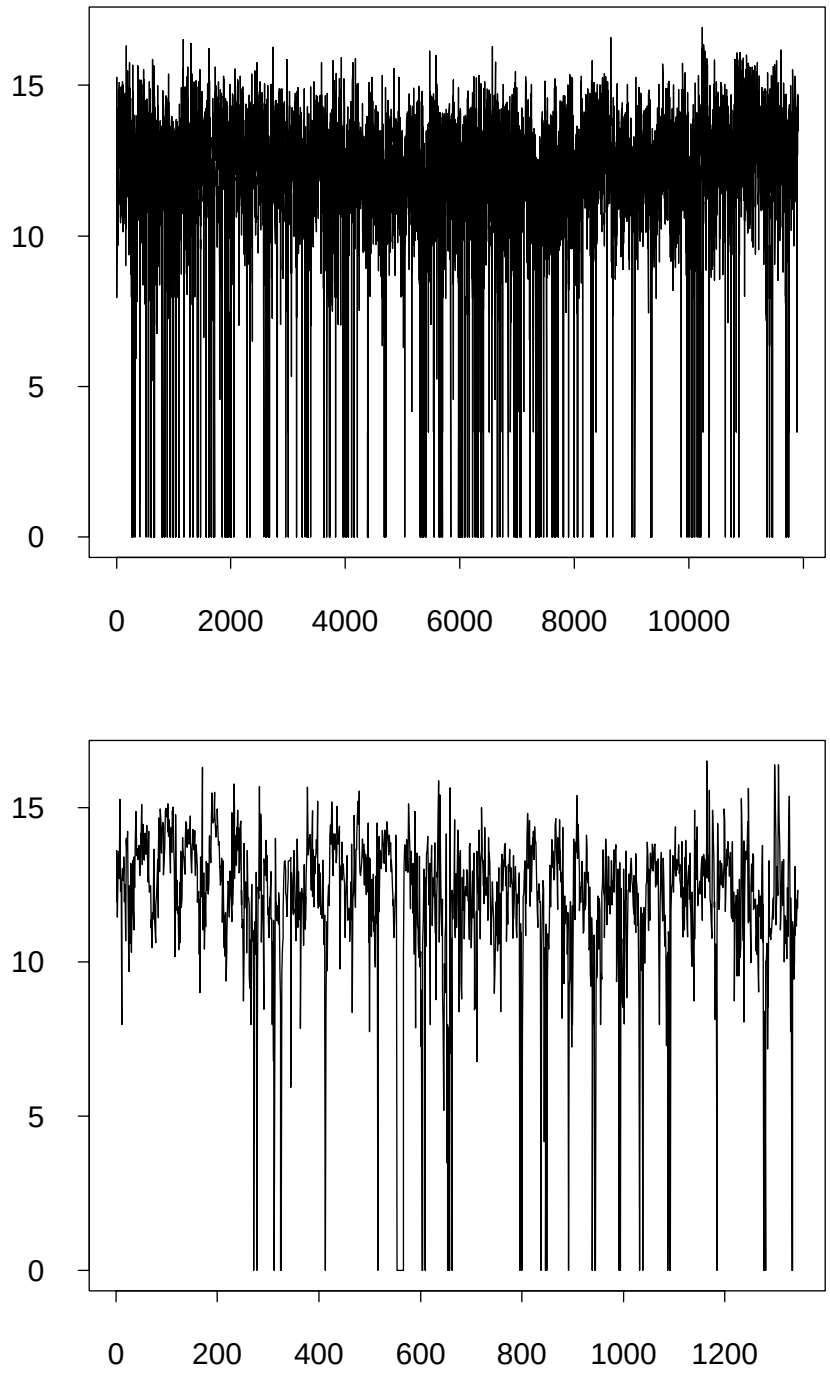

Figure 3: Log of the HTTP-request volume plus one, aggregated in half-hour time units, from Feb. 11th, 1995 to Apr. 2nd, 1995, top: the whole time series, bottom: the first four weeks.

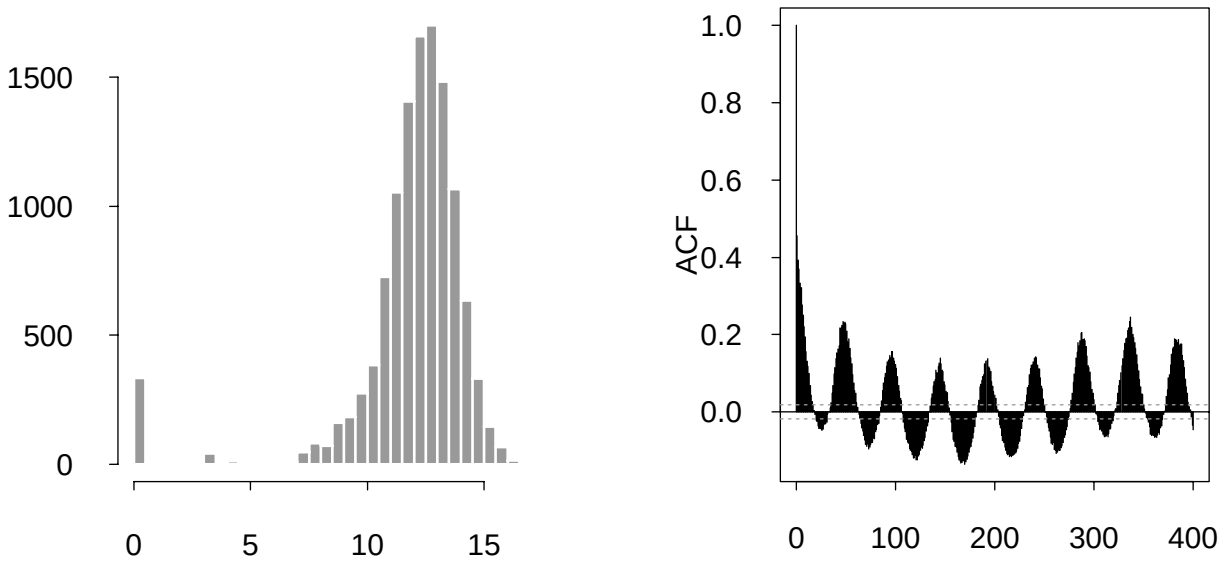

Figure 4: Histogram and sample autocorrelation function of the log HTTP-request time series.

will denote this by $W \sim \mathrm{CG}(\mu, \sigma^2, c)$. The observations Y_t in our state-space model (1, 2, 3) have exactly this kind of distribution, and hence can be regarded as a time series with CG distributions.

We will also use $\phi(\cdot; \mu, \sigma^2)$ and $\Phi(\cdot; \mu, \sigma^2)$ to denote, respectively, the density function and cumulative density function of a Gaussian random variable with mean μ and variance σ^2, with the convention that if μ and σ^2 are omitted, they are implicitly assumed to be 0 and 1 respectively.

We will denote the observations from a time series as $\{y_i,\ i = 1, \ldots, n\}$, with the understanding that depending on context, we could be referring to either of the two time series considered or a generic unspecified time series.

4 Model-Fitting and Forecasting

Our model-fitting procedure consists of two main steps. The first step is to apply a Box-Cox transform to the data to ensure that the empirical distribution of the transformed series is approximately a CG distribution. In carrying out this step, we obtain estimates of the parameters μ, σ^2 and c in the observation equation (3).

The second step is to determine the structure of a reasonable model for the data and estimate its parameters. We determine the structure of the process $\{V_t\}$ by examining its autocovariance function. Once we have determined the structure, we can either use a crude Yule-Walker-like

method for estimation of parameters by matching sample and model autocovariances, or we can use a more sophisticated likelihood-based method such as the Gibbs sampler. When the Yule-Walker-like method is used, forecasts can be generated using particle filtering or some other nonlinear version of the Kalman filter. In the case of Gibbs sampling, forecasts can be obtained directly as part of the parameter estimation procedure.

4.1 Transformation of the Data

Under the model (1,2,3), the observations $\{Y_t\}$ should have a CG distribution. The first step is to find a transformation of the original data which ensures that it does indeed have (approximately) a CG distribution. To make the problem simpler, we restrict attention to the class of possible Box-Cox transformations defined by

$$b(x) = \begin{cases} (x^\lambda - 1)/\lambda, & \lambda > 0 \\ \log(x), & \lambda = 0, \end{cases}$$

for values of λ in the interval $[0, 2]$.

It is not difficult to verify that a linear transformation of a CG random variable is also a CG random variable, and, in fact, $Y \sim \mathrm{CG}(\mu, \sigma^2, c)$ can be written as

$$Y = aY^* + b,$$

where $Y^* \sim \mathrm{CG}(0, 1, c^*)$ with

$$c^* = a^{-1}(c - b). \tag{5}$$

Thus the α-quantile of Y is simply a times the α-quantile of Y^*, plus b, and a plot of the quantiles of Y^* versus the quantiles of Y would form a straight line.

In light of this observation, the following measure of the deviation of the empirical distribution of a time series $\{y_1, \ldots, y_n\}$ from a CG distribution is proposed. Let

$$m_y = \min_{i=1\ldots n} y_i,$$

and let $\{y_i',\ i = 1, \ldots, n'\}$ be the sub-series of $\{y_j\}$ containing exactly those elements which are *not* equal to m_y. (Hence $n' \leq n$.) Let $p_1 = (n - n')/n$ and choose $c^* = \Phi^{-1}(p_1)$. Then plot the sample α-quantiles of the series $\{y_j\}$ versus the α-quantiles of a $\mathrm{CG}(0, 1, c^*)$ distribution, but only for $\alpha \in (p_1, 1]$. This is equivalent to plotting order-statistics of $\{y_j', j = 1, \ldots, n'\}$ versus the quantiles $\{q_j\}$ of the $\mathrm{CG}(0, 1, c^*)$ distribution given by

$$q_j = \Phi^{-1}((1 - p_1)(j - 0.5)/n' + p_1), \quad j = 1, \ldots, n'.$$

If the data really did come from a CG distribution, then the resulting plot should be approximately a straight line.

By normalizing both the order statistics and the computed values $\{q_j\}$ so that the minimum and maximum values on both axes are 0 and 1, respectively, our plot should connect the points $(0, 0)$ and $(1, 1)$ in approximately a straight line. As a measure of deviation of the empirical

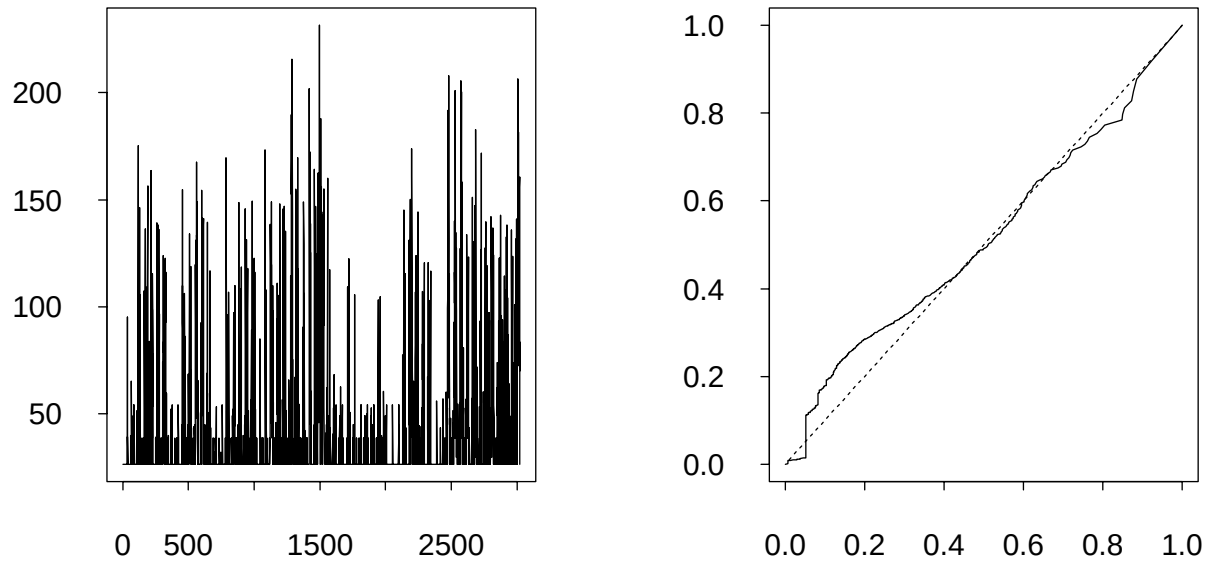

Figure 5: Optimally transformed ($\lambda = 0.24$) disk-trace data, along with the corresponding QQ plot.

distribution from a CG distribution, the total area contained in the unit box $[0,1] \times [0,1]$ which lies between the resulting normalized quantile-quantile plot and the line $\{y = x\}$ is used. We choose the transformation which minimizes this area.

Having defined the measure of deviation, it is a straightforward procedure to use a numerical minimization routine to minimize the measure of deviation of transformed data with respect to the parameter of the Box-Cox transformation.

Using the `nlminb` routine in S-Plus to carry out the numerical minimization, we obtain optimal Box-Cox transform parameters $\lambda \simeq 0.38$ for the HTTP-request data and $\lambda \simeq 0.24$ for the disk-trace data. Plots of the transformed data, along with the corresponding quantile-quantile plots are given in Figures 5 and 6.

4.2 Estimating the Observation Equation Parameters

Once a transformation has been found, we proceed to estimate the parameters μ, σ^2 and c in the observation equation (3). One way of doing this is as follows.

First estimate c using the obvious choice, that is, the minimum observed value,

$$\hat{c} = \min_{i=1,\ldots,n} y_i. \tag{6}$$

Then let $\bar{y} = \frac{1}{n}\sum_{j=1}^{n} y_i$ and $\tilde{y} = \frac{1}{n}\sum_{j=1}^{n} y_i^2$ be sample estimates of the first two moments of the distribution of Y_t. Given $\hat{c}$, we can write down expressions for the stationary mean and

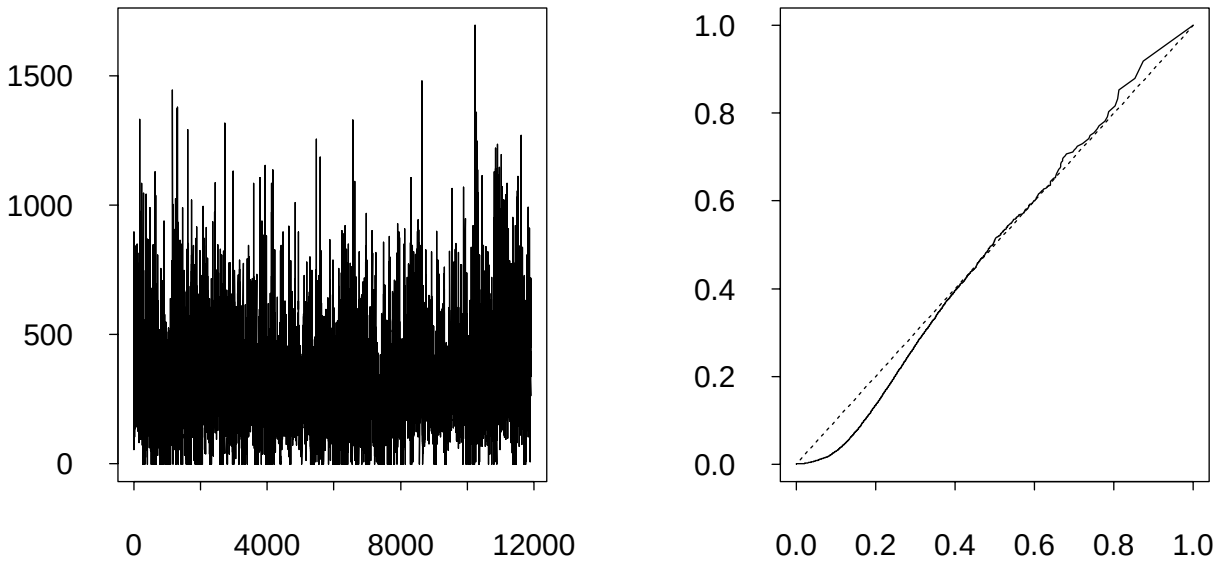

Figure 6: Optimally transformed ($\lambda = 0.38$) HTTP-request data, along with the corresponding QQ plot.

Time Series	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{c}$
Disk-trace	-2.2349	4982.3	26.578
HTTP-request	314.73	37589	-2.614

Table 1: Parameter estimates for the two transformed time series.

second moment of Y_t, and set these equal to sample estimates, obtaining the pair of equations

$$\int_{-\infty}^{\infty} g(x)\phi(x;\mu,\sigma^2)dx = \bar{y} \quad (7)$$

$$\int_{-\infty}^{\infty} g(x)^2\phi(x;\mu,\sigma^2)dx = \tilde{y}. \quad (8)$$

We take the estimates $\hat{\mu}$ and $\hat{\sigma}^2$ to be the values of μ and σ^2 which satisfy (7,8). (The solutions to these equations can be obtained by numerical minimization of the sums of absolute values of differences between the left and right-hand sides of the equations.)

Using this technique to construct observation equation parameter estimates for our two Box-Cox-transformed time series, we obtain the estimates shown in Table 1.

4.3 Constructing the State Equation

Our next object is to build a model for the (unobserved) state process $\{X_t\}$ As a pre-cursor to this, we will examine the autocorrelation function of $\{V_t\}$.

4.3.1 The Autocorrelation of $\{V_t\}$

We first derive a relationship between the autocovariance functions $\gamma_V(\cdot)$ and $\gamma_Y(\cdot)$ of $\{V_t\}$ and $\{Y_t\}$. Let

$$\mu_Y = \mathbf{E}[Y_t] = \int_{-\infty}^{\infty} g(v)\phi(v;\mu,\sigma^2)dv \tag{9}$$

and let $f(x,y;\rho)$ denote the density of a bivariate normal random variable with mean $(\mu,\mu)^T$ and covariance matrix

$$\begin{bmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{bmatrix}.$$

Then since $(V_t, V_{t+h})^T$ is bivariate normal with mean $(\mu,\mu)^T$, and $\mathrm{Cov}(V_t, V_{t+h}) = \rho_V(h)$ (this follows directly from the fact that $\{X_t\}$, and hence $\{V_t\}$, are Gaussian processes), we can write

$$\gamma_Y(h) = \mathbf{E}[Y_t Y_{t+h}] - \mu_Y^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x)g(y)f(x,y;\rho_V(h))dxdy - \mu_Y^2. \tag{10}$$

Thus $\gamma_Y(h)$ is a function of only $\rho_V(h)$ (μ and σ^2 are regarded as constants).

Using standard numerical integration techniques, the two-dimensional integral in (10) can be approximated. Thus, it is possible to construct a mapping d from $\rho_V(h)$ to $\rho_Y(h) = \gamma_Y(h)/\gamma_Y(0)$ (which applies regardless of the choice of h).

To be more precise, we first numerically integrate the expression on the right hand side of (10), fixing $\rho_V(h)$ at values $-.97, -.94, \ldots, 0.92, 0.95$. This gives us estimates of $\gamma_Y(h)$ when $\rho_V(h) = -.97, \ldots, 0.95$. We also compute $\gamma_Y(0) = \mathrm{Var}(Y_t)$. Thus we have estimates of $\rho_Y(h) = \gamma_Y(h)/\gamma_Y(0)$ for $\rho_V(h) = -0.97, \ldots, 0.95$. We then fit fourth-degree polynomials (for instance, by least-squares, using the `lm` command in Splus), to find $\rho_Y(h)$ as a function of $\rho_V(h)$. In other words, we find a mapping d such that

$$\rho_Y(h) \simeq d(\rho_V(h)) = \alpha_0 + \alpha_1\rho_V(h) + \alpha_2\rho_V(h)^2 + \alpha_3\rho_V(h)^3 + \alpha_4\rho_V(h)^4.$$

The mappings constructed for the two transformed time series (based on the estimated parameters given in Table 1) are shown in Figure 7. The fitted polynomials are

$$d_1(x) = 0.0002 + 0.5885x + 0.3408x^2 + 0.0359x^3 + 0.0252x^4, \tag{11}$$

for the disk-trace data and

$$d_2(x) = 0.9857x + 0.0056x^2 + 0.0057x^3 + 0.0024x^4, \tag{12}$$

for the HTTP-request data. Note that $d_2(x)$ is not very different from the identity function. This makes sense since the cut-off point c for the HTTP-request data is at a relatively low quantile of the underlying normally distributed random variables V_t, and hence the effect of applying the function $g(\cdot)$ to V_t is relatively small.

Now that we have constructed approximate mappings from the autocorrelation function of $\{V_t\}$ to that of $\{Y_t\}$, we can proceed to apply the inverse mappings to the sample autocorrelation

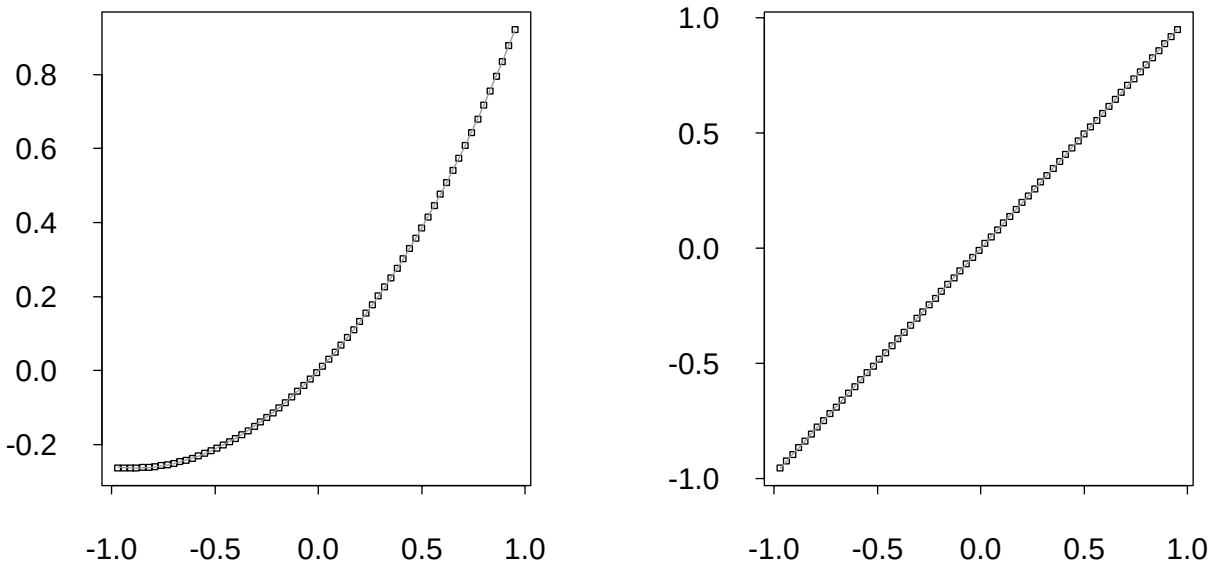

Figure 7: The mappings from the autocorrelation function $\rho_V(h)$ to the autocorrelation function $\rho_Y(h)$: left: $d_1(\cdot)$, the mapping for the disk-trace data, right: $d_2(\cdot)$, the mapping for the HTTP-request data. Solid lines represent fitted values and boxes mark points computed by numerical integration.

functions of our transformed time series. This yields estimates $\hat{\rho}_V(\cdot)$ of the sample autocorrelation functions of the underlying processes $\{V_t\}$ (note that these are the same as the sample autocorrelations of $\{h^T X_t\}$).

Figures (8) and (9) show the estimated sample autocorrelation functions $\hat{\rho}_V(h) = d_1^{-1}(\hat{\rho}_Y(h))$ for the disk-trace data and $\hat{\rho}_V(h) = d_2^{-1}(\hat{\rho}_Y(h))$ for the HTTP-request data.

4.3.2 Constructing a Model for $\{X_t\}$

Having estimated the autocorrelation function of $\{V_t\}$ (which is the same as the autocorrelation function of $\{X_t\}$), the next task is to construct a model for the underlying state process $\{X_t\}$ for each of our time series. At this point, the estimated autocorrelation function can be used in the usual fashion to obtain an initial guess at the structure of the model.

The two estimated autocorrelation functions obtained (shown in Figures 8 and 9) exhibit several key features. They both have a slowly damped sinusoidal component with period 48 (corresponding to one day), they also have another less obvious sinusoidal component with period 336 (corresponding to one week). Both functions have additional rapidly decaying positive correlation at small lags.

An AR(2) process with an autoregressive polynomial having complex conjugate roots has a

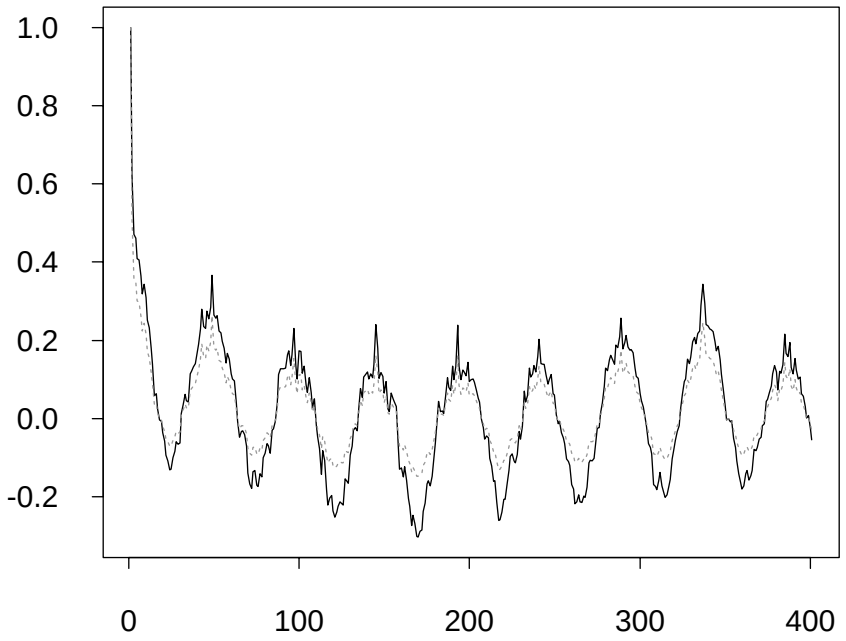

Figure 8: The estimated sample autocorrelation $\hat{\rho}_V(\cdot)$ for the disk-trace data, plotted for lags 0 to 400. The sample autocorrelation of $\{Y_t\}$ is shown with a dotted line.

damped sinusoidal autocovariance function, and an AR(1) with a positive autoregressive coefficient has an exponentially decaying autocovariance function. Since the autocovariance function of a sum of independent stationary time series is the sum of the individual autocovariance functions, it makes sense to try to model our process as the sum of two independent AR(2) processes, and an AR(1) process (also independent of the two AR(2) processes).

To be precise, consider the case where $\{V_t\}$ is the sum of three independent time series, $\{Q_t\}, \{R_t\}$ and $\{S_t\}$, which are the stationary solutions of the equations

$$Q_t - \phi_1^q Q_{t-1} - \phi_2^q Q_{t-2} = Z_t^q \tag{13}$$

$$R_t - \phi_1^r R_{t-1} - \phi_2^r R_{t-2} = Z_t^r \tag{14}$$

$$S_t - \phi^s S_{t-1} = Z_t^s \tag{15}$$

where $\{Z_t^q\}, \{Z_t^r\}$ and $\{Z_t^s\}$ are three independent sequences of independent and identically distributed normal random variables, with mean zero and variances σ_1^2, σ_2^2 and σ_3^2, respectively.

Thus the argument to the function $g(\cdot)$ in the observation equation (3) will be the sum of the AR(2) processes $\{Q_t\}$ and $\{R_t\}$ as well as the AR(1) process $\{S_t\}$. The parameters $\phi_1^q, \phi_2^q, \phi_1^r$ and ϕ_2^r will be further constrained (as described below) so that the autocorrelations of $\{Q_t\}$ and $\{R_t\}$ have peaks at multiples of 48 and 336, respectively. The object is for $\{Q_t\}$ to capture the sinusoidal component of the autocorrelation function which has period 48, for $\{R_t\}$ to capture the lower frequency sinusoidal component with period 336, and for the process $\{S_t\}$ to account for remaining autocorrelation at low lags.

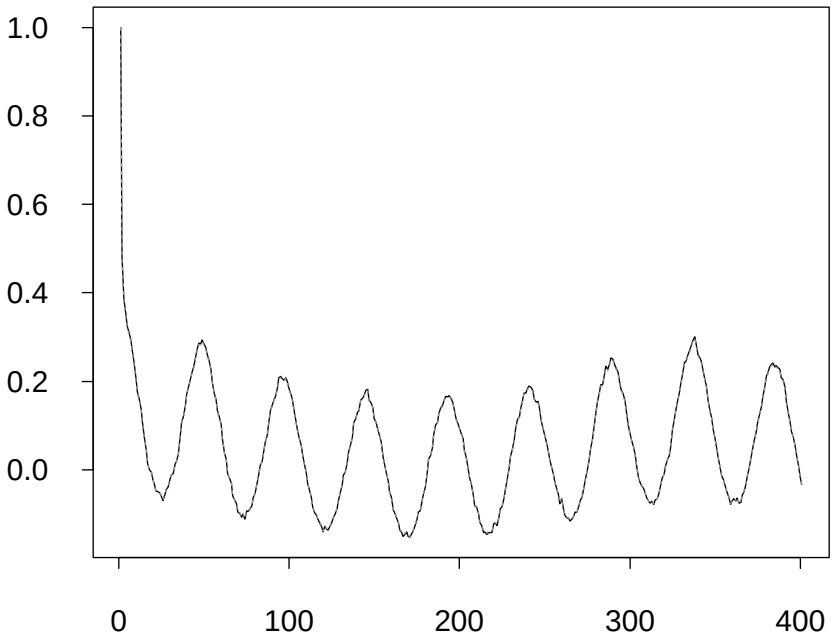

Figure 9: The estimated sample autocorrelation $\hat{\rho}_V(\cdot)$ for the HTTP-request data, plotted for lags 0 to 400. The sample autocorrelation of $\{Y_t\}$ is shown with a dotted line. For this time series the two functions are virtually identical.

Notice that we are effectively using AR(2) models with constrained parameters to capture daily and weekly patterns in the data. Another possibility would be to incorporate seasonal components into our model, for instance, by replacing the observation equation (3) with $Y_t = g(V_t + s_t)$, where s_t is a deterministic sequence or a harmonic function with period 48 or 336. (This would also require adjustments to be made to the procedure for estimation of μ, σ^2 described in Section 4.2.)

As long as the polynomials $\phi_q(z) = 1 - \phi_1^q z - \phi_2^q z^2$ and $\phi_r(z) = 1 - \phi_1^r z - \Phi_2^r z^2$ have complex conjugate roots, i.e.

$$\phi_q(z) = (1 - r_q^{-1} e^{-i\theta_q} z)(1 - r_q^{-1} e^{i\theta_q} z), \quad \text{and} \tag{16}$$

$$\phi_r(z) = (1 - r_r^{-1} e^{-i\theta_r} z)(1 - r_r^{-1} e^{i\theta_r} z), \tag{17}$$

then the autocovariances $\gamma_Q(\cdot)$ and $\gamma_R(\cdot)$ of $\{Q_t\}$ and $\{R_t\}$ can be written as

$$\gamma_Q(h) = \beta_q r_q^{-h} \sin(\theta_q h + \psi_q)$$
$$\gamma_R(h) = \beta_r r_r^{-h} \sin(\theta_r h + \psi_r),$$

where

$$\beta_q = \frac{\sigma_1^2 r_q^4}{(r_q^2 - 1)(r_q^4 - 2r_q^2 \cos(2\theta_q) + 1)^{1/2} \sin(\theta_q)},$$

with

$$\psi_q = \arctan\left(\frac{r_q^2+1}{r_q^2-1}\right)\tan(\theta_q),$$

and analogous formulae apply for β_r and ψ_r.

To enforce our frequency requirements in the autocorrelation function, we restrict the polynomials $\phi_q(\cdot)$ and $\phi_r(\cdot)$ by requiring $\theta_q = 2\pi/48$ and $\theta_r = 2\pi/336$.

Having defined the components, we can now write down our state space model as (1,2,3) with $X_t = (Q_t, Q_{t-1}, R_t, R_{t-1}, S_t)^T$,

$$F = \begin{bmatrix} \phi_1^q & \phi_2^q & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \phi_1^r & \phi_2^r & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \phi^s \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_2^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_3^2 \end{bmatrix}, \tag{18}$$

and $h = (1,0,1,0,1)^T$. The parameters μ, σ and c are replaced by their estimates as obtained in Subsection 4.2. Note that since $h^T X_t$ is normalized so that the variance of V_t is σ^2, the variances σ_j^2 can all be scaled by a constant without altering the model. In other words, only the relative magnitudes are important. Therefore without loss of generality, fix $\sigma_3 = 1 - (\phi^s)^2$, so that the variance of S_t is equal to one, and regard only $r_q, r_r, \phi^s, \sigma_1$ and σ_2 as parameters.

4.4 Crude Parameter Estimation

One simple approach for estimating the unknown parameters $r_q, r_r, \phi^s, \sigma_1$ and σ_2 is to match the autocorrelation of $\{V_t\}$ with the estimated sample autocorrelation $\hat{\rho}_V(\cdot)$ as computed in Subsection 4.3.1. This is a nonlinear problem which in general cannot be solved analytically. However, with the wide availability of software which performs numerical minimization, it is a simple matter to choose parameters to minimize some measure of the difference between the model and sample autocorrelation.

For our problem, it is easy to compute the autocovariance (and hence the autocorrelation) of $\{h^T X_t\}$,

$$\gamma_{h^T X}(k) = \gamma_Q(k) + \gamma_R(k) + \gamma_S(k),$$

given the parameters. Since $\{V_t\}$ is simply a scaled and shifted version of $\{h^T X_t\}$, we have $\rho_V(k) = \rho_{h^T X}(k) = \gamma_{h^T X}(k)/\gamma_{h^T X}(0)$. Next, let us define the error measure

$$e(r_q, r_r, \phi^s, \sigma_1, \sigma_2) = \sum_{j=1}^{400} d^j |\rho_V(j) - \hat{\rho}_V(j)|, \tag{19}$$

where $d \in (0,1]$ is some damping factor and $\hat{\rho}_V(\cdot)$ is the inverse-mapped sample autocorrelation described in Subsection 4.3.1. The damping factor d is used to assign higher importance to lower lags of the autocorrelation. This is important since for purposes of short-range forecasting, it is clearly better to have a good match at small lags and a poor match at large lags than

Time Series	$\hat{\phi}_1^q$	$\hat{\phi}_2^q$	$\hat{\phi}_1^r$	$\hat{\phi}_2^r$	$\hat{\phi}^s$
Disk-trace	1.98072	-0.99781	1.98868	-0.98905	0.55634
HTTP-request	1.97974	-.99683	1.96038	-0.96111	0.20587

Time Series	σ_1	$\mathrm{Var}(Q_t)$	σ_2	$\mathrm{Var}(R_t)$	σ_3	$\mathrm{Var}(S_t)$
Disk-trace	0.005154	0.35621	0.001377	0.22928	0.6905	1.0000
HTTP-request	0.005758	0.30771	0.003646	0.23460	0.9576	1.0000

Table 2: Parameter estimates for the disk-trace and HTTP-request time series.

to have a good match at large lags and a poor match at small lags. Using numerical minimization techniques, it is then possible to minimize the error as a function of the five unknown parameters.

Using this technique, with a damping factor $d = 0.99$, we obtain the parameter estimates given in Table 2. The corresponding model and (transformed) sample autocorrelation functions are shown in Figures 10 and 11.

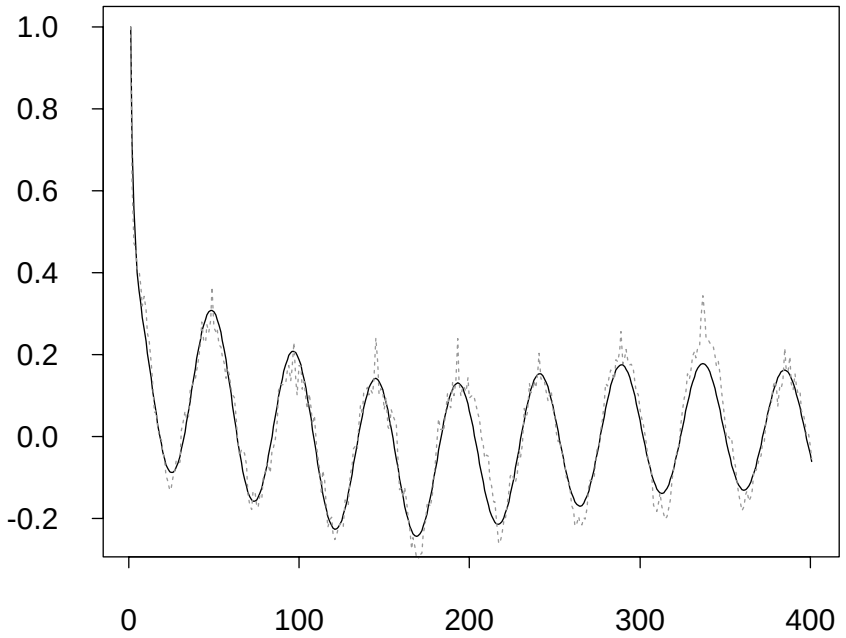

Figure 10: Model (fitted) and (transformed) sample autocorrelations for the disk-trace data, plotted for lags 0 up to 400. The sample autocorrelations are shown with a dotted line.

Once parameters have been estimated, forecasting can be carried out using a simulation-based approach. Since the observation equation (3) is nonlinear, the Kalman filter cannot be used for filtering and forecasting. However, recently developed sequential Monte Carlo methods (see, e.g., Doucet et al., 2001, for a good overview of such methods) can be used to compute good

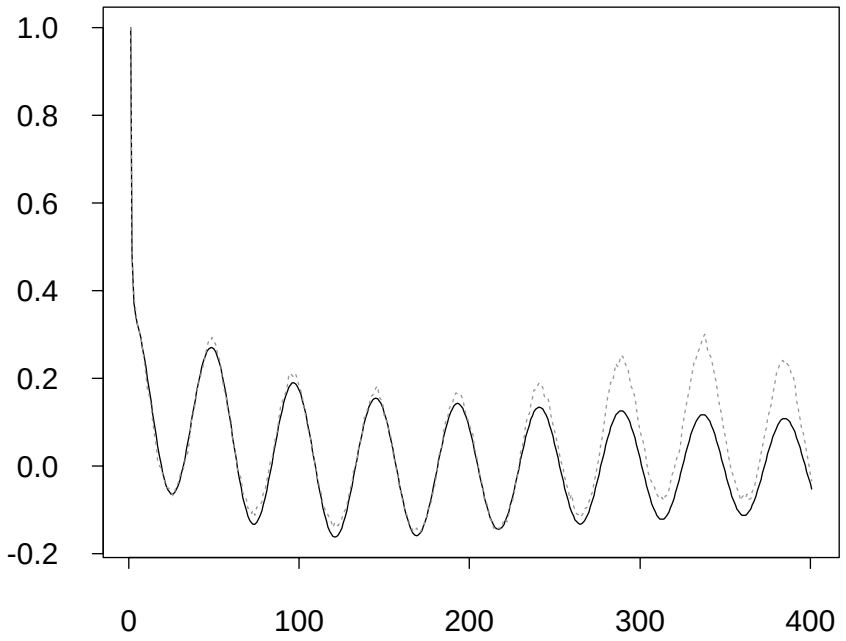

Figure 11: Model (fitted) and (transformed) sample autocorrelations for the HTTP-request data, plotted for lags 0 up to 400. The sample autocorrelations are shown with a dotted line.

approximations to filtering and predictive distributions. In this paper we will not go further into the details of these algorithms. Rather, we will revisit the problem of parameter estimation and forecasting, using a more sophisticated likelihood-based procedure.

4.5 Parameter Estimation and Forecasting Using the Gibbs Sampler

In the previous section, we briefly discussed model-fitting by minimizing a measure of the difference between sample and true autocovariance functions. Even if it were possible to match these functions exactly for some set of lags, the resulting estimators would typically be inefficient relative to maximum likelihood estimates (see, e.g. Brockwell and Davis, 1991, Chapter 8). Therefore, in this section, we consider a more computationally intensive procedure based on the likelihood. Essentially, we adopt the approach of Carlin et al. (1992) to perform a Bayesian analysis of the data. There are, however, several complications which prevent their method from being applied directly to our problem. Hence we present a slightly modified version of their algorithm here.

4.5.1 The Sampler

Let $Y = \{Y_1, \ldots, Y_n\}$ denote the entire observation process and $X^* = \{\kappa X_1, \kappa X_2, \ldots, \kappa X_n\}$, $\kappa = \sigma(h^T \Lambda h)^{-1/2}$, denote the "scaled state process". Let θ vector of parameters we wish to estimate.

The Gibbs sampler was proposed by Carlin et al. (1992) for use in a wide-class of state-space modelling problems. It provides a standard approach for constructing a Markov chain $\{M_k = (\mathcal{X}^{(k)}, \theta^{(k)}),\ k = 1, 2, \ldots\}$ whose limiting distribution is the posterior distribution $p(X^*, \theta|Y) \propto p(Y, X^*, \theta)$. Here $\mathcal{X}^{(k)}$ is a set $\{X_t^{*(k)},\ t = 1, \ldots, n\}$ representing a kth approximate sample from the marginal posterior distribution of the scaled state process and $\theta^{(k)}$ represents a kth approximate sample from the marginal posterior of the parameter vector. After an initial burn-in period, samples $(\mathcal{X}^{(k)}, \theta^{(k)})$ from the chain can be regarded as approximate (non-independent) draws from the desired posterior distribution.

Before formally stating the Gibbs sampling algorithm, we point out that

$$p(Y, X^*, \theta) = I_c(Y, X^*)p(X^*, \theta) = I_c(Y, X^*)p(X^*|\theta)p(\theta),$$

where $I_c(Y, X^*)$ is equal to one if if $Y_i = g(V_i) = g(h^T X_i^* + \mu)$ for $i \in \{1, 2, \ldots, n\}$, and zero otherwise. Furthermore,

$$p(X^*, \theta) = p(X_1^*|\theta) \prod_{i=2}^{n} p(X_i^*|X_{i-1}^*),$$

where $X_1^*|\theta \sim N(0, \kappa^2\Lambda)$ and, for $i > 1$, $X_i^*|X_{i-1}^* \sim N(FX_{i-1}^*, \kappa^2\Sigma)$. Therefore it is a straightforward matter to compute the density $p(Y, X^*, \theta)$.

Our variation on the Gibbs sampler of Carlin et al. (1992) is as follows.

Gibbs Sampling Algorithm

1. Set $k = 1$. Choose some initial state $M_1 = (\mathcal{X}^{(1)}, \theta^{(1)})$ satisfying $I_c(Y, \mathcal{X}^{(1)}) = 1$.
2. Replace k by $k + 1$. Set $\theta^{(k)} = \theta^{(k-1)}$ and $\mathcal{X}^{(k)} = \mathcal{X}^{(k-1)}$.
3. For $i = 1, 2, \ldots, n-1$, replace the pair of vectors $(X_i^{*(k)}, X_{i+1}^{*(k)})$ by a draw from its conditional distribution, given Y, $\{X_j^{*(k)},\ j \neq i,\ j \neq i+1\}$, and $\theta^{(k)}$.
4. Update the components of $\theta^{(k)}$ one at a time. For each component $\theta_i^{(k)}$, carry out the following (Metropolis-Hastings) procedure. Draw a proposal Z from a density $g_i(\theta_i^{(k)}, \cdot)$. Compute
$$\alpha = \frac{p(Y, \mathcal{X}^{(k)}, \theta') g_i(Z, \theta_i^{(k)})}{p(Y, \mathcal{X}^{(k)}, \theta^{(k)}) g_i(\theta_i^{(k)}, Z)},$$
where θ' is $\theta^{(k)}$ with its ith element replaced by Z. With probability $\min(1, \alpha)$, replace $\theta^{(k)}$ by θ'.
5. Go back to Step 2.

(Note that we have abused terminology slightly since our Gibbs sampler includes Metropolis-Hastings update steps for the components of θ. For this to be a true Gibbs sampler, the proposals in Step 4 should be drawn from the appropriate full-conditional distributions.)

There are two important differences between this algorithm and that of Carlin et al. (1992). The first is that in Step 2, we draw from full-conditional distributions for blocks of two consecutive scaled state vectors at a time. This is essential for the particular model structure (see (18)) that we use; if we were to adopt the typical approach of updating only $X_i^{*(k)}$ given $Y, X_j^{*(k)},\ j \neq i\}$, and $\theta^{(k)}$, then the resulting Markov chain would not be irreducible. The fact that we use overlapping blocks does not matter, since each update still preserves the invariant (posterior) distribution of the chain. The second difference is that the censored Gaussian distributions in our model cannot be represented as scale mixtures of Gaussian distributions as required in Carlin et al. (1992).

4.5.2 Sampling for Step 3

In spite of the fact that our observations are not scale mixtures of Gaussian random variables, it is not difficult to implement the Gibbs sampler. The proposal distributions in Steps 4-6 are chosen to be easy to sample from. The only real difficulty in implementation of this algorithm is drawing from the conditional distributions in Step 3.

There are three cases we need to consider in drawing from the conditional distributions in Step 3: the case where $i = 1$, the case where $1 < i < n-2$, and the case where $i = n-1$.

For $i = 1$, we need to sample from (X_1^*, X_2^*) given Y, θ and $\{X_j^*,\ j > 2\}$. By the Markov

property, this is the same as sampling from (X_1^*, X_2^*) given Y_1, Y_2, θ, and X_3^*. Since $\{X_t^*\}$ is a zero-mean Gaussian process, the joint distribution of (X_1^*, X_2^*, X_3^*), given θ, is multivariate normal with mean $(0,0,0)$ and variance

$$\kappa^2 \begin{bmatrix} \Lambda & \Lambda F^T & \Lambda (F^2)^T \\ F\Lambda & \Lambda & \Lambda F^T \\ F^2\Lambda & F\Lambda & \Lambda \end{bmatrix}.$$

The matrices F and Λ depend on θ. By a standard conditioning result for multivariate normal distributions (see, e.g. Anderson, 1984, Theorem 2.5.1), the distribution of (X_1^*, X_2^*) given X_3^*, θ is then

$$(X_1^*, X_2^*)|X_3^*, \theta \sim \mathrm{N}\left(m_1, \kappa^2 V_1\right),$$

where

$$m_1 = \begin{bmatrix} \Lambda (F^2)^T \\ \Lambda F^T \end{bmatrix} \Lambda^{-1} X_3^* \quad \text{and} \quad V_1 = \begin{bmatrix} \Lambda & \Lambda F^T \\ F\Lambda & \Lambda \end{bmatrix} - \begin{bmatrix} \Lambda (F^2)^T \\ \Lambda F^T \end{bmatrix} \Lambda^{-1} \left[F^2\Lambda, F\Lambda\right].$$

Hence to get a draw from the desired distribution, we simply need to draw from a multivariate normal with mean m_1 and variance $\kappa^2 V_1$, conditioned on the event

$$\{g(h^T X_1^* + \mu) = y_1\} \cap \{g(h^T X_2^* + \mu) = y_2\}.$$

It is relatively straightforward to do this - an algorithm is given in the Appendix.

For $i = 2, \ldots, n-2$, we use a similar approach. This time we are interested in the conditional distribution of (X_i^*, X_{i+1}^*) given X_{i-1}^*, X_{i+2}^*, Y and θ. We begin by considering the distribution

$$(X_i^*, X_{i+1}^*, X_{i+2}^*|X_{i-1}^*, \theta) \sim \mathrm{N}\left(\begin{bmatrix} FX_{i-1}^* \\ F^2 X_{i-1}^* \\ F^3 X_{i-1}^* \end{bmatrix}, \kappa^2 \begin{bmatrix} \Sigma & \Sigma F^T & G_{13} \\ F\Sigma & \Sigma' & G_{23} \\ G_{13}^T & G_{23}^T & \Sigma'' \end{bmatrix}\right),$$

where $G_{13} = \Sigma (F^2)^T$, $G_{23} = F(\Sigma F^T) F^T + \Sigma F^T$, $\Sigma' = F\Sigma F^T + \Sigma$, and $\Sigma'' = F\Sigma' F^T + \Sigma$. Again using Theorem 2.5.1 of Anderson (1984), we get

$$(X_i^*, X_{i+1}^*)|X_{i+2}^*, X_{i-1}^*, \theta \sim \mathrm{N}(m_i, \kappa^2 V_2),$$

where

$$m_i = \begin{bmatrix} FX_{i-1}^* \\ F^2 X_{i-1}^* \end{bmatrix} + \begin{bmatrix} G_{13} \\ G_{23} \end{bmatrix} \Sigma''^{-1} (X_{i+2}^* - F^3 X_{i-1}^*),$$

and

$$V_2 = \begin{bmatrix} \Sigma & \Sigma F^T \\ F\Sigma & \Sigma' \end{bmatrix} - \begin{bmatrix} G_{13} \\ G_{23} \end{bmatrix} \Sigma''^{-1} [G_{13}^T, G_{23}^T].$$

To draw from the desired distribution, we can again use the algorithm in the Appendix to draw from a multivariate normal with mean m_i and variance $\kappa^2 V_2$, conditioned on the event

$$\{g(h^T X_i^* + \mu) = y_i\} \cap \{g(h^T X_{i+1}^* + \mu) = y_{i+1}\}.$$

For $i = n-1$, we have

$$(X_i^*, X_{i+1}^*)|X_{i-1}^* \sim \mathrm{N}(m_i, V_3),$$

with

$$m_i = \begin{bmatrix} FX^*_{i-1} \\ F^2X^*_{i-1} \end{bmatrix} \quad \text{and} \quad V_3 = \begin{bmatrix} \Sigma & \Sigma F^T \\ F\Sigma & \Sigma' \end{bmatrix}.$$

As in the previous two cases, the algorithm in the Appendix can be used to draw from this distribution, conditioned on the event

$$\{g(h^T X^*_i + \mu) = y_i\} \cap \{g(h^T X^*_{i+1} + \mu) = y_{i+1}\}.$$

4.5.3 Forecasting

The Gibbs sampler is readily extended to give predictive distributions of $Y_{n+1}, Y_{n+2}, \ldots$, given $Y_1, \ldots, Y_n$. The basic idea is as follows. Let $f > 0$ be some forecast horizon. Since

$$\begin{aligned} p(X_{n+1}, \ldots, X_{n+f}|Y) &= \int p(X_{n+1}, \ldots, X_{n+f}|Y, X^*, \theta)p(X^*, \theta|Y)d(X^*, \theta) \\ &= \int p(X_{n+1}, \ldots, X_{n+f}|X^*_n, \theta)p(X^*_n, \theta|Y)d(X^*_n, \theta), \end{aligned} \tag{20}$$

the method of composition can be used to draw a sample from $p(X_{n+1}, \ldots, X_{n+f}|Y)$, by first drawing a sample from $p(X^*_n, \theta|Y)$, and then, conditioned on this, drawing a sample from $p(X_{n+1}, \ldots, X_{n+f}|X^*_n, \theta)$. Our predictive sample for the observation process is then $\{Y_{n+k} = g(\kappa X_{n+k} + \mu),\ k = 1, 2, \ldots, f\}$.

Thus at the end of Step 3, we simply "simulate into the future" using the current value of $X^{*(k)}_n$ and θ, The resulting simulation can be regarded as a draw from the predictive distribution of $Y_{n+1}, \ldots, Y_{n+f}$, given $Y_1, \ldots, Y_n$. It is important to remember that draws at successive iterations of k are not necessarily independent.

A convenient property of this approach is that forecasts are not based on some fixed (for instance, maximum likelihood) estimate of model parameters. Rather, they take into account uncertainty in parameter estimation, since the integral in (20) is taken over all possible parameter values θ.

4.5.4 Results

In order to implement the Gibbs sampler for our problem, we use the five-dimensional parameter vector $\theta = (\theta_1, \ldots, \theta_5)$ specified by $r_q = 1.01 + |\theta_1|, r_r = 1.01 + |\theta_2|, \sigma_1 = \exp(\theta_3 - 2), \sigma_2 = \exp(\theta_4 - 2)$, and $\phi^s = \theta_5$. This parameterization ensures that θ can move freely in $\mathbb{R}^5$ and the constraints $r_q \geq 1.01, r_r \geq 1.01, \sigma_1 > 0, \sigma_2 > 0$ will always be satisfied. (The constraints $r_q > 1$ and $r_r > 1$ ensure that the processes $\{Q_t\}$ and $\{R_t\}$ are stationary; using 1.01 instead of 1 prevents numerical overflow/underflow problems which can occur otherwise.) We choose a flat (improper) prior

$$p(\theta) = I_{(-1,1)}(\theta_4) = I_{(-1,1)}(\phi^s).$$

(The restriction $\theta_4 \in (-1, 1)$ guarantees that the process $\{S_t\}$ is stationary.) To update the components of θ, we use random walk proposal densities, so that $g_i(\theta^{(k)}_i, \cdot)$ is normally dis-

tributed with mean $\theta_i^{(k)}$ and variance ν_i, and we choose (after some experimentation to get "good" mixing properties) $\nu_1 = \nu_2 = \nu_5 = 0.01$ and $\nu_3 = \nu_4 = 0.006$.

Implementing the Gibbs sampler with these parameters for both the disk-trace and HTTP-request data, we generate chains of length 11000, and discard the first 1000 iterates as "burn-in" (see, e.g. Gilks et al., 1996, for discussion of the burn-in problem). The resulting posterior distributions for our parameters are summarized, for the disk-trace data, and the HTTP-request data, respectively, in the box-plots in Figures 12 and 13.

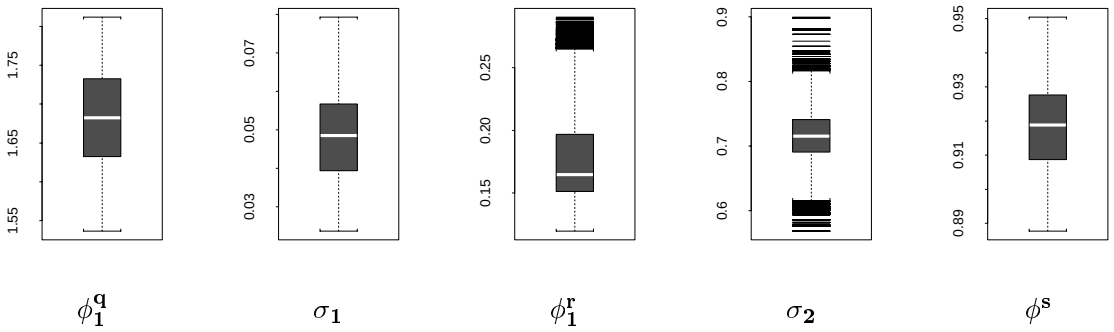

Figure 12: Box-plots of Gibbs-sampled values of the parameters for the disk-trace data. Parameters r_q and r_r are translated back to ϕ_1^q and ϕ_2^q.

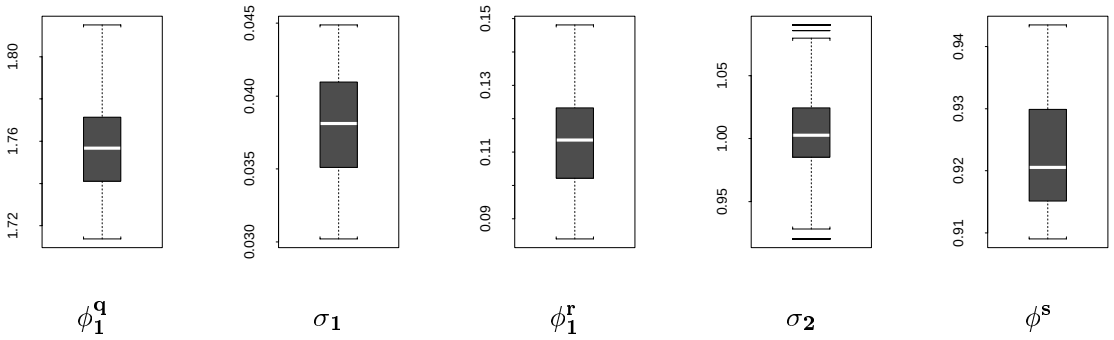

Figure 13: Box-plots of Gibbs-sampled values of the parameters for the HTTP-request data. Parameters r_q and r_r are translated back to ϕ_1^q and ϕ_2^q.

We also obtain approximations to predictive distributions using the method of Section 4.5.3. Sorting our samples of size 10000 for each possible point in the future allows us to compute estimates of the quantiles of the distributions. These sample quantiles are plotted, along with a final portion of the observed data, for the disk-trace and HTTP-request time series, respectively, in Figures 14 and 15. The forecasts, being obtained by simulation from the CG distribution, clearly have the desired saturation property, and quantiles of the predictive distributions can be obtained relatively accurately.

The three component processes Q_t,R_t and S_t can be examined periodically during the running of the Gibbs sampler to assess goodness of fit. For our time series, we would expect the simulated

processes $\{Q_t\}$ and $\{R_t\}$ to look like AR(2) processes, and the simulated process $\{S_t\}$ to look like an AR(1). Examination of the sample autocovariance functions of these processes at several points during the running of the Gibbs sampler demonstrates several common features for both of our time series. Firstly, none of the three processes has negligible variance, although the autocovariance of R_t (which has period 336) is damped so rapidly, that it could potentially be replaced by an AR(1) process in order to reduce the complexity of the model. Secondly, the process S_t does behave like an AR(1) in the sense that low-order autocorrelations are approximately $\rho_S(h) = (\phi^s)^{|h|}$. However, it also exhibits high correlation at lags 48, 96, and so on, suggesting that Q_t is not completely capturing the seasonal component of period 48, and that other methods of dealing with seasonality might be more effective.

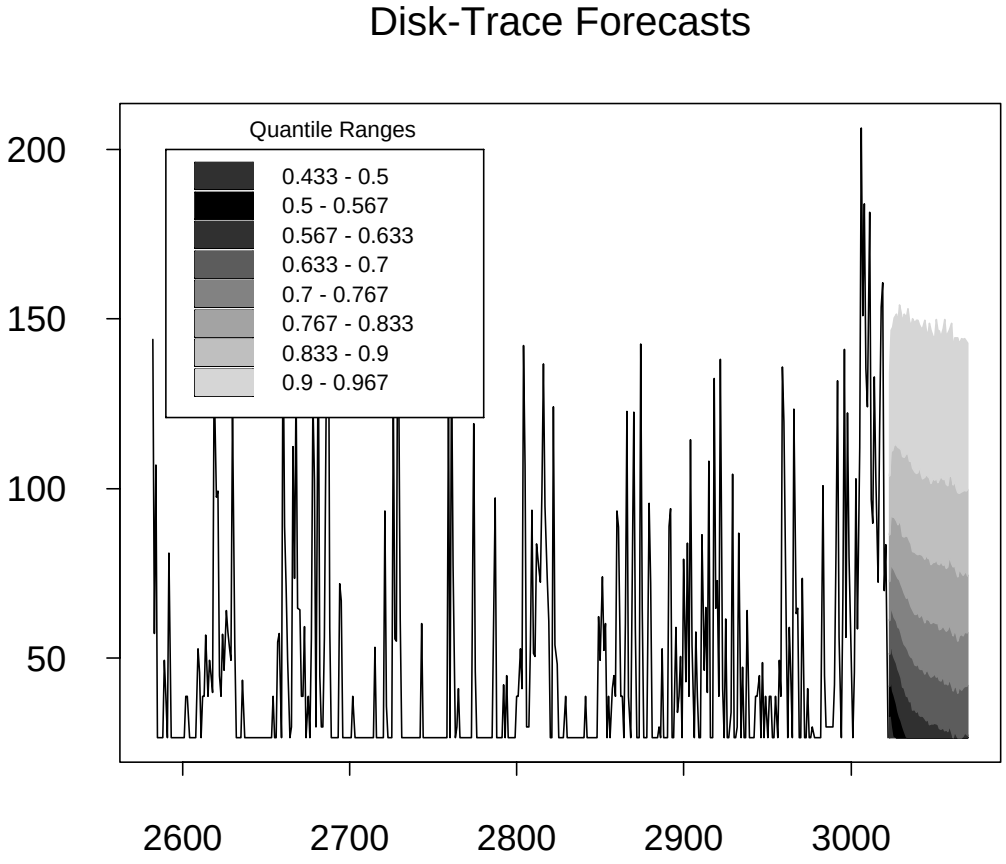

Figure 14: Predictive distributions for the Box-Cox transformed disk-trace data, based on the state-space model of Section 4.3.2. The forecast horizon is 48 half-hour units, or one day.

5 Concluding Remarks

We have demonstrated methods for construction of nonlinear state-space (Tobit) models for the two traffic volume time series we considered. The methods involve preliminary analysis of the estimated autocorrelation function of the uncensored data, for the purpose of determining

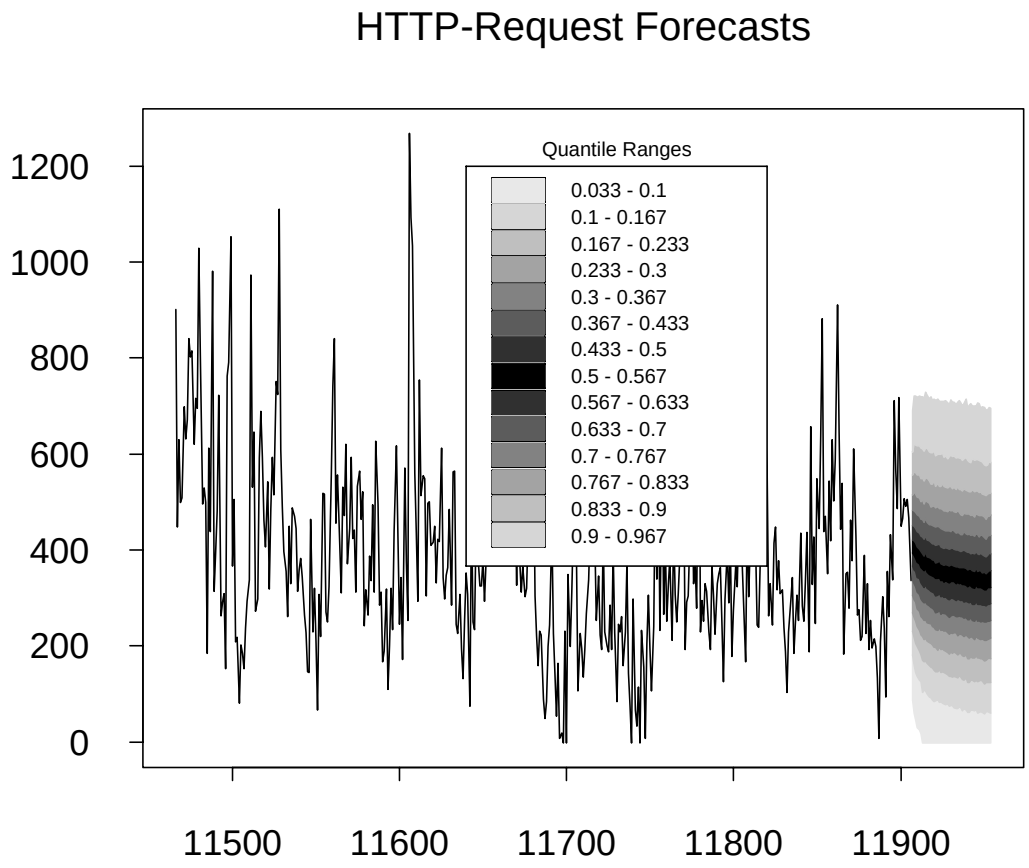

Figure 15: Predictive distributions for the Box-Cox transformed HTTP-request data, based on the state-space model of Section 4.3.2. The forecast horizon is 48 half-hour units, or one day.

the structure of the state equation, followed by an application of a variant of the Gibbs sampler of Carlin et al. (1992). While there are other available methods for working with the models we consider, the Gibbs sampler has at least two advantages: it does not rely on approximations to the likelihood, and it provides forecasts which take into account parameter uncertainty. The Tobit models we fit match both the marginal distribution and the autocorrelation structure of our time series relatively accurately.

In principle, exactly the same methods used in this paper can be used when observations $\{Y_t\}$ are multivariate, with individual components being censored, possibly at different levels. This simply involves (apart from constructing a more complex, probably higher-dimensional state equation) sampling from Gaussian random variables with more conditioning constraints.

As has already been recognized by many authors, the Tobit model has a wide variety of potential applications in modelling censored time series. In light of this, along with the previous remarks, we believe that there is wide scope for potential application of the methods described in this paper.

6 Acknowledgements

The authors are grateful to Christos Faloutsos for his comments and his help in obtaining the data. This work is supported in part by the Research Grants Council of Hong Kong under Grants CUHK6082/98P and 2060205, and the National Science Foundation under Grants DMS-9819950 and IIS-0083148 .

7 Appendix

Here we describe a procedure for sampling from the conditional multivariate normal random variables in Section 4.5.2.

The general problem is to draw from a p-dimensional multivariate normal distribution

$$X \sim \mathrm{N}(\mu, \Sigma),$$

conditioning on the inequality constraints

$$H_1 X > k_1,$$

where H_1 is a $q_1 \times p$ matrix and k_1 is a q_1-dimensional vector, and the equality constraints

$$H_2 X = k_2, \tag{21}$$

where H_2 is a $q_2 \times p$ matrix and k_2 is a q_2-dimensional vector. (The multivariate relation holds if and only if each of its components satisfies the relation.) We assume that $q_1 + q_2 < p$.

The first step is to construct a full-rank $p \times p$ matrix P with block structure

$$P = \begin{bmatrix} P_0 \\ H_1 \\ H_2 \end{bmatrix}.$$

(This can be done using the Gram-Schmidt orthogonalization procedure.) Next we note that $PX \sim \mathrm{N}(P\mu, P\Sigma P^T)$, and we partition

$$P\mu = \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix}, \quad P\Sigma P^T = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix},$$

where Q_{11} is a $(p - q_2) \times (p - q_2)$ matrix and ν_1 is a $(p - q_2)$-dimensional vector.

The original set of equality constraints (21) is the same as the constraint that the bottom q_2-dimensional subvector of PX is equal to k_2, so we set the last q_2 elements of PX equal to (the corresponding elements of) k_2. Given this subvector, the top $p - q_2$-dimensional subvector of PX has a normal distribution with mean μ_0 and variance Σ_0, where

$$\mu_0 = \nu_1 + Q_{12} Q_{22}^{-1} (k_2 - \nu_2) \text{ and } \Sigma_0 = Q_{11} - Q_{12} Q_{22}^{-1} Q_{21}.$$

The next step is to draw the first subvector of PX repeatedly from the $N(\mu_0, \Sigma_0)$ distribution, until its last q_1 components are greater than the corresponding components of k_1. (Various tricks can be used here to reduce the expected number of draws required.)

Finally, the (conditional) draw of PX is premultipled by P^{-1} to obtain a sample from the desired conditional distribution.

References

T.W. Anderson. *An Introduction to Multivariate Statistical Analysis.* Wiley, second edition, 1984.

P.J. Brockwell and R.A. Davis. *Time Series: Theory and Methods.* Springer, second edition, 1991.

B.P. Carlin, N.G. Polson, and D.S. Stoffer. A Monte Carlo approach to nonnormal and nonlinear state-space modeling. *Journal of the American Statistical Association*, 87(418):493–500, 1992. ISSN 0162-1459.

A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice.* Springer, New York, 2001.

W.R. Gilks, S. Richardson, and D.J.Spiegelhalter. *Markov Chain Monte Carlo in Practice.* CRC Press, 1996.

G. Kitagawa and W. Gersch. *Smoothness Priors Analysis of Time Series.* Springer, New York, 1996.

L.F. Lee. Estimation of dynamic and ARCH Tobit models. *Journal of Econometrics*, 92: 355–390, 1999.

V.J. Ribeiro R.H.R. Riedi, M.S. Crouse and R.G. Baraniuk. A multifractal wavelet model with application to network traffic. *IEEE Transactions on Information Theory*, 45:992–1018, 1999.

C. Ruemmler and J. Wilkes. Unix disk access patterns. In *USENIX Winter 1993 Technical Conference Proceedings*, pages 405–420, 1993.

R.H. Shumway and D.S. Stoffer. *Time Series Analysis and its Applications.* Springer, 2000.

M. West and J. Harrison. *Bayesian Forecasting and Dynamic Models.* Springer, New York, second edition, 1997.

S.L. Zeger and R. Brookmeyer. Regression analysis with censored autocorrelated data. *Journal of the American Statistical Association*, 81:722–729, 1986.