1
Generalized Method of Moments

1.1 Introduction

This chapter describes generalized method of moments (GMM) estimation for linear and non-linear models with applications in economics and finance. GMM estimation was formalized by Hansen (1982), and since has become one of the most widely used methods of estimation for models in economics and finance. Unlike maximum likelihood estimation (MLE), GMM does not require complete knowledge of the distribution of the data. Only specified moments derived from an underlying model are needed for GMM estimation. In some cases in which the distribution of the data is known, MLE can be computationally very burdensome whereas GMM can be computationally very easy. The log-normal stochastic volatility model is one example. In models for which there are more moment conditions than model parameters, GMM estimation provides a straightforward way to test the specification of the proposed model. This is an important feature that is unique to GMM estimation.

This chapter is organized as follows. GMM estimation for linear models is described in Section 1.2. Section 1.3 describes methods for estimating the efficient weight matrix. Sections 1.4 and 1.5 give examples of estimation and inference using the `S+Finmetrics` function `GMM`. Section 1.6 describes GMM estimation and inference for nonlinear models. Section 1.7 provides numerous examples of GMM estimation of nonlinear models in finance including Euler equation asset pricing models, discrete-time stochastic volatility models, and continous-time interest rate diffusion models.

The theory and notation for GMM presented herein follows the excellent treatment given in Hayashi (2000). Other good textbook treatments of GMM at an intermediate level are given in Hamilton (1994), Ruud (2000), Davidson and MacKinnon (2004), and Greene (2004). The most comprehensive textbook treatment of GMM is Hall (2005). Excellent surveys of recent developments in GMM are given in the special issues of the *Journal of Business and Economic Statistics* (1996, 2002). Discussions of GMM applied to problems in finance are given in Ogaki (1992), Ferson (1995), Andersen and Sorensen (1996), Campbell, Lo and MacKinlay (1997), James and Webber (2000), Cochrane (2001), Jagannathan and Skoulakis (2002), and Hall (2005).

1.2 Single Equation Linear GMM

Consider the linear regression model

$$y_t = \mathbf{z}_t' \boldsymbol{\delta}_0 + \varepsilon_t,\ t = 1, \ldots, n \tag{1.1}$$

where $\mathbf{z}_t$ is an $L \times 1$ vector of explanatory variables, $\boldsymbol{\delta}_0$ is a vector of unknown coefficients and ε_t is a random error term. The model (1.1) allows for the possibility that some or all of the elements of z_t may be correlated with the error term ε_t, i.e., $E[z_{tk}\varepsilon_t] \neq 0$ for some k. If $E[z_{tk}\varepsilon_i] \neq 0$ then z_{tk} is called an *endogenous variable*. It is well known that if $\mathbf{z}_t$ contains endogenous variables then the least squares estimator of $\boldsymbol{\delta}_0$ in (1.1) is biased and inconsistent.

Associated with the model (1.1), it is assumed that there exists a $K \times 1$ vector of *instrumental variables* $\mathbf{x}_t$ which may contain some or all of the elements of $\mathbf{z}_t$. Let $\mathbf{w}_t$ represent the vector of unique and non-constant elements of $\{y_t, \mathbf{z}_t, \mathbf{x}_t\}$. It is assumed that $\{\mathbf{w}_t\}$ is a stationary and ergodic stochastic process.

The instrumental variables $\mathbf{x}_t$ satisfy the set of K orthogonality conditions

$$E[\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\delta}_0)] = E[\mathbf{x}_t\varepsilon_t] = E[\mathbf{x}_t(y_t - \mathbf{z}_t'\boldsymbol{\delta}_0)] = \mathbf{0} \tag{1.2}$$

where $\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\delta}_0) = \mathbf{x}_t\varepsilon_t = \mathbf{x}_t(y_t - \mathbf{z}_t'\boldsymbol{\delta}_0)$. Expanding (1.2), gives the relation

$$\boldsymbol{\Sigma}_{xy} = \boldsymbol{\Sigma}_{xz}\boldsymbol{\delta}_0$$

where $\boldsymbol{\Sigma}_{xy} = E[\mathbf{x}_t y_t]$ and $\boldsymbol{\Sigma}_{xz} = E[\mathbf{x}_t\mathbf{z}_t']$. For identification of $\boldsymbol{\delta}_0$, it is required that the $K \times L$ matrix $E[\mathbf{x}_t\mathbf{z}_t'] = \boldsymbol{\Sigma}_{xz}$ be of full rank L. This *rank condition* ensures that $\boldsymbol{\delta}_0$ is the unique solution to (1.2). Note, if $K = L$, then $\boldsymbol{\Sigma}_{xz}$ is invertible and $\boldsymbol{\delta}_0$ may be determined using

$$\boldsymbol{\delta}_0 = \boldsymbol{\Sigma}_{xz}^{-1}\boldsymbol{\Sigma}_{xy}$$

A necessary condition for the identification of $\boldsymbol{\delta}_0$ is the *order condition*

$$K \geq L \tag{1.3}$$

which simply states that the number of instrumental variables must be greater than or equal to the number of explanatory variables in (1.1). If $K = L$ then $\boldsymbol{\delta}_0$ is said to be (apparently) just identified; if $K > L$ then $\boldsymbol{\delta}_0$ is said to be (apparently) over-identified; if $K < L$ then $\boldsymbol{\delta}_0$ is not identified. The word "apparently" in parentheses is used to remind the reader that the rank condition

$$rank(\boldsymbol{\Sigma}_{xz}) = L \tag{1.4}$$

must also be satisfied for identification.

In the regression model (1.1), the error terms are allowed to be conditionally heteroskedastic as well as serially correlated. For the case in which ε_t is conditionally heteroskedastic, it is assumed that $\{\mathbf{g}_t\} = \{\mathbf{x}_t\varepsilon_t\}$ is a stationary and ergodic martingale difference sequence (MDS) satisfying

$$E[\mathbf{g}_t\mathbf{g}_t'] = E[\mathbf{x}_t\mathbf{x}_t'\varepsilon_t^2] = \mathbf{S}$$

where $\mathbf{S}$ is a non-singular $K \times K$ matrix. The matrix $\mathbf{S}$ is the asymptotic variance-covariance matrix of the sample moments $\bar{\mathbf{g}} = n^{-1}\sum_{t=1}^{n}\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\delta}_0)$. This follows from the central limit theorem for ergodic stationary martingale difference sequences (see Hayashi page 106)

$$\sqrt{n}\bar{\mathbf{g}} = \frac{1}{\sqrt{n}}\sum_{t=1}^{n}\mathbf{x}_t\varepsilon_t \xrightarrow{d} N(\mathbf{0}, \mathbf{S})$$

where $avar(\bar{\mathbf{g}}) = \mathbf{S}$ denotes the variance-covariance matrix of the limiting distribution of $\sqrt{n}\bar{\mathbf{g}}$.

For the case in which ε_t is serially correlated and possibly conditionally heteroskedastic as well, it is assumed that $\{\mathbf{g}_t\} = \{\mathbf{x}_t\varepsilon_t\}$ is a stationary and ergodic stochastic process that satisfies

$$\begin{aligned}
\sqrt{n}\bar{\mathbf{g}} &= \frac{1}{\sqrt{n}}\sum_{t=1}^{n}\mathbf{x}_t\varepsilon_t \xrightarrow{d} N(\mathbf{0}, \mathbf{S}) \\
\mathbf{S} &= \sum_{j=-\infty}^{\infty}\boldsymbol{\Gamma}_j = \boldsymbol{\Gamma}_0 + \sum_{j=1}^{\infty}(\boldsymbol{\Gamma}_j + \boldsymbol{\Gamma}_j')
\end{aligned}$$

where $\boldsymbol{\Gamma}_j = E[\mathbf{g}_t\mathbf{g}_{t-j}'] = E[\mathbf{x}_t\mathbf{x}_{t-j}'\varepsilon_t\varepsilon_{t-j}]$. In the above, $avar(\bar{\mathbf{g}}) = \mathbf{S}$ is also referred to as the *long-run variance* of $\bar{\mathbf{g}}$.

1.2.1 Definition of the GMM Estimator

The *generalized method of moments* (GMM) estimator of $\boldsymbol{\delta}$ in (1.1) is constructed by exploiting the orthogonality conditions (1.2). The idea is to create a set of estimating equations for $\boldsymbol{\delta}$ by making sample moments match the population moments defined by (1.2). The sample moments based on (1.2) for an arbitrary value $\boldsymbol{\delta}$ are

$$\begin{aligned}\mathbf{g}_n(\boldsymbol{\delta}) &= \frac{1}{n}\sum_{t=1}^{n} g(\mathbf{w}_t,\boldsymbol{\delta}) = \frac{1}{n}\sum_{t=1}^{n}\mathbf{x}_t(y-\mathbf{z}_t'\boldsymbol{\delta}) \\ &= \begin{pmatrix} \frac{1}{n}\sum_{t=1}^{n} x_{1t}(y-\mathbf{z}_t'\boldsymbol{\delta}) \\ \vdots \\ \frac{1}{n}\sum_{t=1}^{n} x_{Kt}(y-\mathbf{z}_t'\boldsymbol{\delta}) \end{pmatrix}\end{aligned}$$

These moment conditions are a set of K linear equations in L unknowns. Equating these sample moments to the population moment $E[\mathbf{x}_t\varepsilon_t] = \mathbf{0}$ gives the estimating equations

$$\mathbf{S}_{xy} - \mathbf{S}_{xz}\boldsymbol{\delta} = \mathbf{0} \tag{1.5}$$

where $\mathbf{S}_{xy} = n^{-1}\sum_{t=1}^{n}\mathbf{x}_t y_t$ and $\mathbf{S}_{xz} = n^{-1}\sum_{t=1}^{n}\mathbf{x}_t\mathbf{z}_t'$ are the sample moments.

If $K = L$ ($\boldsymbol{\delta}_0$ is just identified) and $\mathbf{S}_{xz}$ is invertible then the GMM estimator of $\boldsymbol{\delta}$ is

$$\hat{\boldsymbol{\delta}} = \mathbf{S}_{xz}^{-1}\mathbf{S}_{xy}$$

which is also known as the *indirect least squares* estimator. If $K > L$ then there may not be a solution to the estimating equations (1.5). In this case, the idea is to try to find $\boldsymbol{\delta}$ that makes $\mathbf{S}_{xy} - \mathbf{S}_{xz}\boldsymbol{\delta}$ as close to zero as possible. To do this, let $\hat{\mathbf{W}}$ denote a $K \times K$ symmetric and positive definite (p.d.) weight matrix, possibly dependent on the data, such that $\hat{\mathbf{W}} \xrightarrow{p} \mathbf{W}$ as $n \to \infty$ with $\mathbf{W}$ symmetric and p.d. Then the GMM estimator of $\boldsymbol{\delta}$, denoted $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$, is defined as

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) = \arg\min_{\boldsymbol{\delta}} \; J(\boldsymbol{\delta},\hat{\mathbf{W}})$$

where

$$\begin{aligned} J(\boldsymbol{\delta},\hat{\mathbf{W}}) &= n\mathbf{g}_n(\boldsymbol{\delta})'\hat{\mathbf{W}}\mathbf{g}_n(\boldsymbol{\delta}) \qquad (1.6)\\ &= n(\mathbf{S}_{xy} - \mathbf{S}_{xz}\boldsymbol{\delta})'\hat{\mathbf{W}}(\mathbf{S}_{xy} - \mathbf{S}_{xz}\boldsymbol{\delta})\end{aligned}$$

Since $J(\boldsymbol{\delta},\hat{\mathbf{W}})$ is a simple quadratic form in $\boldsymbol{\delta}$, straightforward calculus may be used to determine the analytic solution for $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$:

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) = (\mathbf{S}_{xz}'\hat{\mathbf{W}}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\hat{\mathbf{W}}\mathbf{S}_{xy} \tag{1.7}$$

Asymptotic Properties

Under standard regularity conditions (see Hayashi Chapter 3), it can be shown that

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) \xrightarrow{p} \boldsymbol{\delta}_0$$
$$\sqrt{n}\left(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) - \boldsymbol{\delta}_0\right) \xrightarrow{d} N(\mathbf{0}, avar(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})))$$

where

$$avar(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})) = (\boldsymbol{\Sigma}_{xz}'\mathbf{W}\boldsymbol{\Sigma}_{xz})^{-1}\boldsymbol{\Sigma}_{xz}'\mathbf{W}\mathbf{S}\mathbf{W}\boldsymbol{\Sigma}_{xz}(\boldsymbol{\Sigma}_{xz}'\mathbf{W}\boldsymbol{\Sigma}_{xz})^{-1} \tag{1.8}$$

A consistent estimate of $avar(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))$, denoted $\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))$, may be computed using

$$\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})) = (\mathbf{S}_{xz}'\hat{\mathbf{W}}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\hat{\mathbf{W}}\hat{\mathbf{S}}\hat{\mathbf{W}}\mathbf{S}_{xz}(\mathbf{S}_{xz}'\hat{\mathbf{W}}\mathbf{S}_{xz})^{-1} \tag{1.9}$$

where $\hat{\mathbf{S}}$ is a consistent estimate for $\mathbf{S} = avar(\bar{\mathbf{g}})$.

The Efficient GMM Estimator

For a given set of instruments $\mathbf{x}_t$, the GMM estimator $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$ is define for an arbitrary positive definite and symmetric weight matrix $\hat{\mathbf{W}}$. The asymptotic variance of $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$ in (1.8) depends on the chosen weight matrix $\hat{\mathbf{W}}$. A natural question to ask is: What weight matrix $\mathbf{W}$ produces the smallest value of $avar(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))$? The GMM estimator constructed with this weight matrix is called the *efficient GMM estimator*. Hansen (1982) showed that efficient GMM estimator results from setting $\hat{\mathbf{W}} = \hat{\mathbf{S}}^{-1}$ such that $\hat{\mathbf{S}} \xrightarrow{p} \mathbf{S}$. For this choice of $\hat{\mathbf{W}}$, the asymptotic variance formula (1.8) reduces to

$$avar(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})) = (\boldsymbol{\Sigma}_{xz}'\mathbf{S}^{-1}\boldsymbol{\Sigma}_{xz})^{-1} \tag{1.10}$$

of which a consistent estimate is

$$\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})) = (\mathbf{S}_{xz}'\hat{\mathbf{S}}^{-1}\mathbf{S}_{xz})^{-1} \tag{1.11}$$

The efficient GMM estimator is defined as

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}) = \arg\min_{\boldsymbol{\delta}} \; n\mathbf{g}_n(\boldsymbol{\delta})'\hat{\mathbf{S}}^{-1}\mathbf{g}_n(\boldsymbol{\delta})$$

which requires a consistent estimate of $\mathbf{S}$. However, consistent estimation of $\mathbf{S}$, in turn, requires a consistent estimate of $\boldsymbol{\delta}$. To see this, consider the case in which ε_t in (1.1) is conditionally heteroskedastic so that $\mathbf{S} = E[\mathbf{g}_t\mathbf{g}_t'] = E[\mathbf{x}_t\mathbf{x}_t'\varepsilon_t^2]$. A consistent estimate of $\mathbf{S}$ has the form[1]

$$\hat{\mathbf{S}} = \frac{1}{n}\sum_{t=1}^{n}\mathbf{x}_t\mathbf{x}_t'\hat{\varepsilon}_t^2 = \frac{1}{n}\sum_{t=1}^{n}\mathbf{x}_t\mathbf{x}_t'(y_t - \mathbf{z}_t'\hat{\boldsymbol{\delta}})^2$$

such that $\hat{\boldsymbol{\delta}} \xrightarrow{p} \boldsymbol{\delta}$. Similar arguments hold for the case in which $\mathbf{g}_t = \mathbf{x}_t\varepsilon_t$ is a serially correlated and heteroskedastic process.

[1] Davidson and MacKinnon (1993, section 16.3) suggest using a simple degrees-of-freedom corrected estimate of S that replaces n^{-1} in (1.17) with $(n-k)$ to improve the finite sample performance of tests based on (1.11).

Two Step Efficient GMM

The two-step efficient GMM estimator utilizes the result that a consistent estimate of $\boldsymbol{\delta}$ may be computed by GMM with an arbitrary positive definite and symmetric weight matrix $\hat{\mathbf{W}}$ such that $\hat{\mathbf{W}} \xrightarrow{p} \mathbf{W}$. Let $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$ denote such an estimate. Common choices for $\hat{\mathbf{W}}$ are $\hat{\mathbf{W}} = \mathbf{I}_k$ and $\hat{\mathbf{W}} = \mathbf{S}_{xx}^{-1} = (n^{-1}\mathbf{X}'\mathbf{X})^{-1}$, where $\mathbf{X}$ is an $n \times k$ matrix with t^{th} row equal to $\mathbf{x}_t'$[2]. Then, a first step consistent estimate of $\mathbf{S}$ is given by

$$\hat{\mathbf{S}}(\hat{\mathbf{W}}) = \frac{1}{n}\sum_{t=1}^{n} \mathbf{x}_t\mathbf{x}_t'(y_t - \mathbf{z}_t'\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))^2 \tag{1.12}$$

The *two-step efficient GMM estimator* is then defined as

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}(\hat{\mathbf{W}})) = \arg\min_{\boldsymbol{\delta}}\ n\mathbf{g}_n(\boldsymbol{\delta})'\hat{\mathbf{S}}^{-1}(\hat{\mathbf{W}})\mathbf{g}_n(\boldsymbol{\delta}) \tag{1.13}$$

Iterated Efficient GMM

The *iterated efficient GMM estimator* uses the two-step efficient GMM estimator $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}(\hat{\mathbf{W}}))$ to update the estimation of $\mathbf{S}$ in (1.12) and then recomputes the estimator in (1.13). The process is repeated (iterated) until the estimates of $\boldsymbol{\delta}$ do not change significantly from one iteration to the next. Typically, only a few iterations are required. The resulting estimator is denoted $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{iter}^{-1})$. The iterated efficient GMM estimator has the same asymptotic distribution as the two-step efficient estimator. However, in finite samples the two estimators may differ. As Hamilton (1994, page 413) points out, the iterated GMM estimator has the practical advantage over the two-step estimator in that the resulting estimates are invariant with respect to the scale of the data and to the initial weighting matrix $\hat{\mathbf{W}}$.

Continuous Updating Efficient GMM

This estimator simultaneously estimates $\mathbf{S}$, as a function of $\boldsymbol{\delta}$, and $\boldsymbol{\delta}$. It is defined as

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{CU}^{-1}) = \arg\min_{\boldsymbol{\delta}}\ n\mathbf{g}_n(\boldsymbol{\delta})'\hat{\mathbf{S}}^{-1}(\boldsymbol{\delta})\mathbf{g}_n(\boldsymbol{\delta}) \tag{1.14}$$

where the expression for $\hat{\mathbf{S}}(\boldsymbol{\delta})$ depends on the estimator used for $\mathbf{S}$. For example, with conditionally heteroskedastic errors $\hat{\mathbf{S}}(\delta)$ takes the form

$$\hat{\mathbf{S}}(\boldsymbol{\delta}) = \frac{1}{n}\sum_{t=1}^{n} \mathbf{x}_t\mathbf{x}_t'(y_t - \mathbf{z}_t'\boldsymbol{\delta})^2$$

[2] In the function `GMM`, the default initial weight matrix is the identity matrix. This can be changed by supplying a weight matrix using the optional argument `w=my.weight.matrix`. Using $\hat{\mathbf{W}} = \mathbf{S}_{xx}^{-1}$ is often more numerically stable than using $\hat{\mathbf{W}} = \mathbf{I}_k$.

Hansen, Heaton and Yaron (1996) call $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{CU}^{-1})$ the *continuous updating* (CU) *efficient GMM estimator*. This estimator is asymptotically equivalent to the two-step and iterated estimators, but may differ in finite samples. The CU efficient GMM estimator does not depend on an initial weight matrix $\mathbf{W}$, and, like the iterated efficient GMM estimator, the numerical value of CU estimator is invariant to the scale of the data. It is computationally more burdensome than the iterated estimator, especially for large nonlinear models, and is more prone to numerical instability. However, Hansen, Heaton and Yaron find that the finite sample performance of the CU estimator, and test statistics based on it, is often superior to the other estimators. The good finite sample performance of the CU estimator relative to the iterated GMM estimator may be explained by the connection between the CU estimator and empirical likelihood estimators. See Imbens (2002) and Newey and Smith (2004) for further discussion on the relationship between GMM estimators and empirical likelihood estimators.

The J-Statistic

The *J-statistic*, introduced in Hansen (1982), refers to the value of the GMM objective function evaluated using an efficient GMM estimator:

$$J = J(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1}) = n\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))'\hat{\mathbf{S}}^{-1}\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})) \tag{1.15}$$

where $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})$ denotes any efficient GMM estimator of $\boldsymbol{\delta}$ and $\hat{\mathbf{S}}$ is a consistent estimate of $\mathbf{S}$. If $K = L$ then $J = 0$, and if $K > L$ then $J > 0$. Under regularity conditions (see Hayashi chapter 3) and if the moment conditions (1.2) are valid, then as $n \to \infty$

$$J \xrightarrow{d} \chi^2(K - L)$$

Hence, in a well specified overidentified model with valid moment conditions the $J-$statistic behaves like a chi-square random variable with degrees of freedom equal to the number of overidentifying restrictions. If the model is mis-specified and or some of the moment conditions (1.2) do not hold (e.g., $E[x_{it}\varepsilon_t] = E[x_{it}(y_t - \mathbf{z}_t'\boldsymbol{\delta}_0)] \neq 0$ for some i) then the $J-$statistic will be large relative to a chi-square random variable with $K - L$ degrees of freedom.

The $J-$statistic acts as an omnibus test statistic for model mis-specification. A large $J-$statistic indicates a mis-specified model. Unfortunately, the $J-$statistic does not, by itself, give any information about how the model is mis-specified.

Normalized Moments

If the model is rejected by the $J-$statistic, it is of interested to know why the model is rejected. To aid in the diagnosis of model failure, the magnitudes of the individual elements of the normalized moments $\sqrt{n}\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))$

may point the reason why the model is rejected by the $J-$statistic. Under the null hypothesis that the model is correct and the orthogonality conditions are valid, the *normalized moments* satisfy

$$\sqrt{n}\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})) \xrightarrow{d} N(\mathbf{0}, \mathbf{S} - \mathbf{\Sigma}_{xz}[\mathbf{\Sigma}_{xz}'\mathbf{S}^{-1}\mathbf{\Sigma}_{xz}]^{-1}\mathbf{\Sigma}_{xz}')$$

As a result, for a well specified model the individual moment $t-$ratios

$$t_i = \mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))_i / SE(\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))_i),\ i = 1, \ldots, K \tag{1.16}$$

where

$$SE(\mathbf{g}_n(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))_i = \left(\left[\hat{\mathbf{S}} - \hat{\mathbf{\Sigma}}_{xz}[\hat{\mathbf{\Sigma}}_{xz}'\hat{\mathbf{S}}^{-1}\hat{\mathbf{\Sigma}}_{xz}]^{-1}\hat{\mathbf{\Sigma}}_{xz}' \right] / T \right)_{ii}^{1/2}$$

are asymptotically standard normal. When the model is rejected using the $J-$statistic, a large value of t_i indicates mis-specification with respect to the ith moment condition. Since the rank of $\mathbf{S} - \mathbf{\Sigma}_{xz}[\mathbf{\Sigma}_{xz}'\mathbf{S}^{-1}\mathbf{\Sigma}_{xz}]^{-1}\mathbf{\Sigma}_{xz}'$ is $K - L$, the interpretation of the moment $t-$ratios (1.16) may be difficult in models for which the degree of over-identification is small. In particular, if $K - L = 1$ then $t_1 = \cdots = t_K$.

Two Stage Least Squares as Efficient GMM

If, in the linear GMM regression model (1.1), the errors are conditionally homoskedastic then

$$E[\mathbf{x}_t\mathbf{x}_t'\varepsilon_t^2] = \sigma^2\mathbf{\Sigma}_{xx} = \mathbf{S}$$

A consistent estimate of $\mathbf{S}$ has the form $\hat{\mathbf{S}} = \hat{\sigma}^2\mathbf{S}_{xx}$ where $\hat{\sigma}^2 \xrightarrow{p} \sigma^2$. Typically,

$$\hat{\sigma}^2 = n^{-1}\sum_{t=1}^{n}(y_t - \mathbf{z}_t'\boldsymbol{\delta})^2$$

where $\hat{\boldsymbol{\delta}} \rightarrow \boldsymbol{\delta}_0$. The efficient GMM estimator becomes:

$$\begin{aligned}
\hat{\boldsymbol{\delta}}(\hat{\sigma}^{-2}\mathbf{S}_{xx}^{-1}) &= (\mathbf{S}_{xz}'\sigma^{-2}\mathbf{S}_{xx}^{-1}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\sigma^{-2}\mathbf{S}_{xx}^{-1}\mathbf{S}_{xy} \\
&= (\mathbf{S}_{xz}'\mathbf{S}_{xx}^{-1}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\mathbf{S}_{xx}^{-1}\mathbf{S}_{xy} \\
&= \hat{\boldsymbol{\delta}}(\mathbf{S}_{xx}^{-1})
\end{aligned}$$

which does not depend on $\hat{\sigma}^2$. The estimator $\hat{\boldsymbol{\delta}}(\mathbf{S}_{xx}^{-1})$ is, in fact, identical to the *two stage least squares* (TSLS) estimator of $\boldsymbol{\delta}$:

$$\begin{aligned}
\hat{\boldsymbol{\delta}}(\mathbf{S}_{xx}^{-1}) &= (\mathbf{S}_{xz}'\mathbf{S}_{xx}^{-1}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\mathbf{S}_{xx}^{-1}\mathbf{S}_{xy} \\
&= (\mathbf{Z}'\mathbf{P}_X\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{P}_X\mathbf{y} \\
&= \hat{\boldsymbol{\delta}}_{TSLS}
\end{aligned}$$

where $\mathbf{Z}$ denotes the $n \times L$ matrix of observations with tth row $\mathbf{z}_t'$, $\mathbf{X}$ denotes the $n \times k$ matrix of observations with tth row $\mathbf{x}_t'$, and $\mathbf{P}_X = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is the idempotent matrix that projects onto the columns of $\mathbf{X}$.

Using (1.10), the asymptotic variance of $\hat{\boldsymbol{\delta}}(\mathbf{S}_{xx}^{-1}) = \hat{\boldsymbol{\delta}}_{TSLS}$ is

$$avar(\hat{\boldsymbol{\delta}}_{TSLS}) = (\boldsymbol{\Sigma}_{xz}'\mathbf{S}^{-1}\boldsymbol{\Sigma}_{xz})^{-1} = \sigma^2(\boldsymbol{\Sigma}_{xz}'\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\Sigma}_{xz})^{-1}$$

Although $\hat{\boldsymbol{\delta}}(\mathbf{S}_{xx}^{-1})$ does not depend on $\hat{\sigma}^2$, a consistent estimate of the asymptotic variance does:

$$\widehat{avar}(\hat{\boldsymbol{\delta}}_{TSLS}) = \hat{\sigma}^2(\mathbf{S}_{xz}'\mathbf{S}_{xx}^{-1}\mathbf{S}_{xz})^{-1}$$

Similarly, the J-statistic also depends on $\hat{\sigma}^2$ and takes the form

$$J(\hat{\boldsymbol{\delta}}_{TSLS}, \hat{\sigma}^{-2}\mathbf{S}_{xx}^{-1}) = n\frac{(\mathbf{s}_{xy} - \mathbf{S}_{xz}\hat{\boldsymbol{\delta}}_{TSLS})'\mathbf{S}_{xx}^{-1}(\mathbf{s}_{xy} - \mathbf{S}_{xz}\hat{\boldsymbol{\delta}}_{TSLS})}{\hat{\sigma}^2}$$

The TSLS J-statistics is also known as Sargan's statistic (see Sargan 1958).

1.3 Estimation of **S**

To compute any of the efficient GMM estimators, a consistent estimate of $\mathbf{S} = avar(\bar{\mathbf{g}})$ is required. The method used to estimate $\mathbf{S}$ depends on the time series properties of the population moment conditions $\mathbf{g}_t$. Two cases are generally considered. In the first case, $\mathbf{g}_t$ is assumed to be serially uncorrelated but may be conditionally heteroskedastic. In the second case, $\mathbf{g}_t$ is assumed to be serially correlated as well as potentially conditionally heteroskedastic. The following sections discuss estimation of $\mathbf{S}$ in these two cases. Similar estimators were discussed in the context of linear regression in Chapter 6, Section 5. In what follows, the assumption of a linear model (1.1) is dropped and the K moment conditions embodied in the vector $\mathbf{g}_t$ are assumed to be nonlinear functions of $q \leq K$ model parameters $\boldsymbol{\theta}$ and are denoted $\mathbf{g}_t(\boldsymbol{\theta})$. The moment conditions satisfy $E[\mathbf{g}_t(\boldsymbol{\theta}_0)] = \mathbf{0}$ and $\mathbf{S} = avar(\bar{\mathbf{g}}) = avar(n^{-1}\sum_{i=1}^{n}\mathbf{g}_t(\boldsymbol{\theta}_0))$.

1.3.1 Serially Uncorrelated Moments

In many situations the population moment conditions $\mathbf{g}_t(\boldsymbol{\theta}_0)$ form an ergodic-stationary MDS with an appropriate information set I_t. In this case,

$$\mathbf{S} = avar(\bar{\mathbf{g}}) = E[\mathbf{g}_t(\boldsymbol{\theta}_0)\mathbf{g}_t(\boldsymbol{\theta}_0)']$$

Following White (1982), a *heteroskedasticity consistent* (HC) estimate of $\mathbf{S}$ has the form

$$\hat{\mathbf{S}}_{HC} = \frac{1}{n}\sum_{t=1}^{n}\mathbf{g}_t(\hat{\boldsymbol{\theta}})\mathbf{g}_t(\hat{\boldsymbol{\theta}})' \tag{1.17}$$

where $\hat{\boldsymbol{\theta}}$ is a consistent estimate of ${\boldsymbol{\theta}_0}^3$. Davidson and MacKinnon (1993, section 16.3) suggest using a simple degrees-of-freedom corrected estimate of $\mathbf{S}$ that replaces n^{-1} in (1.17) with $(n-k)^{-1}$ to improve the finite sample performance of tests based on (1.11).

1.3.2 Serially Correlated Moments

If the population moment conditions $\mathbf{g}_t(\boldsymbol{\theta}_0)$ are an ergodic-stationary but serially correlated process then

$$\mathbf{S} = avar(\bar{\mathbf{g}}) = \mathbf{\Gamma}_0 + \sum_{j=1}^{\infty}(\mathbf{\Gamma}_j + \mathbf{\Gamma}_j')$$

where $\mathbf{\Gamma}_j = E[\mathbf{g}_t(\boldsymbol{\theta}_0)\mathbf{g}_{t-j}(\boldsymbol{\theta}_0)']$. In this case a *heteroskedasticity and autocorrelation consistent* (HAC) estimate of $\mathbf{S}$ has the form

$$\hat{\mathbf{S}}_{HAC} = \frac{1}{n}\sum_{j=1}^{n-1} w_{j,n}(\hat{\mathbf{\Gamma}}_j(\hat{\boldsymbol{\theta}}) + \hat{\mathbf{\Gamma}}_j'(\hat{\boldsymbol{\theta}}))$$

where $w_{j,n}$ $(j = 1, \ldots, b_n)$ are kernel function weights, b_n is a non-negative bandwidth parameter that may depend on the sample size, $\hat{\mathbf{\Gamma}}_j(\hat{\boldsymbol{\theta}}) = \frac{1}{n}\sum_{t=j+1}^{n} \mathbf{g}_t(\hat{\boldsymbol{\theta}})\mathbf{g}_{t-j}(\hat{\boldsymbol{\theta}})'$, and $\hat{\boldsymbol{\theta}}$ is a consistent estimate of $\boldsymbol{\theta}_0$. Different HAC estimates of $\mathbf{S}$ are distinguished by their kernel weights and bandwidth parameter. The most common kernel functions are listed in Table 1.1. For all kernels except the quadratic spectral, the integer bandwidth parameter, b_n, acts as a lag truncation parameter and determines how many autocovariance matrices to include when forming $\hat{\mathbf{S}}_{HAC}$. Figure 1.1 illustrates the first ten kernel weights for the kernels listed in Table 1.1 evaluated using the default values of b_n for $n = 100$.The choice of kernel and bandwidth determine the statistical properties of $\hat{\mathbf{S}}_{HAC}$. The truncated kernel is often used if the moment conditions follow a finite order moving average process. However, the resulting estimate of $\mathbf{S}$ is not guaranteed to be positive definite. Use of the Bartlett, Parzen or quadratic spectral kernels ensures that $\hat{\mathbf{S}}_{HAC}$ will be positive semi-definite. For these kernels, Andrews (1991) studied the asymptotic properties $\hat{\mathbf{S}}_{HAC}$. He showed that $\hat{\mathbf{S}}_{HAC}$ is consistent for $\mathbf{S}$ provided that $b_n \to \infty$ as $n \to \infty$. Furthermore, for each kernel, Andrews determined the rate at which $b_n \to \infty$ to asymptotically minimize $MSE(\hat{\mathbf{S}}_{HAC}, \mathbf{S})$. For the Bartlett, Parzen and quadratic spectral kernels the rates are $n^{1/3}$, $n^{1/5}$, and $n^{1/5}$, respectively. Using the optimal bandwidths, Andrews found that the $\hat{\mathbf{S}}_{HAC}$ based on the quadratic spectral kernel has the smallest asymptotic MSE, followed closely by $\hat{\mathbf{S}}_{HAC}$ based on the Parzen kernel.

[3] For example, $\hat{\boldsymbol{\theta}}$ may be an inefficient GMM estimate based on an arbitrary p.d. weight matrix.

Kernel	**Default** b_n	$w_{j,n}$
Truncated	$int\left[4 \cdot \left(\frac{n}{100}\right)^{1/5}\right]$	1 for $a_j < 1$ 0 for $a_j \geq 1$
Bartlett	$int\left[4 \cdot \left(\frac{n}{100}\right)^{1/4}\right]$	$1 - a_j$ for $a_j \leq 1$ 0 for $a_j > 1$
Parzen	$int\left[4 \cdot \left(\frac{n}{100}\right)^{4/25}\right]$	$1 - 6a_j^2 + 6a_j^3$ for $0 \leq a_j \leq .5$ $2(1 - a_j)^3$ for $.5 \leq a_j \leq 1$ 0 for $a_j > 1$
Quadratic spectral	$int\left[4 \cdot \left(\frac{n}{100}\right)^{4/25}\right]$	$\frac{25}{12\pi^2 d_i^2}\left[\frac{\sin(m_j)}{m_j} - \cos(m_j)\right]$
note: $a_j = i/(b_n + 1)$, $d_j = j/b_n$, $m_j = 6\pi d_i/5$		

TABLE 1.1. Common kernel weights and default bandwidths

Automatic Bandwidth Selection

Based on extensive Monte Carlo experiments, Newey and West (1994) conclude that the choice of bandwidth parameter is more important than the choice of kernel for the finite sample performance of $\hat{\mathbf{S}}_{HAC}$. Increasing b_n reduces the bias of $\hat{\mathbf{S}}_{HAC}$ but increases the variance. The default values for b_n in Table 1.1 are ad hoc, being motivated by the convergence rates of $\hat{\mathbf{S}}_{HAC}$ for the respective kernels. To overcome the ad hoc choice of b_n, Andrews (1991) and Newey and West (1994) propose data dependent automatic bandwidth selection procedures that asymptotically minimize $MSE(\hat{\mathbf{S}}_{HAC}, \mathbf{S})$. The details of these procedures are tedious and so are not repeated here. The interested reader is referred to den Haan and Levin (1997), who nicely summarize the procedures and give guidance to the practitioner. However, as den Haan and Levin point out, the so-called "automatic" bandwidth selection procedures still depend on a $K \times 1$ user-specified vector of weights $\mathbf{w}$ for the elements of $\mathbf{g}_t(\hat{\boldsymbol{\theta}})$ and a method of providing initial estimates of $\mathbf{S}$.

Pre-Whitening and Re-coloring

If $\mathbf{g}_t(\hat{\boldsymbol{\theta}})$ is highly autocorrelated, Andrews and Monahan (1992) found that the finite sample behavior of $\hat{\mathbf{S}}_{HAC}$ is often improved if a pre-whitening and re-coloring procedure is used. This procedure works as follows:

1. Pre-whiten $\mathbf{g}_t(\hat{\boldsymbol{\theta}})$ by estimating a VAR(1) for $\mathbf{g}_t(\hat{\boldsymbol{\theta}})$

$$\mathbf{g}_t(\hat{\boldsymbol{\theta}}) = \Phi \mathbf{g}_{t-1}(\hat{\boldsymbol{\theta}}) + e_t(\hat{\boldsymbol{\theta}})$$

   and forming the residuals $e_t(\hat{\boldsymbol{\theta}}) = \mathbf{g}_t(\hat{\boldsymbol{\theta}}) - \hat{\Phi}\mathbf{g}_{t-1}(\hat{\boldsymbol{\theta}})$

2. Construct an HAC estimator for $e_t(\hat{\boldsymbol{\theta}})$:

$$\hat{\boldsymbol{\Sigma}}_{HAC} = \frac{1}{n}\sum_{j=1}^{b_n} w_{j,n}(\hat{\boldsymbol{\Gamma}}_j + \hat{\boldsymbol{\Gamma}}_j')$$

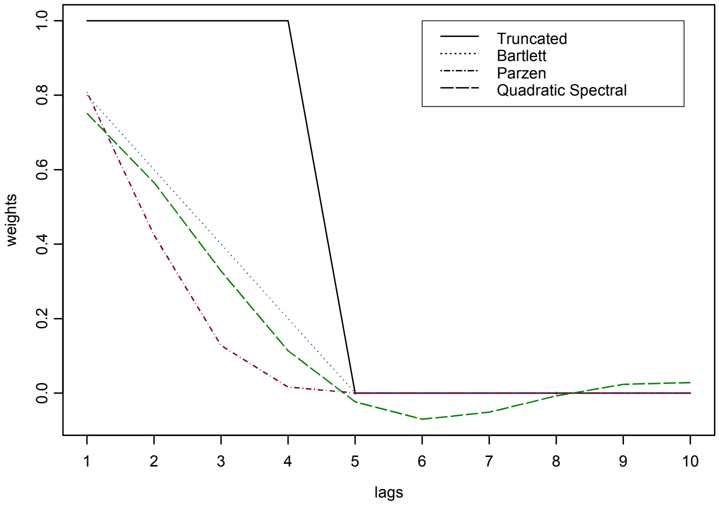

FIGURE 1.1. Kernel weights evaluated at default bandwidths for $n = 100$.

where $\hat{\mathbf{\Gamma}}_j = \frac{1}{n}\sum_{t=j+1}^{n} e_t(\hat{\boldsymbol{\theta}})e_{t-j}(\hat{\boldsymbol{\theta}})$

3. Form the re-colored HAC estimate for $\mathbf{S}$ using

$$\hat{\mathbf{S}}_{PWHAC} = (\mathbf{I}_K - \hat{\Phi})^{-1}\hat{\mathbf{\Sigma}}_{HAC}(\mathbf{I}_K - \hat{\Phi})^{-1\prime}$$

1.3.3 Estimating $\mathbf{S}$ *Using the* ***S+FinMetrics*** *Function* ***var.hac***

HAC estimates of $\mathbf{S}$ for an $n \times k$ time series $\mathbf{x}_t$, based on the procedures described in the previous section, may be computed using the `S+FinMetrics` function `var.hac`[4]. The arguments expected by `var.hac` are

```
> args(var.hac)
function(x, bandwidth = NULL, window = "parzen", na.rm = F,
automatic = "none", df.correction = 0, prewhiten = F,
w = NULL, demean = T)
```

The optional arguments `bandwidth` and `window` are used to specify the bandwidth parameter b_n and kernel weight function $w_{j,n}$, respectively. Valid kernels are those listed in Table 1.1: `truncated`, `bartlett`, `parzen`

[4] The function `var.hac` is an enhanced version of the `S+FinMertics` function `asymp.var`.

and qs. If the bandwidth is not specified, then the default value for b_n from Table 1.1 is used for the specified kernel. The argument df.correction specifies a non-negative integer to be subtracted from the sample size to perform a degrees-of-freedom correction. The argument automatic determines if the Andrews (1991) or Newey-West (1994) automatic bandwidth selection procedure is to be used to set b_n. If automatic="andrews" or automatic="nw" then the argument w must be supplied a vector of weights for each variable in x. The Andrews-Monahan (1992) VAR(1) pre-whitening and re-coloring procedure is performed if prewhiten=T.

1.4 GMM Estimation Using the S+FinMetrics Function GMM

GMM estimation of general linear and nonlinear models may be performed using the S+FinMetrics function GMM. The arguments expected by GMM are

```
> args(GMM)
function(start, moments, control = NULL, scale = NULL, lower
=  - Inf, upper = Inf, trace = T, method =
"iterative", w = NULL, max.steps = 100, w.tol = 0.0001,
ts = F, df.correction = T, var.hac.control =
var.hac.control(), w0.efficient = F, ...)
```

The required arguments are start, which is a vector of starting values for the parameters of the model, and moments, which is an S-PLUS function to compute the sample moment conditions used for estimating the model. The moments function must be of the form f(parm,...), where parm is a vector of L parameters, and return a matrix of dimension $n \times K$ giving the GMM moment conditions $\mathbf{x}_t'\varepsilon_t$ for $t = 1, \ldots, n$. The optional arguments control, scale, lower and upper are used by the S-PLUS optimizer function nlregb. See the online help for nlregb and nlregb.control for details. Setting trace=T, displays iterative information from the optimization. The argument method determines the type of GMM estimation to be performed. Valid choices are "iterative", for iterated GMM estimation, and "simultaneous", for continuous updating GMM estimation. If method="iterative" then the argument max.steps determines the number of iterations to be performed. The argument w specifies the weight matrix used for constructing the GMM objective function (1.6)[5]. If w=NULL, then an estimate of the efficient weight matrix based on the asymptotic variance of the sample moment conditions will be used. In this case, the argument ts determines if an HC or HAC covariance estimator is computed and the arguments df.correction and var.hac.control control various

[5] The current version of GMM uses the inverse of w as the initial weight matrix.

options associated with these estimators. The user may supply a positive definite and symmetric weight matrix to be used as the initial weight matrix if `method="iterative"`. This weight matrix may be fixed throughout the estimation by setting `max.step=0`. If `method="interative"` and `max.step=0` then the argument `w0.efficient` indicates whether the user-supplied weight matrix is an efficient weight matrix. This is useful for computing certain types of test statistics based on the GMM objective function. The argument `...` specifies any optional arguments that will be passed to the `moments` function used for computing the GMM moment conditions. Typically, these arguments specify the data used to compute the moment conditions.

The `GMM` function produces an object of class `GMM` for which there are `print` and `summary` methods, and extractor function `coef`.

Example 1 *Estimating the classical linear regression model by GMM*

Consider the classical linear regression model

$$y_t = \mathbf{x}_t' \boldsymbol{\beta}_0 + \varepsilon_t \tag{1.18}$$

where the explanatory variables $\mathbf{x}_t$ are assumed to be orthogonal to the error term. However, ε_t is allowed to be conditionally heteroskedastic and/or serially correlated. In this model the explanatory variables are also the instrumental variables so that $\mathbf{z}_t = \mathbf{x}_t$ and $K = L$. The population orthogonality condition is

$$E[g_t(\mathbf{w}_t, \boldsymbol{\beta}_0)] = E[\mathbf{x}_t \varepsilon_t] = E[\mathbf{x}_t(y_t - \mathbf{x}_t' \boldsymbol{\beta}_0)] = \mathbf{0}$$

where $g_t(\mathbf{w}_t, \boldsymbol{\beta}) = \mathbf{x}_t \varepsilon_t$ and $\mathbf{w}_t = (y_t, \mathbf{x}_t)'$. The sample moment condition used for estimation is

$$g_n(\mathbf{w}_t, \boldsymbol{\beta}) = \frac{1}{n} \sum_{t=1}^{n} \mathbf{x}_t(y_t - \mathbf{x}_t' \boldsymbol{\beta})$$

which gives rise to the GMM estimating equation

$$\mathbf{S}_{xy} - \mathbf{S}_{xx} \boldsymbol{\beta} = \mathbf{0}$$

Since $K = L$ the model is just identified, and, provided $\mathbf{S}_{xx}$ is invertible, the GMM estimator is equivalent to the least squares estimator

$$\hat{\boldsymbol{\beta}} = \mathbf{S}_{xx}^{-1} \mathbf{S}_{xy}$$

The estimate $\hat{\boldsymbol{\beta}}$ is asymptotically normally distributed with asymptotic variance

$$avar(\hat{\boldsymbol{\beta}}) = (\boldsymbol{\Sigma}_{xx} \mathbf{S}^{-1} \boldsymbol{\Sigma}_{xx})^{-1}$$

where $\mathbf{S} = avar(\bar{\mathbf{g}})$. If ε_t is $iid\ (0, \sigma^2)$, say, then $\mathbf{S} = E[\mathbf{x}_t\mathbf{x}_t'\varepsilon_t^2] = \sigma^2\boldsymbol{\Sigma}_{xx}$ and $avar(\hat{\boldsymbol{\beta}}) = \boldsymbol{\sigma}^2\boldsymbol{\Sigma}_{xx}^{-1}$, which is the usual formula for $avar(\hat{\boldsymbol{\beta}})$ in the classical linear regression formula.

As an example of a simple linear regression model, consider the Capital Asset Pricing Model (CAPM)

$$R_t - r_{ft} = \alpha + \beta(R_{Mt} - r_{ft}) + \varepsilon_t, t = 1, \ldots, n \tag{1.19}$$

where R_t denotes the return on an asset, r_t denotes the risk free rate and R_{Mt} denotes the return on a market portfolio proxy. Using the notation for the linear model (1.18), $y_t = R_t - r_{ft}$ and $x_t = (1, R_{Mt} - r_{ft})'$. The data for this example are the monthly excess returns on Microsoft stock and the S&P 500 index over the period February, 1990 through December, 2000 in the S+FinMetrics "timeSeries" object excessReturns.ts. Assuming the error term is orthogonal to $R_{Mt} - r_{ft}$, the CAPM may be consistently estimated using ordinary least squares

```
> ols.fit = OLS(MSFT~SP500, data = excessReturns.ts)
> ols.fit

Call:
OLS(formula = MSFT ~SP500, data = excessReturns.ts)

Coefficients:
 (Intercept)  SP500
 0.0175      1.5677

Degrees of freedom: 131 total; 129 residual
Time period: from Feb 1990 to Dec 2000
Residual standard error: 0.09094843
```

An S-PLUS function to compute the moment conditions for the linear regression model is

```
ols.moments = function(parm,y=NULL,x=NULL) {
  x = as.matrix(x)
  x*as.vector(y - x%*%parm)
}
```

where parm is an $L \times 1$ vector of parameter $\boldsymbol{\beta}$, y is an $n \times 1$ vector of observations on the dependent variable and x is an $n \times L$ matrix of observations on the explanatory variables. The function returns an $n \times L$ matrix of moment conditions $\mathbf{x}_t'\varepsilon_t = \mathbf{x}_t'(y_t - \mathbf{x}_t'\boldsymbol{\beta})$ for $t = 1, \ldots, n$:

```
> ols.moments(c(1,1),y = excessReturns.df[,"MSFT"],
+              x = cbind(1, excessReturns.df[,"SP500"]))
numeric matrix: 131 rows, 2 columns.
             [,1]          [,2]
```

```
 [1,] -0.9409745 -0.0026713864
 [2,] -0.9027093 -0.0161178959
...
[131,] -1.2480621  0.0009318206
```

To estimate the CAPM regression (1.19) with GMM assuming heteroskedastic errors use

```
> start.vals = c(0,1)
> names(start.vals) = c("alpha", "beta")
> gmm.fit = GMM(start.vals, ols.moments, max.steps = 1,
+           y = excessReturns.df[,"MSFT"],
+           x = cbind(1, excessReturns.df[,"SP500"]))
> class(gmm.fit)
[1] "GMM"
```

Notice how the data are passed to the function `ols.moments` through the ... argument. The object `gmm.fit` returned by `GMM` is of class "`GMM`" and has components

```
> names(gmm.fit)
 [1] "parameters"         "objective"
 [3] "message"            "grad.norm"
 [5] "iterations"         "r.evals"
 [7] "j.evals"            "scale"
 [9] "normalized.moments" "vcov"
[11] "method"             "df.J"
[13] "df.residual"        "call"
```

The online help for `GMM` gives a complete description of these components[6]. Typing the name of the "`GMM`" object invokes the `print` method

```
> gmm.fit

Call:
GMM(start = start.vals, moments = ols.moments, max.steps = 1,
y = excessReturns.df[, "MSFT"], x = cbind(1,
excessReturns.df[, "SP500"]))

Coefficients:
  alpha   beta
 0.0175 1.5677
```

[6] Since the linear model is just identified, the weight matrix in the GMM objective function is irrelevant and so the `weight.matrix` component of `gmm.fit` is not returned. Also, the `moments.vcov` component is not returned since all of the normalized moments are equal to zero.

```
Test of Overidentification:
model is just-identified

Optimization Info:
 Number of Iterations: 1
 Convergence: x convergence
```

As expected, the GMM estimates of α and β are equivalent to the least squares estimates. Also, since the linear model is just identified the GMM objective function ($J-$statistic) is identically zero at the optimum and, therefore, there is no test for overidentifying restrictions.

The `summary` method provides information about the statistical significance of the estimated parameters:

```
> summary(gmm.fit)

Call:
GMM(start = start.vals, moments = ols.moments, max.steps = 1,
y = excessReturns.df[, "MSFT"], x = cbind(1,
excessReturns.df[, "SP500"]))

Coefficients:
       Value Std.Error t value Pr(>|t|)
alpha 0.0175 0.0079    2.2175  0.0283
 beta 1.5677 0.1905    8.2274  0.0000

Test of Overidentification:
model is just-identified

Optimization Info:
 Number of Iterations: 1
 Convergence: x convergence
```

By default, the `GMM` function computes an HC estimate of the asymptotic variance of the sample moment conditions, and so values in the column labeled `Std.Error` are heteroskedasticity consistent (HC) standard errors. To be sure, these standard errors may be compared to those computed from the OLS fit with the White HC correction:

```
> summary(ols.fit,correction="white")

Call:
OLS(formula = MSFT ~SP500, data = excessReturns.ts)

Residuals:
     Min      1Q  Median      3Q     Max
 -0.3101 -0.0620 -0.0024  0.0581  0.2260
```

```
Coefficients:
             Value Std. Error t value Pr(>|t|)
(Intercept) 0.0175 0.0079     2.2175  0.0283
      SP500 1.5677 0.1905     8.2274  0.0000
```

To compute the GMM estimator using an HAC estimate of the asymptotic variance of the sample moment conditions, call the `GMM` function with the optional argument `ts=T`. The type of HAC estimate used is determined by the options set in `var.hac.control`. For example, to compute the OLS estimates with the usual Newey-West HAC standard errors use

```
> gmm.fit2 = GMM(c(0,1), ols.moments, max.steps = 1,
+  y = excessReturns.df[,"MSFT"],
+  x = cbind(1, excessReturns.df[,"SP500"]), ts = T,
+  var.hac.control = var.hac.control(window = "bartlett",
+  bandwidth = floor(4 * (nrow(excessReturns.df)/100)^(2/9))))
```

The standard errors for the GMM estimates above are identical to those returned by

```
> summary(ols.fit,correction="nw")
```

Example 2 *Estimating the instrumental variables regression model using GMM*

As in Campbell and Mankiw (1990), consider the stylized consumption function

$$\begin{aligned}\Delta c_t &= \delta_0 + \delta_1 \Delta y_t + \delta_2 r_t + \varepsilon_t,\ t = 1, \ldots, T \\ &= \boldsymbol{\delta}' \mathbf{z}_t + \varepsilon_t\end{aligned} \tag{1.20}$$

where c_t denotes the log of real per capita consumption (excluding durables), y_t denotes the log of real disposable income, and r_t denotes the ex post real interest rate (T-bill rate - inflation rate). Assume that $\{\Delta c_t, \Delta y_t, r_t\}$ are stationary and ergodic, and that $\{\varepsilon_t, I_t\}$ is a stationary and ergodic martingale difference sequence (MDS) where $I_t = \{\Delta c_s, \Delta y_s, r_s\}_{s=1}^{t}$ denotes the observed information set at time t. In (1.20), the variables Δy_t and r_t are likely to be contemporaneously correlated with ε_t and so the least squares estimates of $\boldsymbol{\delta}$ are likely to be biased and inconsistent. Because $\{\varepsilon_t, I_t\}$ is a stationary and ergodic MDS, $E[\varepsilon_t | I_{t-1}] = 0$ which implies that any variable in I_{t-1} is a potential instrument. Furthermore, for any variable $x_{t-1} \subset I_{t-1}$, $\{x_{t-1}\varepsilon_t\}$ is an uncorrelated sequence.

The data for this example are annual data over the period 1960 - 1995, and are in the “`timeSeries`” object `consump.ts`

```
> colIds(consump.ts)
[1] "GC" "GY" "R3"
```

```
> consump.ts@documentation
[1] "GY = log growth rate of real income"
[2] "GC = log growth rate of real consumption"
[3] "R3 = real 3-month T-bill rate"
[4] "source: Wooldridge (2002), Introduction to"
[5] "Econometrics, 2nd Edition"
[6] "South-Western Thompson"
```

The following data frame `consump` is created for use with the function `GMM`:

```
> nobs = numRows(consump.ts)
> consump = seriesData(consump.ts)
> consump$const = rep(1,nobs)
```

An `S-PLUS` function to compute the linear instrumental variables regression model moment conditions $g(\mathbf{w}_t, \boldsymbol{\delta}) = \mathbf{x}_t(y_t - \mathbf{z}_t'\boldsymbol{\delta})$ for $t = 1, \ldots n$, is

```
iv.moments = function(parm, y, X, Z) {
     # parm = L x 1 vector of parameters
     # y = n x 1 response vector
     # X = n x K matrix of instruments
     # Z = n x L matrix of explanatory variables
     X = as.matrix(X)
     Z = as.matrix(Z)
     X*as.vector(y - Z%*%parm)
}
```

Applying this function to the consumption data with $\boldsymbol{\delta} = (1, 1, 1)'$, $\mathbf{z}_t = (\Delta y_t, r_t, 1)'$ and $\mathbf{x}_t = (\Delta c_{t-1}, \Delta y_{t-1}, r_{t-1}, 1)'$ gives

```
> iv.moments(c(1,1,1),y = consump[2:nobs,1],
+            X = consump[1:(nobs-1),],
+            Z = consump[2:nobs,2:4])
             GC            GY           R3      const
 1 -0.014143206 -0.0070677434 -0.0124922035 -1.0156263
 2 -0.013609181 -0.0154800892 -0.0141370678 -1.0244252
...
35 -0.015828003 -0.0143904735 -0.0175688817 -1.0395788
```

To estimate the consumption function (1.20) by two-step efficient GMM using $\hat{\mathbf{W}} = \mathbf{I}_K$ as the initial weight matrix and assuming conditionally heteroskedastic errors, call `GMM` with the optional arguments `method = "iterative"` and `max.steps = 1`:

```
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.2step = GMM(start.vals, iv.moments,
+                       method = "iterative", max.steps=1,
+                     y = consump[2:nobs,1],
+                     X = consump[1:(nobs-1),],
+                     Z = consump[2:nobs,2:4])
1-step objective = 1.02951e-8
2-step objective = 1.57854
```

A summary of the model fit is

```
> summary(gmm.fit.2step)

Call:
GMM(start = start.vals, moments = iv.moments, method =
"iterative", max.steps = 1, y = consump[2:nobs, 1],
X = consump[1:(nobs - 1),  ], Z = consump[2:nobs,2:4])

Coefficients:
        Value Std.Error t value Pr(>|t|)
   GY  0.6277  0.1500    4.1852  0.0002
   R3 -0.0099  0.0981   -0.1009  0.9202
const  0.0071  0.0037    1.9134  0.0647

Test of Overidentification:
 J-stat Df P.value
 1.5785 1  0.209

Optimization Info:
Number of Iterative Steps: 2
```

The coefficient on Δy_t is 0.6277, with an estimated standard error of 0.1761, and the coefficient on r_t is slightly negative, with an estimated standard error of 0.1156. The J-statistic is 1.5785, and has a p-value of 0.209 based on the chi-square distribution with one degree of freedom. The data appear to support the single over-identifying restriction.

To estimate the consumption function (1.20) by iterated efficient GMM assuming conditionally heteroskedastic errors call `GMM` with `method = "iterative"` and `max.steps` set to a large number[7]:

```
> gmm.fit.iter = GMM(start.vals, iv.moments,
+                    method = "iterative", max.steps = 100,
+                    y = consump[2:nobs,1],
+                    X = consump[1:(nobs-1),],
+                    Z = consump[2:nobs,2:4])
1-step objective = 1.02951e-8
```

[7] Notice that the 1-step objective almost equal to zero. This is caused by using $\hat{\mathbf{W}} = \mathbf{I}_k$ as the initial weight matrix since the scaling of the individual moment conditions is very different. Using $\hat{\mathbf{W}} = \mathbf{S}_{xx}^{-1}$ generally provides a better scaling of the moment conditions.

```
2-step objective = 1.57854
...
13-step objective = 1.85567
```

To compute the continuously updated efficient GMM estimator (1.14), call `GMM` with `method = "simultaneous"`:

```
> start.vals = gmm.fit.iter$parameters
> gmm.fit.cu = GMM(start.vals, iv.moments,
+                  method = "simultaneous",
+                  y = consump[2:nobs,1],
+                  X = consump[1:(nobs-1),],
+                  Z = consump[2:nobs,2:4])
```

Good starting values are important for the CU estimator, and the above estimation uses the iterated GMM estimates as starting values.

Finally, to compute an inefficient 1-step GMM estimator with $\mathbf{W} = \mathbf{I}_4$ use

```
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.1step = GMM(start.vals, iv.moments,
+                     method = "iterative", max.steps = 0,
+                     w = diag(4), w0.efficient = F,
+                     y = consump[2:nobs,1],
+                     X = consump[1:(nobs-1),],
+                     Z = consump[2:nobs,2:4])
1-step objective = 1.02951e-8
1-step objective = 1.02951e-8
Warning messages:
1: Maximum iterative steps exceeded. in: GMM(start.vals,
iv.moments, method = "iterative", max.steps =  ....
2: The J-Statistic is not valid since the weight matrix is
not efficient. in: GMM(start.vals, iv.moments, method
= "iterative", max.steps =  ....
```

Table 1.2 summarizes the different efficient GMM estimators for the parameters in (1.20). The results are very similar across the efficient estimations[8].

Example 3 *Estimating the instrumental variables regression model using TSLS*

[8] To match the default Eviews output for GMM, set the optional argument `df.correction=F`.

<table>
<tr><th colspan="5">$\Delta c_t = \delta_1 + \delta_2 \Delta y_t + \delta_3 r_t + \varepsilon_t$
$\mathbf{x}_t = (1, \Delta c_{t-1}, \Delta y_{t-1}, r_{t-1})'$, $E[\mathbf{x}_t \varepsilon_t] = 0$, $E[\mathbf{x}_t \mathbf{x}_t' \varepsilon_t^2] = \mathbf{S}$</th></tr>
<tr><th>Estimator</th><th>δ_1</th><th>δ_2</th><th>δ_3</th><th>$J - stat$</th></tr>
<tr><td>2-step efficient</td><td>.007</td><td>.627</td><td>−.010</td><td>1.578</td></tr>
<tr><td></td><td>(.004)</td><td>(.150)</td><td>(.098)</td><td>(.209)</td></tr>
<tr><td>Iterated efficient</td><td>.008</td><td>.591</td><td>−.032</td><td>1.855</td></tr>
<tr><td></td><td>(.004)</td><td>(.144)</td><td>(.095)</td><td>(.173)</td></tr>
<tr><td>CU efficient</td><td>.008</td><td>.574</td><td>−.054</td><td>1.747</td></tr>
<tr><td></td><td>(.003)</td><td>(.139)</td><td>.095</td><td>(.186)</td></tr>
<tr><td>1-step inefficient</td><td>.003</td><td>.801</td><td>−.024</td><td>−</td></tr>
<tr><td></td><td>(.005)</td><td>(.223)</td><td>(.116)</td><td>−</td></tr>
</table>

TABLE 1.2. Efficient GMM estimates of the consumption function parameters.

The TSLS estimator of $\boldsymbol{\delta}$ may be computed using the function GMM by supplying the fixed weight matrix $\mathbf{W} = \mathbf{S}_{xx}^{-1}$ as follows[9]

```
> w.tsls = crossprod(consump[1:(nobs-1),])/nobs
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.tsls = GMM(start.vals,iv.moments,
+                    method = "iterative",max.steps = 0,
+                    w = w.tsls,w0.efficient = T,
+                    y = consump[2:nobs,1],
+                    X = consump[1:(nobs-1),],
+                    Z = consump[2:nobs,2:4])
1-step objective = 1.12666e-4
Warning messages:
  Maximum iterative steps exceeded. in: GMM(start.vals,
iv.moments, method = "iterative", max.steps =  ....
> gmm.fit.tsls

Call:
GMM(start = start.vals, moments = iv.moments, method =
"iterative", w = w.tsls, max.steps = 0, w0.efficient
= T, y = consump[2:nobs, 1], X = consump[1:(nobs -1), ],
Z = consump[2:nobs, 2:4])

Coefficients:
      GY      R3   const
  0.5862 -0.0269  0.0081

Test of Overidentification:
```

[9] Recall, the GMM function uses the inverse of w as the weight matrix.

```
 J-stat Df P.value
 0.0001 1  0.9915

Optimization Info:
Number of Iterative Steps: 1
```

The TSLS estimate of $\boldsymbol{\delta}$ is similar to the efficient iterated estimate. The $J-$statistic and the estimate of $avar(\hat{\boldsymbol{\delta}}_{TSLS})$ computed using $\hat{\mathbf{W}} = \hat{\mathbf{S}}_{xx}^{-1}$, however, are not correct since $\mathbf{S}_{xx}^{-1}$ is proportional to the efficient weight matrix. To get the correct values for these quantities a consistent estimate $\hat{\sigma}^2$ of σ^2 is required to form the efficient weight matrix $\hat{\sigma}^2\mathbf{S}_{xx}^{-1}$. This is easily accomplished using

```
# compute TSLS estimate of error variance
> y = as.vector(consump[2:nobs,1])
> X = as.matrix(consump[1:(nobs-1),])
> Z = as.matrix(consump[2:nobs,2:4])
> d.hat = coef(gmm.fit.tsls)
> e.hat = y - Z%*%d.hat
> df = nrow(Z) - ncol(Z)
> s2 = as.numeric(crossprod(e.hat)/df)
# compute correct efficient weight matrix for tsls
# that contains error variance term
> w.tsls2 = crossprod(X)*s2/nobs
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.tsls2 = GMM(start.vals,iv.moments,
+                     method = "iterative",max.steps = 0,
+                     w = w.tsls2,w0.efficient = T,
+                     y = consump[2:nobs,1],
+                     X = consump[1:(nobs-1),],
+                     Z = consump[2:nobs,2:4])
1-step objective = 2.01841

> summary(gmm.fit.tsls2)

Call:
GMM(start = start.vals, moments = iv.moments, method =
"iterative", w = w.tsls2, max.steps = 0, w0.efficient
= T, y = consump[2:nobs, 1], X = consump[1:(nobs -1), ],
Z = consump[2:nobs, 2:4])

Coefficients:
        Value Std.Error t value Pr(>|t|)
   GY  0.5862  0.1327    4.4177  0.0001
   R3 -0.0269  0.0753   -0.3576  0.7230
```

```
const  0.0081  0.0032    2.5285  0.0166

Test of Overidentification:
 J-stat Df P.value
 2.0184 1  0.1554
```

1.5 Hypothesis Testing for Linear Models

The following sections discuss hypothesis testing in linear models estimated by GMM. The main types of hypothesis tests are for coefficient restrictions, overidentification restrictions, subsets of orthogonality restrictions, and instrument relevance. Except for the tests for instrument relevance, the tests extend in atraightforward to nonlinear models estimated by GMM.

1.5.1 Testing Restrictions on Coefficients

Hypothesis testing on coefficients in linear GMM models is surveyed in Newey and West (1987) and nicely summarized in chapter 3 of Hayashi (2000).

Wald Statistics

Wald-type statistics are based on the asymptotic normality of the GMM estimator $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$ for an arbitrary weight matrix $\hat{\mathbf{W}}$. Simple tests on individual coefficients of the form

$$H_0 : \delta_k = \delta_k^0 \tag{1.21}$$

may be conducted using the asymptotic t-ratio

$$t_k = \frac{\hat{\delta}_k(\hat{\mathbf{W}}) - \delta_k^0}{\widehat{SE}(\hat{\delta}_k(\hat{\mathbf{W}}))} \tag{1.22}$$

where $\widehat{SE}(\hat{\delta}_k(\hat{\mathbf{W}}))$ is the square root of the k^{th} diagonal element of (1.9). Under the null hypothesis (1.21), the t-ratio (1.22) has an asymptotic standard normal distribution.

Linear hypotheses of the form

$$H_0 : \mathbf{R}\boldsymbol{\delta} = \mathbf{r} \tag{1.23}$$

where $\mathbf{R}$ is a fixed $q \times L$ matrix of rank q and $\mathbf{r}$ is a fixed $q \times 1$ vector may be tested using the Wald statistic

$$Wald = n(\mathbf{R}\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) - \mathbf{r})' \left[\mathbf{R}\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))\mathbf{R}'\right]^{-1} (\mathbf{R}\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}) - \mathbf{r}) \tag{1.24}$$

where $\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))$ is given by (1.9). Under the null (1.23), the Wald statistic (1.24) has a limiting chi-square distribution with q degrees of freedom.

The Wald statistic (1.24) is valid for any consistent and asymptotically normal GMM estimator $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$ based on an arbitrary symmetric and positive definite weight matrix $\hat{\mathbf{W}} \xrightarrow{p} \mathbf{W}$. Usually, Wald statistics are compute using $\hat{\mathbf{W}} = \hat{\mathbf{S}}^{-1}$ so that $\widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}))$ is given by (1.11).

Nonlinear hypotheses of the form

$$H_0 : \mathbf{a}(\boldsymbol{\delta}_0) = \mathbf{0} \tag{1.25}$$

where $\mathbf{a}(\boldsymbol{\delta}_0) = \mathbf{0}$ imposes q nonlinear restrictions and $\frac{\partial \mathbf{a}(\boldsymbol{\delta}_0)}{\partial \boldsymbol{\delta}'}$ has full rank q, may be tested using the Wald statistic

$$Wald = n\mathbf{a}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))' \left[\frac{\partial \mathbf{a}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))}{\partial \boldsymbol{\delta}'} \widehat{avar}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})) \frac{\partial \mathbf{a}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}}))}{\partial \boldsymbol{\delta}'} \right]^{-1} \mathbf{a}(\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})) \tag{1.26}$$

Under the null (1.25), the Wald statistic (1.26) has a limiting chi-square distribution with q degrees of freedom.

GMM LR-type Statistics

Linear and nonlinear restrictions on the model coefficients $\boldsymbol{\delta}$ may also be tested using a likelihood ratio (LR) type statistic. In efficient GMM estimation, the unrestricted objective function is $J(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1})$, for a consistent estimate $\hat{\mathbf{S}}$ of $\mathbf{S}$. The restricted efficient GMM estimator solves

$$\tilde{\delta}_R(\hat{\mathbf{S}}^{-1}) = \arg\min_{\delta} \; J(\boldsymbol{\delta}, \hat{\mathbf{S}}^{-1}) \text{ subject to } H_0 \tag{1.27}$$

The GMM LR-type statistic is the difference between the restricted and unrestricted J-statistics:

$$LR_{GMM} = J(\tilde{\boldsymbol{\delta}}_R(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1}) - J(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1}) \tag{1.28}$$

Under the null hypotheses (1.23) - (1.25), LR_{GMM} has a limiting chi-square distribution with q degrees of freedom. As $n \rightarrow \infty$, it can be shown that $Wald - LR_{GMM} \xrightarrow{p} 0$, although the two statistics may differ in finite samples. For linear restrictions, $Wald$ and LR_{GMM} are numerically equivalent provided the same value of $\hat{\mathbf{S}}$ is used to compute the restricted and unrestricted efficient GMM estimator. Typically $\hat{\mathbf{S}}$ computed under the unrestricted model is used in constructing LR_{GMM}. In this case, when the restricted efficient GMM estimator is computed by solving (1.27) the weight matrix $\hat{\mathbf{S}}_{UR}^{-1}$ is held fixed during the estimation (no iteration is performed on the weight matrix). If LR_{GMM} is computed using two different consistent estimates of $\mathbf{S}$, say $\hat{\mathbf{S}}$ and $\tilde{\mathbf{S}}$, then it is not guaranteed to be positive in finite samples but is asymptotically valid. The LR_{GMM} statistic has the advantage over the Wald statistic for nonlinear hypotheses in that it is invariant to how the nonlinear restrictions are represented. Additionally, Monte Carlo studies have shown that LR_{GMM} often performs better than

$Wald$ in finite samples. In particular, $Wald$ tends to over reject the null when it is true.

Example 4 *Testing the PIH*

The pure permanent income hypothesis (PIH) due to Hall (1980) states that c_t in (1.20) is a martingale so that $\Delta c_t = \varepsilon_t$ is a MDS. Hence, the PIH implies the linear restrictions

$$H_0 : \delta_1 = \delta_2 = 0$$

which are of the form (1.23) with

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \ \mathbf{r} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

If there are temporary income consumers then $\delta_1 > 0$.

The Wald statistic (1.24) based on the iterated efficient GMM estimator may be computed using

```
> Rmat = matrix(c(1,0,0,1,0,0),2,3)
> rvec = c(0,0)
> dhat = coef(gmm.fit.iter)
> avarRbhat = Rmat%*%gmm.fit.iter$vcov%*%t(Rmat)
> Rmr = Rmat%*%dhat - rvec
> wald.stat = as.numeric(t(Rmr)%*%solve(avarRbhat)%*%Rmr)
> wald.stat
[1] 16.99482
```

Since there are $q = 2$ linear restrictions, the Wald statistic has an asymptotic chi-square distribution with 2 degrees of freedom. The p-value is

```
> 1 - pchisq(wald.stat,2)
[1] 0.0002039964
```

which suggests rejecting the PIH at any reasonable level of significance.

To compute the GMM LR-type statistic (1.28), one must compute restricted and unrestricted GMM estimates using the same estimate $\hat{\mathbf{S}}^{-1}$of the efficient weight matrix and evaluate the corresponding $J-$statistics. Consider computing (1.28) using the iterated efficient GMM estimate as the unrestricted estimate. Its $J-$ statistic is $J(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1}) = 1.855$. To compute the restricted efficient GMM estimate, using $\hat{\mathbf{S}}^{-1}$ from the unrestricted iterated efficient estimation, use

```
> s.ur = solve(gmm.fit.iter$weight.matrix)
> s.ur = (s.ur + t(s.ur))/2
> start.vals = 0.1
> names(start.vals) = c("const")
> gmm.fit.r = GMM(start.vals, iv.moments,
+                 method = "iterative", max.steps = 0,
+                 w = s.ur, w0.efficient = T,
+                 y = consump[2:nobs,1],
+                 X = consump[1:(nobs-1),],
+                 Z = consump[2:nobs,4])
1-step objective = 18.8505
```

The second line above is used to ensure that the weight matrix passed to `GMM` is symmetric. The restricted model is specified using the function `iv.moments` with $z_t = (1)$, and $\mathbf{x}_t = (\Delta c_{t-1}, \Delta y_{t-1}, r_{t-1}, 1)'$. The restricted fit is given by

```
> summary(gmm.fit.r)

Call:
GMM(start = start.vals, moments = iv.moments, method =
"iterative", w = s.ur, max.steps = 0, w0.efficient = T,
y = consump[2:nobs, 1], X = consump[1:(nobs - 1),],
Z = consump[2:nobs, 4])

Coefficients:
        Value Std.Error t value Pr(>|t|)
const  0.0209  0.0012   17.0687  0.0000

Test of Overidentification:
  J-stat Df P.value
 18.8505 3  0.0003

Optimization Info:
Number of Iterative Steps: 1
```

The restricted $J-$statistic is $J(\tilde{\boldsymbol{\delta}}_R(\hat{\mathbf{S}}^{-1}), \hat{\mathbf{S}}^{-1}) = 18.8505$. The GMM-LR statistic is then

```
> gmm.lr = gmm.fit.r$objective - gmm.fit.iter$objective
> gmm.lr
[1] 16.99482
```

which is numerically identical to the Wald statistic computed earlier.

1.5.2 Testing Subsets of Orthogonality Conditions

Consider the linear GMM model (1.1) with instruments $\mathbf{x}_t = (\mathbf{x}'_{1t}, \mathbf{x}'_{2t})'$ such that $\mathbf{x}_{1t}$ is $K_1 \times 1$ and $\mathbf{x}_{2t}$ is $K_2 \times 1$ with $K_1 \geq L$ and $K_1 + K_2 = K$. The instruments $\mathbf{x}_{1t}$ are assumed to be valid (i.e., $E[\mathbf{x}_{1t}\varepsilon_t] = 0$) whereas the instruments $\mathbf{x}_{2t}$ are suspected not to be valid (i.e., $E[\mathbf{x}_{2t}\varepsilon_t] \neq 0$). A procedure to test for the validity of $\mathbf{x}_{2t}$ due to Newey (1985) is as follows.

First, estimate (1.1) by efficient GMM using the full set of instruments $\mathbf{x}_t$ giving

$$\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{Full}^{-1}) = (\mathbf{S}_{xz}'\hat{\mathbf{S}}_{Full}^{-1}\mathbf{S}_{xz})^{-1}\mathbf{S}_{xz}'\hat{\mathbf{S}}_{Full}^{-1}\mathbf{S}_{xy}$$

where

$$\hat{\mathbf{S}}_{Full} = \begin{bmatrix} \hat{\mathbf{S}}_{11,Full} & \hat{\mathbf{S}}_{12,Full} \\ \hat{\mathbf{S}}_{21,Full} & \hat{\mathbf{S}}_{22,Full} \end{bmatrix}$$

such that $\hat{\mathbf{S}}_{11,Full}$ is $K_1 \times K_1$. Second, estimate (1.1) by efficient GMM using only the instruments $\mathbf{x}_{1t}$ and using the weight matrix $\hat{\mathbf{S}}_{11,Full}^{-1}$ giving

$$\tilde{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{11,Full}^{-1}) = (\mathbf{S}_{x_1z}'\hat{\mathbf{S}}_{11,Full}^{-1}\mathbf{S}_{x_1z})^{-1}\mathbf{S}_{x_1z}'\hat{\mathbf{S}}_{11,Full}^{-1}\mathbf{S}_{x_1y}$$

Next, form the statistic[10]

$$C = J(\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{Full}^{-1}), \hat{\mathbf{S}}_{Full}^{-1}) - J(\tilde{\boldsymbol{\delta}}(\hat{\mathbf{S}}_{11,Full}^{-1}), \hat{\mathbf{S}}_{11,Full}^{-1}) \tag{1.29}$$

Under the null hypothesis that $E[\mathbf{x}_t\varepsilon_t] = \mathbf{0}$, the statistic C has a limiting chi-square distribution with $K - K_1$ degrees of freedom.

Example 5 *Testing the endogeneity of r_t in the consumption function*

Consider testing the hypothesis that r_t in (1.20) is exogenous ($E[r_t\varepsilon_t] = 0$). In this case the full set of instruments is $\mathbf{x}_t = (\Delta c_{t-1}, \Delta y_{t-1}, r_{t-1}, 1, r_t)'$, and the reduced set is $\mathbf{x}_{1t} = (\Delta c_{t-1}, \Delta y_{t-1}, r_{t-1}, 1)'$. Efficient GMM estimation of the full model is achieved using

```
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.full = GMM(start.vals,iv.moments,
+                   method = "iterative", max.steps = 100,
+                   y = consump[2:nobs,"GC"],
+                   X = cbind(consump[1:(nobs-1),],
+                             consump[2:nobs,"R3"]),
+                   Z = consump[2:nobs,2:4])
```

The efficient weight matrix $\hat{\mathbf{S}}_{11,Full}^{-1}$ may be extracted using

```
> w11.full = solve(gmm.fit.full$weight.matrix[1:4,1:4])
> w11.full = (w11.full + t(w11.full))/2
```

Efficient GMM estimation using $\mathbf{x}_{1t}$ together with $\hat{\mathbf{S}}_{11,Full}^{-1}$ may be computed using

```
> start.vals = rep(0.1,3)
> names(start.vals) = c("GY","R3","const")
> gmm.fit.11 = GMM(start.vals,iv.moments,
```

[10] The use of $\hat{\mathbf{S}}_{11,Full}^{-1}$ guarantees that the C statistic is non-negative.

```
+                   method = "iterative", max.steps = 0,
+                   w = w11.full, w0.efficient = T,
+                   y = consump[2:nobs,"GC"],
+                   X = consump[1:(nobs-1),],
+                   Z = consump[2:nobs,2:4])
```

The C statistic (1.29) for testing the exogeneity of r_t is then

```
> C.stat = gmm.fit.full$objective - gmm.fit.11$objective
> C.stat
[1] 0.01821106
```

Since $K = 5$ and $K_1 = 4$, C has an limiting chi-square distribution with 1 degree of freedom. The p-value for the test

```
> 1 - pchisq(C.stat,1)
[1] 0.8926527
```

indicates that r_t may be treated as exogenous in the consumption function.

1.5.3 Testing Instrument Relevance

In order to obtain consistent GMM estimates, the instruments $\mathbf{x}_t$ must be uncorrelated with the error term ε_t (valid instruments), and they must be correlated with the endogenous variables $\mathbf{z}_t$ (relevant instruments). The subset orthogonality tests of the previous section can be used to test instrument validity. This section discusses some simple *tests for instrument relevance.*

Instrument relevance is related to the rank condition (1.4). To see this, consider the simple GMM regression involving a single endogenous variable and a single instrument

$$\begin{aligned} y_t &= z_t\delta + \varepsilon_t \\ E[x_t\varepsilon_t] &= 0 \end{aligned}$$

The rank condition (1.4) reduces to $rank(\Sigma_{zx}) = 1$, which implies that $\Sigma_{zx} \neq 0$. Assuming that both z_t and x_t are demeaned, the rank condition can be restated as $cov(x, z) = \Sigma_{zx} \neq 0$. Hence the rank condition will be satisfied as long as x is correlated with z. If there are K instruments $x_{1t}, \ldots, x_{Kt}$ but only one endogenous variable z_t then the rank condition holds as long as $cov(x_k, z) \neq 0$ *for some* k. If $cov(x_k, z) \approx 0$ for all k then the instruments are called *weak.*

Testing instrument relevance is important in practice because recent research (e.g., Stock and Wright (2000)) has shown that standard GMM procedures for estimation and inference may be highly misleading if instruments are weak. If instruments are found to be weak then non-standard methods of inference should be used for constucting confidence intervals and performing hypothesis tests. Stock, Wright and Yogo (2002) give a nice survey of the issues associated with using GMM in the presence of weak instruments, and discuss the non-standard inference procedures that should be used.

In the general linear GMM regression (1.1), the relevance of the set of instruments $\mathbf{x}_t$ for each endogenous variable in $\mathbf{z}_t$ can be tested as follows. First, let $\mathbf{z}_{1t}$ denote the $L_1 \times 1$ vector of non-constant endogenous variables in $\mathbf{z}_t$, and let $\mathbf{x}_{1t}$ denote the $K_1 \times 1$ remaining deterministic or exogenous variables in $\mathbf{z}_t$ such that $\mathbf{z}_t = (\mathbf{z}_{1t}', \mathbf{x}_{1t}')'$ and $L_1 + K_1 = L$. Similarly, define $\mathbf{x}_{2t}$ as the $K_2 \times 1$ vector of exogenous variables that are excluded from $\mathbf{z}_{t.}$ so that $\mathbf{x}_t = (\mathbf{x}_{1t}', \mathbf{x}_{2t}')'$ and $K_1 + K_2 = K$. What is important for the rank condition are the correlations between the endogenous variables in $\mathbf{z}_{1t}$ and the instruments in $\mathbf{x}_{2t}$. To measure these correlations and to test for instrument relevance, estimate by least squares the so-called first stage regression

$$z_{1lt} = \mathbf{x}_{1t}'\boldsymbol{\pi}_{1l} + \mathbf{x}_{2t}\boldsymbol{\pi}_{2l} + v_{lt},\ l = 1, \ldots, L_1$$

for each endogenous variable in $\mathbf{z}_{1t}$. The t-ratios on the variables in $\mathbf{x}_{2t}$ can be used to assess the strength of the correlation between z_{1lt} and the variables in $\mathbf{x}_{2t}$. The F-statistic for testing $\boldsymbol{\pi}_{2l} = \mathbf{0}$ can be used to assess the joint relevance of $\mathbf{x}_{2t}$ for z_{1lt}.

Example 6 *Testing instrument relevance in the consumption function*

In the consumption function regression, $\mathbf{z}_{1t} = (\Delta y_t, r_t)'$, $x_{1t} = (1)$, and $\mathbf{x}_{2t} = (\Delta c_{t-1}, \Delta y_{t-1}, r_{t-1})'$. The first stage regressions for Δy_t and r_t may be computed simultaneously using the S+FinMetrics function OLS as follows

```
> firstStage.fit = OLS(cbind(GY,R3) ~ tslag(GC) + tslag(GY)
+                        + tslag(R3), data = consump)
> class(firstStage.fit)
[1] "mOLS"
```

When a multivariate response is specified in the call to OLS, a regression is performed for each response variable and the returned object is of class “mOLS”. A summary of each first stage regression is

```
> summary(firstStage.fit)
Response: GY

Call:
OLS(formula = cbind(GY, R3) ~tslag(GC) + tslag(GY) + tslag(
R3), data = consump)

Residuals:
     Min      1Q  Median      3Q     Max
 -0.0305 -0.0122 -0.0021  0.0112  0.0349
```

```
Coefficients:
              Value Std. Error t value Pr(>|t|)
(Intercept)  0.0067  0.0055     1.2323  0.2271
  tslag(GC)  1.2345  0.3955     3.1214  0.0039
  tslag(GY) -0.5226  0.2781    -1.8787  0.0697
  tslag(R3)  0.0847  0.1395     0.6069  0.5483

Regression Diagnostics:

         R-Squared 0.2791
Adjusted R-Squared 0.2093
Durbin-Watson Stat 1.8808

Residual Diagnostics:
              Stat P-Value
Jarque-Bera 0.6181 0.7341
  Ljung-Box 8.9289 0.8812

Residual standard error: 0.0163 on 31 degrees of freedom
F-statistic: 4 on 3 and 31 degrees of freedom, the p-value
is 0.01617

Response: R3

Call:
OLS(formula = cbind(GY, R3) ~tslag(GC) + tslag(GY) + tslag(
R3), data = consump)

Residuals:
     Min      1Q  Median      3Q     Max
 -0.0264 -0.0069  0.0021  0.0068  0.0436

Coefficients:
              Value Std. Error t value Pr(>|t|)
(Intercept)  0.0083  0.0044     1.8987  0.0669
  tslag(GC)  0.1645  0.3167     0.5192  0.6073
  tslag(GY) -0.4290  0.2228    -1.9259  0.0633
  tslag(R3)  0.8496  0.1117     7.6049  0.0000

Regression Diagnostics:

         R-Squared 0.6519
Adjusted R-Squared 0.6182
Durbin-Watson Stat 1.7470
```

```
Residual Diagnostics:
               Stat P-Value
Jarque-Bera 14.8961  0.0006
  Ljung-Box 20.7512  0.1450

Residual standard error: 0.01305 on 31 degrees of freedom
F-statistic: 19.35 on 3 and 31 degrees of freedom, the p-value
is 2.943e-007
```

In the first stage regression for Δy_t, the t-ratios for Δc_{t-1} and Δy_{t-1} are significant at the 1% and 10% levels, respectively, indicating that these variables are correlated with Δy_t. The F-statistic for testing the joint significance of the variables in $\mathbf{x}_{2t}$ is 4, with a p-value of 0.0167 indicating that $\mathbf{x}_{2t}$ is relevant for Δy_t.

1.6 Nonlinear GMM

Nonlinear GMM estimation occurs when the K GMM moment conditions $g(\mathbf{w}_t, \boldsymbol{\theta})$ are nonlinear functions of the p model parameters $\boldsymbol{\theta}$. Depending on the model, the moment conditions $g(\mathbf{w}_t, \boldsymbol{\theta})$ may be $K \geq p$ nonlinear functions satisfying

$$E[g(\mathbf{w}_t, \boldsymbol{\theta}_0)] = \mathbf{0} \tag{1.30}$$

Alternatively, for a response variable y_t, L explanatory variables $\mathbf{z}_t$, and K instruments $\mathbf{x}_t$, the model may define a nonlinear error term ε_t

$$a(y_t, \mathbf{z}_t; \boldsymbol{\theta}_0) = \varepsilon_t$$

such that

$$E[\varepsilon_t] = E[a(y_t, \mathbf{z}_t; \boldsymbol{\theta}_0)] = 0$$

Given that $\mathbf{x}_t$ is orthogonal to ε_t, define $g(\mathbf{w}_t, \boldsymbol{\theta}_0) = \mathbf{x}_t \varepsilon_t = \mathbf{x}_t a(y_t, \mathbf{z}_t; \boldsymbol{\theta}_0)$ so that

$$E[g(\mathbf{w}_t, \boldsymbol{\theta}_0)] = E[\mathbf{x}_t \varepsilon_t] = E[\mathbf{x}_t a(y_t, \mathbf{z}_t; \boldsymbol{\theta}_0)] = \mathbf{0} \tag{1.31}$$

defines the GMM orthogonality conditions.

In general, the GMM moment equations (1.30) and (1.31) produce a system of K nonlinear equations in p unknowns. Identification of $\boldsymbol{\theta}_0$ requires that

$$\begin{aligned} E[g(\mathbf{w}_t, \boldsymbol{\theta}_0)] &= \mathbf{0} \\ E[g(\mathbf{w}_t, \boldsymbol{\theta})] &\neq \mathbf{0} \text{ for } \boldsymbol{\theta} \neq \boldsymbol{\theta}_0 \end{aligned}$$

and the $K \times p$ matrix

$$\mathbf{G} = E\left[\frac{\partial g(\mathbf{w}_t, \boldsymbol{\theta}_0)}{\partial \boldsymbol{\theta}'}\right] \tag{1.32}$$

has full column rank p. The sample moment condition for an arbitrary $\boldsymbol{\theta}$ is

$$g_n(\boldsymbol{\theta}) = n^{-1}\sum_{t=1}^{n} g(\mathbf{w}_t, \boldsymbol{\theta})$$

If $K = p$, then $\boldsymbol{\theta}_0$ is apparently just identified and the GMM objective function is

$$J(\boldsymbol{\theta}) = ng_n(\boldsymbol{\theta})'g_n(\boldsymbol{\theta})$$

which does not depend on a weight matrix. The corresponding GMM estimator is then

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}}\ J(\boldsymbol{\theta})$$

If $K > p$, then $\boldsymbol{\theta}_0$ is apparently overidentified. Let $\hat{\mathbf{W}}$ denote a $K \times K$ symmetric and p.d. weight matrix, possibly dependent on the data, such that $\hat{\mathbf{W}} \xrightarrow{p} \mathbf{W}$ as $n \to \infty$ with $\mathbf{W}$ symmetric and p.d. Then the GMM estimator of $\boldsymbol{\theta}_0$, denoted $\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})$, is defined as

$$\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}) = \arg\min_{\boldsymbol{\theta}}\ J(\boldsymbol{\theta}, \hat{\mathbf{W}}) = ng_n(\boldsymbol{\theta})'\hat{\mathbf{W}}g_n(\boldsymbol{\theta})$$

The efficient GMM estimator uses $\hat{\mathbf{W}} = \hat{\mathbf{S}}^{-1}$ such that $\hat{\mathbf{S}} \xrightarrow{p} \mathbf{S} = avar(\bar{\mathbf{g}})$. As with efficient GMM estimation of linear models, the efficient GMM estimator of nonlinear models may be computed using a two-step, iterated, or continuous updating estimator.

1.6.1 Asymptotic Properties

Under standard regularity conditions (see Hayashi Chapter 7), it can be shown that

$$\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}) \xrightarrow{p} \boldsymbol{\theta}_0$$
$$\sqrt{n}\left(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}) - \boldsymbol{\theta}_0\right) \xrightarrow{d} N(\mathbf{0}, avar(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})))$$

where

$$avar(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})) = (\mathbf{G}'\mathbf{W}\mathbf{G})^{-1}\mathbf{G}'\mathbf{W}\mathbf{S}\mathbf{W}\mathbf{G}(\mathbf{G}'\mathbf{W}\mathbf{G})^{-1} \tag{1.33}$$

and $\mathbf{G}$ is given by (1.32). If $\mathbf{W} = \mathbf{S}^{-1}$ then

$$avar(\hat{\boldsymbol{\theta}}(\hat{\mathbf{S}}^{-1})) = (\mathbf{G}'\mathbf{S}^{-1}\mathbf{G})^{-1} \tag{1.34}$$

Notice that with nonlinear GMM, the expression for $avar(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}))$ is of the same form as in linear GMM except that $\boldsymbol{\Sigma}_{xz} = E[\mathbf{x}_t\mathbf{z}_t']$ is replaced by $\mathbf{G} = E\left[\frac{\partial g(\mathbf{w}_t, \boldsymbol{\theta}_0)}{\partial \boldsymbol{\theta}'}\right]$.

A consistent estimate of $avar(\hat{\boldsymbol{\theta}}(\mathbf{W}))$, denoted $\widehat{avar}(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}))$, may be computed using

$$\widehat{avar}(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})) = (\hat{\mathbf{G}}'\hat{\mathbf{W}}\hat{\mathbf{G}})^{-1}\hat{\mathbf{G}}\hat{\mathbf{W}}\hat{\mathbf{S}}\hat{\mathbf{W}}\hat{\mathbf{G}}(\hat{\mathbf{G}}'\hat{\mathbf{W}}\hat{\mathbf{G}})^{-1} \tag{1.35}$$

where $\hat{\mathbf{S}}$ is a consistent estimate for $\mathbf{S} = avar(\bar{\mathbf{g}})$ and

$$\hat{\mathbf{G}} = \mathbf{G}_n(\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})) = n^{-1}\sum_{t=1}^{n} \frac{\partial g(\mathbf{w}_t, \hat{\boldsymbol{\theta}}(\hat{\mathbf{W}}))}{\partial \boldsymbol{\theta}'}$$

For the efficient GMM estimator, $\hat{\mathbf{W}} = \hat{\mathbf{S}}^{-1}$, and

$$\widehat{avar}(\hat{\boldsymbol{\theta}}(\hat{\mathbf{S}}^{-1})) = (\hat{\mathbf{G}}'\hat{\mathbf{S}}^{-1}\hat{\mathbf{G}})^{-1} \tag{1.36}$$

If $\{\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\theta}_0)\}$ is an ergodic stationary MDS with $E[\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\theta}_0)\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\theta}_0)'] = \mathbf{S}$, then a consistent estimator of $\mathbf{S}$ takes the form

$$n^{-1}\sum_{t=1}^{n} \mathbf{g}_t(\mathbf{w}_t, \hat{\boldsymbol{\theta}})\mathbf{g}_t(\mathbf{w}_t, \hat{\boldsymbol{\theta}})'$$

where $\hat{\boldsymbol{\theta}}$ is any consistent estimator of $\boldsymbol{\theta}_0$. If $\{\mathbf{g}_t(\mathbf{w}_t, \boldsymbol{\theta}_0)\}$ is a serially correlated ergodic stationary process then

$$\mathbf{S} = avar(\bar{\mathbf{g}}) = \boldsymbol{\Gamma}_0 + 2\sum_{j=1}^{\infty}(\boldsymbol{\Gamma}_j + \boldsymbol{\Gamma}_j')$$

and the methods discussed in Section 1.3 may be used to consistently estimate $\mathbf{S}$.

1.6.2 Hypothesis Tests for Nonlinear Models

Most of the GMM test statistics discussed in the context of linear models have analogues in the context of nonlinear GMM. For example, the J-statistic for testing the validity of the K moment conditions (1.31) has the form (1.15) with the efficient GMM estimate $\hat{\boldsymbol{\theta}}(\hat{\mathbf{S}}^{-1})$ in place of $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})$. The asymptotic null distribution of the J-statistic is chi-squared with $K-p$ degrees of freedom. Wald statistics for testing linear and non-linear restriction on $\boldsymbol{\theta}$ have the same form as (1.24) and (1.26), with $\hat{\boldsymbol{\theta}}(\hat{\mathbf{W}})$ in place of $\hat{\boldsymbol{\delta}}(\hat{\mathbf{W}})$. Similarly, GMM LR-type statistics for testing linear and non-linear restriction on $\boldsymbol{\theta}$ have the form (1.28), with $\tilde{\boldsymbol{\theta}}(\hat{\mathbf{S}}^{-1})$ in place of $\tilde{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})$ and with $\hat{\boldsymbol{\theta}}(\hat{\mathbf{S}}^{-1})$ in place of $\hat{\boldsymbol{\delta}}(\hat{\mathbf{S}}^{-1})$. Testing the relevance of instruments in nonlinear models, however, is not as straightforward as it is in linear models. In nonlinear models, instruments are relevant if the $K \times p$ matrix $\mathbf{G}$ defined in (1.32) has full column rank p. Testing this condition is problematic because $\mathbf{G}$ depends on $\boldsymbol{\theta}_0$ which is unknown. See Wright (2001) for an approach that can be used to test for instrument relevance in nonlinear models.

1.7 Examples of Nonlinear Models

The following sections give detailed examples of estimating nonlinear models using GMM.

1.7.1 Student-t Distribution

As in Hamilton (1994, chapter 14), consider a random sample $y_1, \ldots, y_T$ that is drawn from a centered Student-t distribution with θ_0 degrees of freedom. The density of y_t has the form

$$f(y_t; \theta_0) = \frac{\Gamma[(\theta_0+1)/2]}{(\pi\theta_0)^{1/2}\Gamma(\theta_0/2)}[1 + (y_t^2/\theta_0)]^{-(\theta_0+1)/2}$$

where $\Gamma(\cdot)$ is the gamma function. The goal is to estimate the degrees of freedom parameter θ_0 by GMM using the moment conditions

$$\begin{aligned} E[y_t^2] &= \frac{\theta_0}{\theta_0 - 2} \\ E[y_t^4] &= \frac{3\theta_0^2}{(\theta_0-2)(\theta_0-4)} \end{aligned}$$

which require $\theta_0 > 4$. Let $\mathbf{w}_t = (y_t^2, y_t^4)'$ and define

$$\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}) = \begin{pmatrix} y_t^2 - \theta/(\theta-2) \\ y_t^4 - 3\theta^2/(\theta-2)(\theta-4) \end{pmatrix} \tag{1.37}$$

Then $E[\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)] = \mathbf{0}$ is the moment condition used for defining the GMM estimator for θ_0. Here, $K = 2$ and $p = 1$ so θ_0 is apparently overidentified. Using the sample moments

$$\mathbf{g}_n(\boldsymbol{\theta}) = \frac{1}{n}\sum_{t=1}^{n} \mathbf{g}(\mathbf{w}_t, \theta) = \begin{pmatrix} \frac{1}{n}\sum_{t=1}^{n} y_t^2 - \theta/(\theta-2) \\ \frac{1}{n}\sum_{t=1}^{n} y_t^4 - 3\theta^2/(\theta-2)(\theta-4) \end{pmatrix}$$

the GMM objective function has the form

$$J(\theta) = n\mathbf{g}_n(\theta)'\hat{\mathbf{W}}\mathbf{g}_n(\theta)$$

where $\hat{\mathbf{W}}$ is a 2×2 p.d. and symmetric weight matrix, possibly dependent on the data, such that $\hat{\mathbf{W}} \xrightarrow{p} \mathbf{W}$. The efficient GMM estimator uses the weight matrix $\hat{\mathbf{S}}^{-1}$ such that $\hat{\mathbf{S}} \xrightarrow{p} \mathbf{S} = E[\mathbf{g}(\mathbf{w}_t, \theta_0)\mathbf{g}(\mathbf{w}_t, \theta_0)']$.

A random sample of $n = 250$ observations from a centered Student-t distribution with $\theta_0 = 10$ degrees of freedom may be generated using the S-PLUS function rt as follows

```
> set.seed(123)
> y = rt(250,df = 10)
```

Basic summary statistics, computed using

```
> summaryStats(y)

Sample Quantiles:
    min      1Q  median     3Q   max
 -4.387 -0.6582 -0.0673 0.6886 3.924

Sample Moments:
     mean   std skewness kurtosis
 -0.03566 1.128  -0.3792    4.659

Number of Observations:  250
```

indicate that the data are roughly symmetric about zero and have thicker tails than the normal distribution.

An S-PLUS function to compute the moment condition (1.37) for $t = 1, \ldots, n$ is

```
t.moments <- function(parm,data=NULL) {
     # parm = df parameter
     # data = [y^2, y^4] is assumed to be a matrix
     m1 = parm/(parm - 2)
     m2 = 3*parm*parm/((parm - 2)*(parm - 4))
     t(t(data) - c(m1,m2))
}
```

The function t.moments has arguments parm, specifying the degrees of freedom parameter θ, and data, specifying an $n \times 2$ matrix with tth row $\mathbf{w}_t = (y_t^2, y_t^4)'$.

To compute the iterated efficient GMM estimator of the degrees of freedom parameter θ from the simulated Student-t data use

```
> y = y - mean(y)
> t.data = cbind(y^2,y^4)
> start.vals = 15
> names(start.vals) = c("theta")
> t.gmm.iter = GMM(start.vals, t.moments,
+                  method = "iterative", max.steps = 100,
+                  data = t.data)
1-step objective = 0.471416
2-step objective = 0.302495
3-step objective = 0.302467
> summary(t.gmm.iter)

Call:
GMM(start = start.vals, moments = t.moments, method =
"iterative", max.steps = 100, data = data)

Coefficients:
       Value Std.Error t value Pr(>|t|)
theta 7.8150 1.1230    6.9592  0.0000

Test of Overidentification:
 J-stat Df P.value
 0.3025 1  0.5823

Optimization Info:
Number of Iterative Steps: 3
```

The iterated efficient GMM estimate of θ_0 is 7.8150, with an asymptotic standard error of 1.123. The small J-statistic indicates a correctly specified model.

1.7.2 MA(1) Model

Following Harris (1999), consider GMM estimation of the parameters in the MA(1) model

$$\begin{aligned} y_t = \mu_0 + \varepsilon_t + \psi_0 \varepsilon_{t-1},\ t = 1, \ldots, n \\ \varepsilon_t \sim iid\ (0, \sigma_0^2),\ |\psi_0| < 1 \\ \boldsymbol{\theta}_0 = (\mu_0, \psi_0, \sigma_0^2)' \end{aligned}$$

Some population moment equations that can be used for GMM estimation are

$$\begin{aligned} E[y_t] &= \mu_0 \\ E[y_t^2] &= \mu_0^2 + \sigma_0^2(1 + \psi_0^2) \\ E[y_t y_{t-1}] &= \mu_0^2 + \sigma_0^2 \psi_0 \\ E[y_t y_{t-2}] &= \mu_0^2 \end{aligned}$$

Let $\mathbf{w}_t = (y_t, y_t^2, y_t y_{t-1}, y_t y_{t-2})'$ and define the moment vector

$$\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}) = \begin{pmatrix} y_t - \mu \\ y_t^2 - \mu^2 - \sigma^2(1 + \psi^2) \\ y_t y_{t-1} - \mu^2 - \sigma^2 \psi \\ y_t y_{t-2} - \mu^2 \end{pmatrix} \tag{1.38}$$

Then

$$E[\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)] = \mathbf{0}$$

is the population moment condition used for GMM estimation of the model parameters $\boldsymbol{\theta}_0$. The sample moments are

$$\mathbf{g}_n(\boldsymbol{\theta}) = \frac{1}{n-2}\sum_{t=3}^{n} g(\mathbf{w}_t, \boldsymbol{\theta}) = \begin{pmatrix} \frac{1}{n-2}\sum_{t=3}^{n} y_t - \mu \\ \frac{1}{n-2}\sum_{t=3}^{n} y_t^2 - \mu^2 - \sigma^2(1+\psi^2) \\ \frac{1}{n-2}\sum_{t=3}^{n} y_t y_{t-1} - \mu^2 - \sigma^2\psi \\ \frac{1}{n-2}\sum_{t=3}^{n} y_t y_{t-2} - \mu^2 \end{pmatrix}$$

Since the number of moment conditions $K = 4$ is greater than the number of model parameters $p = 3$, $\boldsymbol{\theta}_0$ is apparently overidentified and the efficient GMM objective function has the form

$$J(\boldsymbol{\theta}) = (n-2)\cdot \mathbf{g}_n(\boldsymbol{\theta})'\hat{\mathbf{S}}^{-1}\mathbf{g}_n(\boldsymbol{\theta})$$

where $\hat{\mathbf{S}}$ is a consistent estimate of $\mathbf{S} = avar(\bar{\mathbf{g}}(\boldsymbol{\theta}_0))$. Notice that the process $\{\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)\}$ will be autocorrelated (at least at lag 1) since y_t follows an MA(1) process. As a result, an HAC type estimator must be used to estimate $\mathbf{S}$.

Simulated MA(1) data with $\boldsymbol{\theta}_0 = (0, 0.5, 1)'$ and $n = 250$ is computed using the S-PLUS function arima.sim[11]

```
> set.seed(123)
> ma1.sim = arima.sim(model = list(ma=-0.5),n=250)
```

This data along with the sample ACF and PACF are illustrated in Figure 1.2.

Summary statistics for the simulated are

```
> summaryStats(ma1.sim)

Sample Quantiles:
    min      1Q median     3Q   max
 -2.606 -0.6466 0.1901 0.8755 3.221

Sample Moments:
   mean   std skewness kurtosis
 0.1126 1.071  -0.0624    2.634

Number of Observations:  250
```

An S-PLUS function to compute the moment conditions (1.38) is[12]

[11] Recall, the S-PLUS function arima.sim reverses the sign of the moving average parameter ψ.

[12] In the function ma1.moments the parameters are unrestricted. To force the moving average parameter to satisfy $|\psi| < 1$ use the logistic transformation $\psi = \exp(\gamma_1)/(1 + \exp(\gamma_1))$, and to force the variance parameter to be positive use $\sigma^2 = \exp(\gamma_2)$.

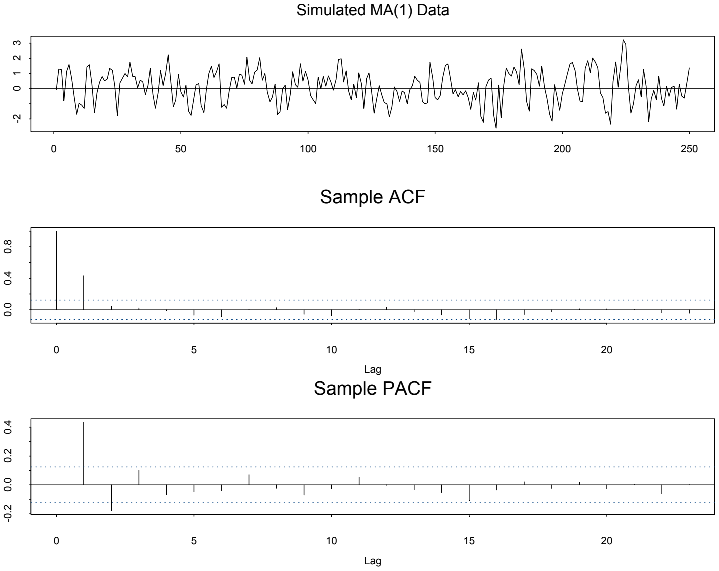

FIGURE 1.2. Simulated data, SACF and SPACF from MA(1) model with $\boldsymbol{\theta} = (0.5, 1)'$.

```
ma1.moments <- function(parm, data = NULL) {
  # parm = (mu,psi,sig2)'
  # data = (y(t),y(t)^2,y(t)*y(t-1),y(t)*y(t-2))
  m1 = parm[1]
  m2 = parm[1]^2 + parm[3]*(1 + parm[2]^2)
  m3 = parm[1]^2 + parm[3]*parm[2]
  m4 = parm[1]^2
  t(t(data) - c(m1,m2,m3,m4))
}
```

The function `ma1.moments` has arguments `parm`, specifying the model parameters $\boldsymbol{\theta} = (\mu, \psi, \sigma^2)'$ and `data`, specifying an $(n-2) \times 4$ matrix with tth row $\mathbf{w}_t = (y_t, y_t^2, yy_{t-1}, y_t y_{t-2})'$. The first five rows of $\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)$ are

```
> ma1.data = cbind(ma1.sim[3:nobs],ma1.sim[3:nobs]^2,
+                  ma1.sim[3:nobs]*ma1.sim[2:(nobs-1)],
+                  ma1.sim[3:nobs]*ma1.sim[1:(nobs-2)])
> start.vals = c(0,0.5,1)
> names(start.vals) = c("mu","psi","sig2")
> ma1.mom = ma1.moments(parm = start.vals, data = ma1.data)
> ma1.mom[1:5,]
         [,1]     [,2]     [,3]      [,4]
[1,]  1.24643  0.303579  1.10981 -0.071482
```

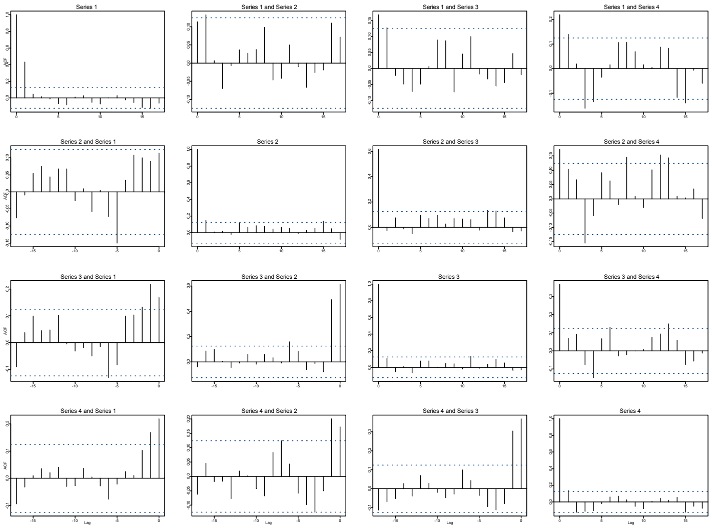

FIGURE 1.3. SACF of $\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)$ from MA(1) model.

```
[2,] -0.80526 -0.601549 -1.50370 -1.040035
[3,]  1.13258  0.032744 -1.41203  1.411681
[4,]  1.58545  1.263659  1.29566 -1.276709
[5,]  0.67989 -0.787743  0.57794  0.770037
```

The sample average of $\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)$ is

```
> colMeans(ma1.mom)
[1]  0.10852675 -0.09134586  0.01198312  0.05919433
```

which is somewhat close to the population value $E[\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)] = \mathbf{0}$. The sample autocorrelations and cross autocorrelations of $\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta}_0)$ are shown in Figure 1.3, which confirm the need for an HAC type estimator for $\mathbf{S}$.

To estimate the MA(1) model by GMM with $\mathbf{S}$ estimated using a truncated (rectangular) kernel with bandwidth equal to one lag use[13]

```
> start.vals = c(0,0.5,1)
> names(start.vals) = c("mu","psi","sig2")
> ma1.gmm.trunc = GMM(start.vals, ma1.moments,
+         data = ma1.data,ts=T,
+         var.hac.control = var.hac.control(bandwidth = 1,
+         window = "truncated"))
```

[13] Recall, $\hat{\mathbf{S}}$ computed with a truncated kernel is not guaranteed to be positive definite.

```
1-step objective = 0.530132
2-step objective = 0.354946
3-step objective = 0.354926
```

The fitted results are

```
> summary(ma1.gmm.trunc)

Call:
GMM(start = start.vals, moments = ma1.moments, ts = T,
var.hac.control = var.hac.control(bandwidth = 1,
window = "truncated"), data = ma1.data)

Coefficients:
       Value Std.Error t value Pr(>|t|)
  mu  0.1018  0.0927    1.0980  0.2733
 psi  0.5471  0.0807    6.7788  0.0000
sig2  0.8671  0.0763   11.3581  0.0000

Test of Overidentification:
 J-stat Df P.value
 0.3549 1  0.5513

Optimization Info:
Number of Iterative Steps: 3
```

The GMM estimate of μ_0 is close to the sample mean, the estimate of ψ_0 is slightly larger than 0.5 and the estimate of σ_0^2 is slightly smaller than 1. The low $J-$statistic indicates a correctly specified model.

To illustrate the impact of model mis-specification on GMM estimation, consider fitting an MA(1) model by GMM to data simulated from an AR(1) model

$$y_t - \mu_0 = \phi_0(y_{t-1} - \mu_0) + \varepsilon_t,\ \varepsilon_t \sim iid\ N(0, \sigma_0^2)$$

The AR(1) model has moments

$$\begin{aligned} E[y_t] &= \mu_0 \\ E[y_t^2] &= \mu_0^2 + \sigma_0^2/(1-\phi_0^2) \\ E[y_t y_{t-1}] &= \mu_0^2 + \phi\sigma_0^2/(1-\phi_0^2) \\ E[y_t y_{t-2}] &= \mu_0^2 + \phi^2\sigma_0^2/(1-\phi_0^2) \end{aligned}$$

Simulated AR(1) data with $\mu_0 = 0$, $\phi_0 = 0.5$, $\sigma_0^2 = 1$ and $n = 250$ is computed using the S-PLUS function `arima.sim`

```
> set.seed(123)
> ar1.sim = arima.sim(model=list(ar=0.5),n=250)
```

The moment data required for GMM estimation of the MA(1) model are

```
> nobs = numRows(ar1.sim)
> ar1.data = cbind(ar1.sim[3:nobs],ar1.sim[3:nobs]^2,
+                  ar1.sim[3:nobs]*ar1.sim[2:(nobs-1)],
+                  ar1.sim[3:nobs]*ar1.sim[1:(nobs-2)])
```

The iterated efficient GMM estimates of the mis-specified MA(1) model are

```
> ma1.gmm.ar1 = GMM(start.vals, ma1.moments,
+     data=ar1.data,ts = T,
+     var.hac.control = var.hac.control(bandwidth = 1,
+     window = "truncated"))
1-step objective = 25.6768
...
9-step objective = 13.0611
> summary(ma1.gmm.ar1, print.moments = T)

Call:
GMM(start = start.vals, moments = ma1.moments, ts = T,
var.hac.control = var.hac.control(bandwidth = 1,
window = "truncated"), data = ar1.data)

Coefficients:
        Value Std.Error  t value Pr(>|t|)
  mu  -0.0407   0.0909   -0.4478   0.6547
 psi   0.4199   0.0789    5.3207   0.0000
sig2   0.7840   0.0737   10.6391   0.0000

Test of Overidentification:
  J-stat Df P.value
 13.0611 1  0.0003

Optimization Info:
Number of Iterative Steps: 9

Normalized Moments:
           Moment Std.Error t value Pr(>|t|)
Moment 1   2.9166 0.8072    3.6131  0.0003
Moment 2   5.3807 1.4888    3.6140  0.0003
Moment 3   5.6389 1.5603    3.6140  0.0003
Moment 4   5.8329 1.6140    3.6140  0.0003
```

The GMM estimates of the MA(1) model parameters look reasonable. However, the large $J-$statistic correctly indicates a mis-specified model. Setting the optional argument `print.moments=T` adds the normalized moment information to the summary output. Since the $J-$statistic is large, all of the normalized moments are significantly different from zero.

1.7.3 Euler Equation Asset Pricing Model

Following Hansen and Singleton (1982), a representative agent is assumed to choose an optimal consumption path by maximizing the present discounted value of lifetime utility from consumption

$$\max \sum_{t=1}^{\infty} E\left[\beta_0^t U(C_t)|I_t\right]$$

subject to the budget constraint

$$C_t + P_t Q_t \leq V_t Q_{t-1} + W_t$$

where I_t denotes the information available at time t, C_t denotes real consumption at t, W_t denotes real labor income at t, P_t denotes the price of a pure discount bond maturing at time $t+1$ that pays V_{t+1}, Q_t represents the quantity of bonds held at t, and β_0 represents a time discount factor. The first order condition for the maximization problem may be represented as the conditional moment equation (Euler equation)

$$E\left[(1+R_{t+1})\beta_0 \frac{U'(C_{t+1})}{U'(C_t)}|I_t\right] - 1 = 0$$

where $1 + R_{t+1} = \frac{V_{t+1}}{V_t}$ is the gross return on the bond at time $t+1$. Assuming utility has the power form

$$U(C) = \frac{C^{1-\alpha_0}}{1-\alpha_0}$$

where α_0 represents the intertemporal rate of substitution (risk aversion), then

$$\frac{U'(C_{t+1})}{U'(C_t)} = \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}$$

and the conditional moment equation becomes

$$E\left[(1+R_{t+1})\beta_0 \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0} |I_t\right] - 1 = 0 \qquad (1.39)$$

Define the nonlinear error term as

$$\begin{aligned} \varepsilon_{t+1} &= a(R_{t+1}, C_{t+1}/C_t; \alpha_0, \beta_0) = (1+R_{t+1})\beta_0 \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0} - 1 \\ &= a(\mathbf{z}_{t+1}, \boldsymbol{\theta}_0) \end{aligned}$$

with $\mathbf{z}_{t+1} = (R_{t+1}, C_{t+1}/C_t)'$ and $\boldsymbol{\theta}_0 = (\alpha_0, \beta_0)'$. Then the conditional moment equation (1.39) may be represented as

$$E[\varepsilon_{t+1}|I_t] = E[a(\mathbf{z}_{t+1}, \boldsymbol{\theta}_0)|I_t] = 0$$

Since $\{\varepsilon_{t+1}, I_{t+1}\}$ is a MDS, potential instruments $\mathbf{x}_t$ include current and lagged values of the elements in $\mathbf{z}_t$ as well as a constant. For example, one could use

$$\mathbf{x}_t = (1, C_t/C_{t-1}, C_{t-1}/C_{t-2}, R_t, R_{t-1})'$$

Since $\mathbf{x}_t \subset I_t$, the conditional moment (1.39) implies that

$$E[\mathbf{x}_t\varepsilon_{t+1}|I_t] = E[\mathbf{x}_t a(\mathbf{z}_{t+1}, \boldsymbol{\theta}_0)|I_t] = \mathbf{0}$$

and by the law of total expectations

$$E[\mathbf{x}_t\varepsilon_{t+1}] = \mathbf{0}$$

For GMM estimation, define the nonlinear residual as

$$e_{t+1} = (1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1$$

and form the vector of moments

$$\begin{aligned} g(\mathbf{w}_{t+1}, \boldsymbol{\theta}) &= \mathbf{x}_t e_{t+1} = \mathbf{x}_t a(\mathbf{z}_{t+1}, \boldsymbol{\theta}) \qquad (1.40) \\ &= \begin{pmatrix} (1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1 \\ (C_t/C_{t-1})\left((1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1\right) \\ (C_{t-1}/C_{t-2})\left((1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1\right) \\ R_t\left((1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1\right) \\ R_{t-1}\left((1 + R_{t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1\right) \end{pmatrix} \end{aligned}$$

In (1.40) there are $K = 5$ moment conditions to identify $L = 2$ model parameters giving $K - L = 3$ overidentifying restrictions. The GMM objective function is

$$J(\boldsymbol{\theta}, \hat{\mathbf{S}}^{-1}) = (n - 2) \cdot g_n(\boldsymbol{\theta})'\hat{\mathbf{S}}^{-1}g_n(\boldsymbol{\theta})$$

where $\hat{\mathbf{S}}$ is a consistent estimate of $\mathbf{S} = avar(\bar{\mathbf{g}})$.

An S-PLUS function to compute the Euler moments (1.40) is

```
euler1.moments <- function(parm, data = NULL) {
   # parm = (beta,gamma)
   # data = (1+R(t+1),C(t+1)/C(t),1,C(t)/C(t-1),
   #          C(t-1)/C(t-2),R(t),R(t-1),...)
```

```
  ncol = numCols(data)
  euler = data[,1]*parm[1]*(data[,2]^(-parm[2])) - 1
  as.matrix(rep(euler,(ncol-2))*data[,3:ncol])
}
```

The function `euler1.moments` has arguments `parm`, specifying the model parameters $\boldsymbol{\theta} = (\beta, \alpha)'$ and `data`, specifying an $(n-2) \times 7$ matrix with tth row $\mathbf{w}_{t+1} = (1 + R_{t+1}, C_{t+1}/C_t, 1, C_t/C_{t-1}, C_{t-1}/C_{t-2}, R_t, R_{t-1})'$.

Verbeek (2000) (chapter 5, section 7) describes an extension of the model presented above that allows the individual to invest in J risky assets with returns $R_{j,t+1}$ $(j = 1, \ldots, J)$, as well as a risk-free asset with certain return $R_{f,t+1}$. Assuming power utility, and restricting attention to unconditional moments (i.e., using $\mathbf{x}_t = 1$), the Euler equations may be written as

$$\begin{aligned} E\left[(1 + R_{f,t+1})\beta_0 \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}\right] - 1 &= 0 && (1.41) \\ E\left[(R_{j,t+1} - R_{f,t+1})\beta_0 \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}\right] &= 0,\ j = 1, \ldots, J \end{aligned}$$

Define the stochastic discount factor as

$$m_{t+1}(\boldsymbol{\theta}_0) = \beta_0 \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}$$

so that the risky assets satisfy

$$E[(R_{j,t+1} - R_{f,t+1}) \cdot m_{t+1}(\boldsymbol{\theta}_0)] = 0$$

Using the identity $E[xy] = cov(x, y) + E[x]E[y]$, the risk premium for the risky assets may be deduced as

$$E[R_{j,t+1} - R_{f,t+1}] = -\frac{cov(m_{t+1}(\boldsymbol{\theta}_0), R_{j,t+1} - R_{f,t+1})}{E[m_{t+1}(\boldsymbol{\theta}_0)]} \qquad (1.42)$$

If the asset pricing model is correct then right-hand-side of (1.42) should explain the cross sectional variation of expected returns across assets.

For GMM estimation define the $J + 1$ vector of moments

$$g(\mathbf{w}_{t+1}, \boldsymbol{\theta}) = \begin{pmatrix} (1 + R_{f,t+1})\beta \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1 \\ (R_{1,t+1} - R_{f,t+1})\beta \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \\ \vdots \\ (R_{J,t+1} - R_{f,t+1})\beta \left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \end{pmatrix}$$

An `S-PLUS` function to compute the above moments is

```
euler2.moments <- function(parm, data = NULL) {
   # parm = (beta,gamma)
   # data = (C(t+1)/C(t),1+Rf(t+1),R1(t+1)-Rf(t+1),...,
   #          RJ(t+1)-Rf(t+1))
   ncol = numCols(data)
   sdf = parm[1]*data[,1]^(-parm[2])
   d1 = (sdf - 1)*data[,2]
   d2 = as.matrix(rep(sdf,(ncol-2))*data[,3:ncol])
   cbind(d1,d2)
}
```

The function `euler2.moments` has arguments `parm`, specifying the model parameters $\boldsymbol{\theta} = (\beta, \alpha)'$ and `data`, specifying an $n \times (J+1)$ matrix with tth row $\mathbf{w}_{t+1} = (C_{t+1}/C_t, 1 + R_{f,t+1}, R_{1,t+1} - R_{f,t+1}, \ldots, R_{J,t+1} - R_{f,t+1})'$.

Following Verbeek (2000), the power utility asset pricing model is estimated using monthly data over the period February 1959 through November 1993[14]. Ten size-based portfolios from the Center for Research in Security Prices (CRSP) data base are used as the risky assets. That is, portfolio 1 contains the monthly returns on the smallest 10% of firms (by market capitalization) listed on the New York Stock Exchange, and portfolio 10 contains the returns on the largest 10% of firms. The risk-free asset is the monthly return on 3-month U.S. T-Bills, and real consumption is measured by total U.S. personal consumption expenditures on nondurables and services. The data is available in the `S+FinMetrics` "`timeSeries`" object `pricing.ts`, which has variables

```
> colIds(pricing.ts)
 [1] "CONS" "R1"   "R2"   "R3"   "R4"   "R5"   "R6"   "R7"
 [9] "R8"   "R9"   "R10"  "RF"
```

The data to be passed to the function `euler2.moments` is constructed using

```
> pricing.mat = as.matrix(seriesData(pricing.ts))
> ncol = numCols(pricing.mat)
> excessRet.mat = apply(pricing.mat[,2:(ncol-1)], 2,
+                       function(x,y){x-y},
+                       pricing.mat[,"RF"])
# data = (C(t+1)/C(t),1+Rf(t+1),R1(t+1)-Rf(t+1),...,
#          RJ(t+1)-Rf(t+1))
> euler.data = cbind(pricing.mat[,"CONS"],
+                    1 + pricing.mat[,"RF"],
+                    excessRet.mat)
```

The iterated efficient GMM estimator may be computed using

[14] The data is taken from Manro Verbeek's web page at `http://www.econ.kuleuven.ac.be/GME`.

```
> start.vals = c(1,5)
> names(start.vals) = c("beta","alpha")
> euler.gmm.fit = GMM(start.vals, euler2.moments,
+                     method = "iterative",data = euler.data)
> summary(euler.gmm.fit)

Call:
GMM(start = start.vals, moments = euler2.moments, method =
"iterative", data = euler.data)

Coefficients:
        Value Std.Error t value Pr(>|t|)
 beta  0.8314  0.1170    7.1049  0.0000
alpha 57.4020 34.3021    1.6734  0.0950

Test of Overidentification:
 J-stat Df P.value
 5.6574 9  0.7737

Optimization Info:
Number of Iterative Steps: 9
```

The small J-statistic indicates a correctly specified model. The estimate of the time discount parameter β is economically reasonable and fairly precise, but the estimate of the risk aversion parameter α is implausibly large and imprecise. The large estimate of α illustrates the well known equity premium puzzle under which the high risk premia on equity assets is associated with extremely risk averse investors.

The 1-step inefficient GMM estimator with $\mathbf{W} = \mathbf{I}_{11}$ may be computed using

```
> euler.gmm.fit2 = GMM(start.vals, euler2.moments,
+                      method = "iterative", max.steps = 0,
+                      w = diag(11), w0.efficient = F,
+                      data = euler.data)
> summary(euler.gmm.fit2)

Call:
GMM(start = start.vals, moments = euler2.moments, method =
"iterative", w = diag(11), max.steps = 0, w0.efficient
= F, data = euler.data)

Coefficients:
        Value Std.Error t value Pr(>|t|)
 beta  0.7031  0.1446    4.8612  0.0000
alpha 91.4097 38.2102    2.3923  0.0172
```

The usual $J-$statistic is not valid for the 1-step estimates so it is not reported. The 1-step estimates are similar to the iterated efficient GMM estimates, and have slightly larger estimated standard errors.

Cochrane (1996, 2001) recommends using the 1-step GMM estimates to compute empirical pricing errors based on (1.42), since these estimates minimize such pricing errors by construction. The pricing errors can then be plotted to evaluate the economic significance of the proposed asset pricing model. The empirical pricing errors may be computed and plotted as follows. First, the historical average excess returns on the ten portfolios are computed using

```
> excessRet.hat = colMeans(excessRet.mat)
```

Next, the average excess returns on the ten portfolios predicted by the model, based on the 1-step inefficient GMM estimates, may be computed using the function

```
predRet <- function(parm,data) {
   # parm = (beta,gamma)
   # data = (C(t+1)/C(t),1+Rf(t+1),R1(t+1)-Rf(t+1),...,
   #          RJ(t+1)-Rf(t+1))
   ncol = numCols(data)
   sdf = parm[1]*data[,1]^(-parm[2])
   tmp <- function(x,y) { -var(x,y)/mean(y) }
   ans = apply(data[,3:ncol], 2, FUN = tmp,sdf)
   ans
}
> predRet.hat = predRet(coef(euler.gmm.fit2),euler.data)
```

Figure 1.4 shows the historical average excess returns and the predicted average excess returns along with a 45^0 line.
If the model is correct then all points should lie on the 45^0 line. The plot shows that the model under-predicts the small firm excess returns and over-predicts the large firm excess returns.

1.7.4 Stochastic Volatility Model

The simple log-normal stochastic volatility (SV) model, due to Taylor (1986), is given by

$$\begin{aligned} y_t &= \sigma_t Z_t,\ t = 1, \ldots, n \qquad (1.43)\\ \ln \sigma_t^2 &= \omega_0 + \beta_0 \ln \sigma_{t-1}^2 + \sigma_{0,u} u_t \\ (Z_t, u_t)' &\sim iid\ N(\mathbf{0}, \mathbf{I}_2) \\ \boldsymbol{\theta}_0 &= (\omega_0, \beta_0, \sigma_{0,u})' \end{aligned}$$

For $0 < \beta_0 < 1$ and $\sigma_{0,u} \geq 0$ the series y_t is strictly stationary and ergodic, and unconditional moments of all orders exist. In the SV model, the series

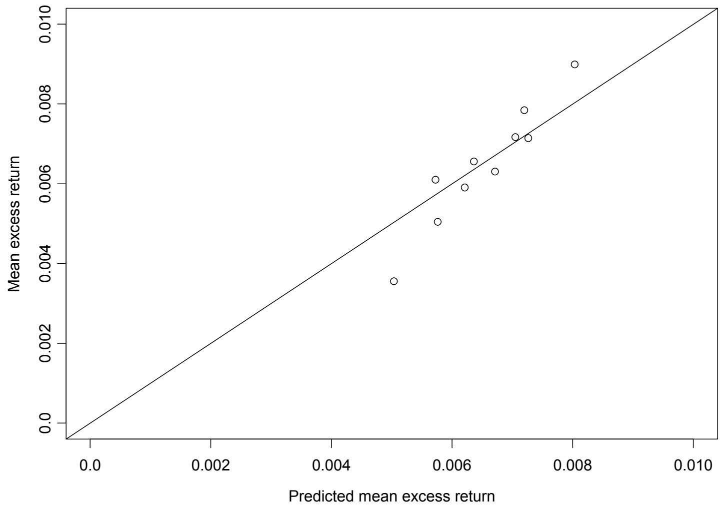

FIGURE 1.4. Monthly actual versus predicted mean excess returns of size-based portfolios.

y_t is serially uncorrelated but dependency in the higher order moments is induced by the serially correlated stochastic volatility term $\ln \sigma_t^2$. Shephard (1996) surveys the use of ARCH and SV models in finance and notes some important advantages of SV models over ARCH models. In particular, their statistical properties are easier to find and understand, they generalize more easily to multivariate settings, and they have simpler continuous-time representations that can be used in contingent claims pricing.

Simulated data from (1.43) with $\omega_0 = -0.736$, $\beta_0 = 0.90$, $\sigma_{0,u} = 0.363$ and $n = 1000$ is illustrated in Figure 1.5.

The data exhibit ARCH-like features such as volatility clustering and fat tails. The data is created with the function `sv.as.gensim`, described in Chapter 22 on EMM estimation, using

```
# beta.t = log(beta/(1-beta))
> n.sim = 1000
> n.burn = 100
> nz = n.sim + n.burn
> set.seed(456)
> sv.as.aux = list(z = rnorm(nz),u = rnorm(nz))
> rho.as = c(-0.736,2.197225,0.363)
> names(rho.as) = c("alpha","beta.t","sigma.u")
> sv.as.sim = sv.as.gensim(rho = rho.as, n.sim = n.sim,
```

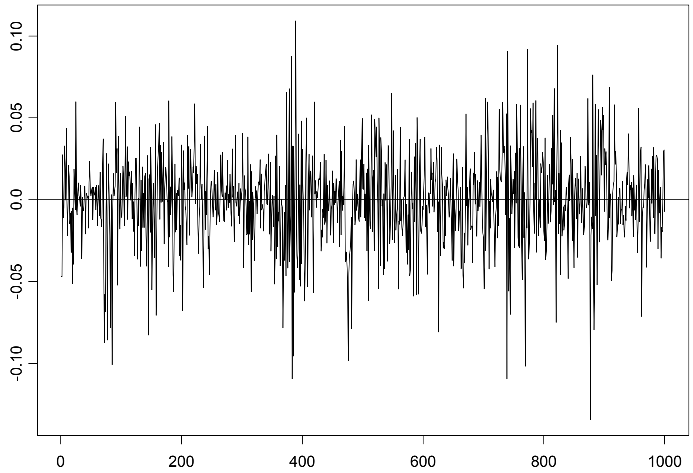

FIGURE 1.5. Simulated data from SV model (1.43) with $\omega_0 = -0.736, \beta_0 = 0.90$ and $\sigma_{0,u} = 0.363$.

```
                           n.burn = n.burn, aux = sv.as.aux)
```

GMM estimation of the SV model is surveyed in Andersen and Sorensen (1996). They recommend using moment conditions for GMM estimation based on lower order moments of y_t, since higher order moments tend to exhibit erratic finite sample behavior. They consider GMM estimation based on (subsets) of 24 moments considered by Jacquier, Polson and Rossi (1994). To describe these moment conditions, first define

$$\mu = \frac{\omega}{1-\beta},\ \sigma^2 = \frac{\sigma_u^2}{1-\beta^2}$$

Then the moment conditions, which follow from properties of the log-normal distribution and the Gaussian AR(1) model, are expressed as

$$\begin{aligned}
E[|y_t|] &= (2/\pi)^{1/2}E[\sigma_t] \\
E[y_t^2] &= E[\sigma_t^2] \\
E[|y_t^3|] &= 2\sqrt{2/\pi}E[\sigma_t^3] \\
E[y_t^4] &= 3E[\sigma_t^4] \\
E[|y_t y_{t-j}|] &= (2/\pi)E[\sigma_t \sigma_{t-j}],\ j = 1,\ldots,10 \\
E[y_t^2 y_{t-j}^2] &= E[\sigma_t^2 \sigma_{t-j}^2],\ j = 1,\ldots,10
\end{aligned}$$

where, for any positive integer j and positive constants r, s,

$$\begin{aligned} E[\sigma_t^r] &= \exp\left(\frac{r\mu}{2} + \frac{r^2\sigma^2}{8}\right) \\ E[\sigma_t^r \sigma_{t-j}^s] &= E[\sigma_t^r]E[\sigma_t^s]\exp\left(\frac{rs\beta^j\sigma^2}{4}\right) \end{aligned}$$

Let

$$\mathbf{w}_t = (|y_t|, y_t^2, |y_t^3|, y_t^4, |y_t y_{t-1}|, \ldots, |y_t y_{t-10}|, y_t^2 y_{t-1}^2, \ldots, y_t^2 y_{t-10}^2)'$$

and define the 24×1 vector

$$g(\mathbf{w}_t, \boldsymbol{\theta}) = \begin{pmatrix} |y_t| - (2/\pi)^{1/2}\exp\left(\frac{\mu}{2} + \frac{\sigma^2}{8}\right) \\ y_t^2 - \exp\left(\mu + \frac{\sigma^2}{2}\right) \\ \vdots \\ y_t^2 y_{t-10}^2 - \exp\left(\mu + \frac{\sigma^2}{2}\right)^2 \exp\left(\beta^{10}\sigma^2\right) \end{pmatrix} \qquad (1.44)$$

Then $E[g(\mathbf{w}_t, \boldsymbol{\theta}_0)] = \mathbf{0}$ is the population moment condition used for GMM estimation of the model parameters $\boldsymbol{\theta}_0$. Since the elements of $\mathbf{w}_t$ are serially correlated, the efficient weight matrix $\mathbf{S} = avar(\bar{\mathbf{g}})$ must be estimated using an HAC estimator.

An `S-PLUS` function to compute the moment conditions (1.44) for $t = 1, \ldots, n$ is

```
sv.moments = function(parm, data = NULL)
{
omega = parm[1]
beta = parm[2]
sigu = parm[3]
mu = omega/(1-beta)
sig2 = (sigu*sigu)/(1-beta*beta)
#
E.sigma = c(sqrt(2/pi) * exp(mu/2 + sig2/8),
            exp(mu + sig2/2),
            2 * sqrt(2/pi) * exp(3*mu/2 + 9*sig2/8),
            3 * exp(2*mu + 2*sig2))
E.sigma.c = c(2/pi * exp(2*(mu/2 + sig2/8)
              + beta^(1:10) * sig2/4),
              exp(2*(mu + sig2/2)
              + 4 * beta^(1:10) * sig2/4))
#
t(t(data) - c(E.sigma, E.sigma.c))
}
```

The transformed simulated data to be passed to the function `sv.moments` for GMM estimation is created using

```
> sv.pow = cbind(abs(sv.as.sim), sv.as.sim^2,
+                abs(sv.as.sim)^3, sv.as.sim^4)
> sv.pow = sv.pow[-(1:10),]
> sv.cm = tslag(sv.as.sim, 1:10, trim=T) *
+               as.vector(sv.as.sim[-(1:10)])
> sv.data = cbind(sv.pow, abs(sv.cm), sv.cm^2)
```

The iterated efficient GMM estimator based on (1.44) may be computed using[15]

```
> start.vals = c(0,0.5,0.5)
> names(start.vals) = c("omega","beta","sigu")
> sv.fit.1 = GMM(start.vals, sv.moments, method = "iterative",
+                ts = T, data = sv.data)
> summary(sv.fit.1)

Call:
GMM(start = start.vals, moments = sv.moments, method =
"iterative", ts = T, data = sv.data)

Coefficients:
         Value Std.Error  t value Pr(>|t|)
omega  -0.4628   0.3731   -1.2405   0.2151
 beta   0.9378   0.0502   18.6959   0.0000
 sigu   0.2233   0.0978    2.2841   0.0226

Test of Overidentification:
  J-stat Df P.value
 18.8761 21 0.5931

Optimization Info:
Number of Iterative Steps: 7
```

The high p-value for the $J-$statistic indicates that the 24 moment conditions are not rejected by the data. Consistent with the findings of Andersen and Sorensen, ω_0 is not estimated very precisely whereas β_0 is estimated fairly precisely.

Now consider estimation of the SV model (1.43) using the daily returns on the S&P 500 index over the period March 14, 1986 through June 30, 2003.

```
> SP500.ts = sp500.ts
```

[15] Discuss choice of starting values.

```
> SP500.ts@data = as.matrix(seriesData(sp500.ts))
>
> SP500.pow = seriesMerge(abs(SP500.ts), SP500.ts^2,
+                         abs(SP500.ts)^3, SP500.ts^4)
> SP500.pow = SP500.pow[-(1:10),]@data
> SP500.cm = tslag(SP500.ts, 1:10, trim=T)@data *
+            as.vector(SP500.ts[-(1:10)]@data)
> SP500.data = cbind(SP500.pow, abs(SP500.cm), SP500.cm^2)
> colIds(SP500.data) = NULL
>
> # iterative GMM estimation of the SV model
> start.vals = c(0,0.5,0.5)
> names(start.vals) = c("omega","beta","sigu")
> sv.fit.SP500 = GMM(start.vals, sv.moments,
+                    method = "iterative", ts = T,
+                    data = SP500.data)
1-step objective = 2.10165e-7
...
9-step objective = 39.9342
> summary(sv.fit.SP500,print.moments=F)

Call:
GMM(start = start.vals, moments = sv.moments, method =
"iterative", ts = T, data = SP500.data)

Coefficients:
         Value Std.Error  t value Pr(>|t|)
omega  -0.1541   0.1758   -0.8767   0.3807
 beta   0.9838   0.0184   53.3692   0.0000
 sigu   0.1548   0.0908    1.7041   0.0884

Test of Overidentification:
  J-stat Df P.value
 39.9342 21 0.0076


Optimization Info:
Number of Iterative Steps: 9
```

The low p-value on the $J-$statistic indicates that the SV model (1.43) does not fit S&P 500 daily returns.

1.7.5 Interest Rate Diffusion Model

Chan, Karolyi, Longstaff and Sanders (1992), hereafter CKLS, considered estimating the parameters of the continuous-time interest rate diffusion model

$$\begin{aligned} dr_t &= (\alpha_0 + \beta_0 r_t)dt + \sigma_0 r_t^{\gamma_0} dW_t \qquad (1.45) \\ \boldsymbol{\theta}_0 &= (\alpha_0, \beta_0, \sigma_0, \gamma_0)' \end{aligned}$$

using GMM. In (1.45), the drift function $(\alpha_0+\beta_0 r_t)dt$ may be re-parameterized as $\kappa_0(\mu_0 - r_t)dt$ where $\alpha_0 = \kappa_0\mu_0$ and $\beta_0 = -\kappa_0$. The parameter μ_0 is the long-run mean, and the parameter κ_0 determines the speed of mean reversion. The model (1.45) encompasses a variety of models that have been proposed for the short-term interest rate. Simulated solutions from (1.45) based on Euler's method for various models using parameters estimated by CKLS are presented in Chapter 19.

CKLS derived moment conditions for GMM estimation of $\boldsymbol{\theta}_0$ from the Euler discretization

$$\begin{aligned} r_{t+\Delta t} - r_t &= (\alpha_0 + \beta_0 r_t)\Delta t + \sigma_0 r_t^{\gamma_0}\sqrt{\Delta t} z_{t+\Delta t} \\ E[z_{t+\Delta t}] &= 0,\ E[z_{t+\Delta t}^2] = 1 \end{aligned}$$

They defined the true model error as

$$\begin{aligned} \varepsilon_{t+\Delta t} &= a(r_{t+\Delta t} - r_t, r_t; \alpha_0, \beta_0, \sigma_0, \gamma_0) = (r_{t+\Delta t} - r_t) - (\alpha_0 + \beta_0 r_t)\Delta t \\ &= a(\mathbf{z}_{t+\Delta t}, \boldsymbol{\theta}_0) \end{aligned}$$

where $\mathbf{z}_{t+\Delta t} = (r_{t+\Delta t} - r_t, r_t)'$. Letting I_t represent information available at time t, the true error satisfies $E[\varepsilon_{t+\Delta t}|I_t] = 0$. Since $\{\varepsilon_{t+\Delta t}, I_{t+\Delta}\}$ is a MDS, potential instruments $\mathbf{x}_t$ include current and lagged values of the elements of $\mathbf{z}_t$ as well as a constant. As the basis for GMM estimation, CKLS used $\mathbf{x}_t = (1, r_t)'$ as the instrument vector and deduced the following four conditional moments

$$\begin{aligned} E[\varepsilon_{t+\Delta t}|I_t] &= 0,\ E[\varepsilon_{t+\Delta t}^2|I_t] = \sigma_0^2 r_t^{2\gamma_0}\Delta t \\ E[\varepsilon_{t+\Delta t} r_t|I_t] &= 0,\ E[\varepsilon_{t+\Delta t}^2 r_t|I_t] = \sigma_0^2 r_t^{2\gamma_0}\Delta t \cdot r_t \end{aligned}$$

For given values of α and β define the nonlinear residual

$$e_{t+\Delta t} = (r_{t+\Delta t} - r_t) - (\alpha + \beta r_t)\Delta t$$

and, for $\mathbf{w}_{t+\Delta t} = (r_{t+\Delta t} - r_t, r_t, r_t^2)'$, define the 4×1 vector of moments

$$g(\mathbf{w}_{t+\Delta t}, \boldsymbol{\theta}) = \begin{pmatrix} e_{t+\Delta t} \\ e_{t+\Delta t}^2 \end{pmatrix} \otimes \mathbf{x}_t = \begin{pmatrix} e_{t+\Delta t} \\ e_{t+\Delta t} r_t \\ e_{t+\Delta t}^2 - \sigma^2 r_t^{2\gamma}\Delta t \\ \left(e_{t+\Delta t}^2 - \sigma^2 r_t^{2\gamma}\Delta t\right) r_t \end{pmatrix}$$

Then $E[g(\mathbf{w}_{t+\Delta t}, \boldsymbol{\theta}_0)] = \mathbf{0}$ gives the GMM estimating equation. Even though $\{\varepsilon_t, I_t\}$ is a MDS, the moment vector $g(\mathbf{w}_t, \boldsymbol{\theta}_0)$ is likely to be autocorrelated since it contains ε_t^2. For the most general specification (1.45), $K = L$ and the model is just identified. For all models listed in Table **??** from Chapter 19, $K > L$, so that they are overidentified.

An S-PLUS function to compute the CKLS moments is

```
ckls.moments <- function(parm, data = NULL,dt = 1/12) {
   # parm = (alpha,beta,sigma,gamma)'
   # data = [r(t+dt)-r(t),r(t)]
   # dt = discretization step
   e.hat = as.vector(data[,1] -
          (parm[1] + parm[2]*data[,2])*dt)
   m2 = e.hat*as.vector(data[,2])
   m3 = e.hat^2 - dt*parm[3]*parm[3]*
        (as.vector(data[,2])^(2*parm[4]))
   m4 = m3*data[,2]
   cbind(e.hat,m2,m3,m4)
}
```

The function `ckls.moments` has arguments `parm`, specifying the model parameters $\boldsymbol{\theta} = (\alpha, \beta, \sigma, \gamma)'$, `data`, specifying an $n \times 2$ matrix with tth row $(r_{t+\Delta t} - r_t, r_t)$, and `dt`, specifying the discretization increment.

GMM estimation is performed on (1.45) using monthly observations on the U.S. 30-day T-Bill rate over the period June 1964 through November 1989. The same data is analyzed in Cliff (2003), and is illustrated in Figure 1.6.

The data to be passed to the function `ckls.moments` is constructed using

```
> data.ckls.ts = seriesMerge(diff(ckls.ts), tslag(ckls.ts))
> colIds(data.ckls.ts)[1] = "RF.diff"
> data.ckls = as.matrix(seriesData(data.ckls.ts))
```

The summary statistics for Δr_t and r_{t-1}

```
> summaryStats(data.ckls)

Sample Quantiles:
             min       1Q median      3Q     max
RF.diff -0.05813 -0.00267 0.0002 0.00324 0.03285
RF.lag1  0.03127  0.04756 0.0603 0.07883 0.16150

Sample Moments:
           mean      std skewness kurtosis
RF.diff 0.00012 0.008187   -1.347   13.354
RF.lag1 0.06659 0.026375    1.206    4.332
```

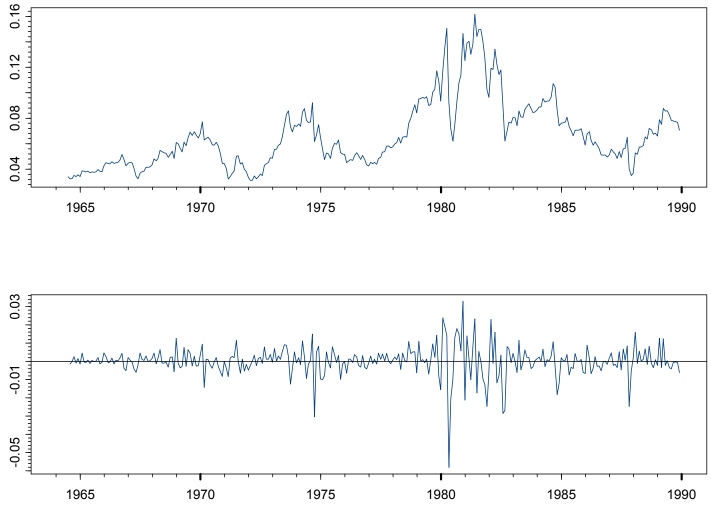

FIGURE 1.6. Monthly observations on the U.S. 30-day T-Bill rate and its first difference.

```
Number of Observations:  305
```

are very close to those presented in Table II of CKLS.

First, consider GMM estimation of the most general model (1.45). Sensible starting values for α and σ may be determined using the following approximations

$$\begin{aligned}\mu &= \frac{\alpha}{-\beta} \approx E[r_t] \\ \sigma^2 &\approx \frac{var(r_{t+\Delta t} - r_t)}{var(r_t)\Delta t}\end{aligned}$$

GMM estimation of $\boldsymbol{\theta}$ may be performed using

```
> start.vals = c(0.06,-0.5,1,1)
> names(start.vals) = c("alpha","beta","sigma","gamma")
> gmm.ckls = GMM(start.vals,ckls.moments,ts = T,
+                 data = data.ckls,dt = 1/12)
> summary(gmm.ckls)

Call:
GMM(start = start.vals, moments = ckls.moments, ts = T, data
= data.ckls, dt = 1/12)
```

```
Coefficients:
        Value Std.Error t value Pr(>|t|)
alpha  0.0419  0.0193    2.1710  0.0307
 beta -0.6077  0.3454   -1.7592  0.0796
sigma  1.3337  1.0004    1.3331  0.1835
gamma  1.5081  0.2900    5.2010  0.0000

Test of Overidentification:
model is just-identified

Optimization Info:
 Number of Iterations: 14
 Convergence: absolute function convergence
```

In the call to GMM, the parameter dt is set to 1/12 because the data are annual rates sampled monthly. Since the model is just identified, the $J-$statistic is zero and the estimates do not depend on a weight matrix. The results are similar to those obtained by CKLS. In particular, the estimate of γ is roughly 1.5 and is highly significant. The estimates of the long-run mean and speed of adjustment are

```
> theta.hat = coef(gmm.ckls)
> theta.hat["alpha"]/-theta.hat["beta"]
      alpha
 0.06895576
> -theta.hat["beta"]
      beta
 0.6076583
```

The GMM estimates for the restricted models in Table ?? of Chapter 19 may be computed in a similar fashion by modifying the function ckls.moments. For example, to estimate the CIR SR model use

```
cir.moments <- function(parm, data = NULL, dt = 1/12) {
   # parm = (alpha,beta,sigma)'
   # data = [r(t+dt)-r(t),r(t)]
   # dt = discretization step
   e.hat = as.vector(data[,1] -
          (parm[1] + parm[2]*data[,2])*dt)
   m2 = e.hat*as.vector(data[,2])
   m3 = e.hat^2 - dt*parm[3]*parm[3]*
        (as.vector(data[,2]))
   m4 = m3*data[,2]
   cbind(e.hat,m2,m3,m4)
}
> start.vals = c(0.06,-0.5,1)
> names(start.vals) = c("alpha","beta","sigma")
> gmm.cir = GMM(start.vals, cir.moments, ts = T,
+               data = data.ckls, dt = 1/12)

> summary(gmm.cir,print.moments = F)

Call:
GMM(start = start.vals, moments = cir.moments, ts = T, data =
data.ckls, dt = 1/12)

Coefficients:
         Value Std.Error  t value Pr(>|t|)
alpha   0.0204   0.0166    1.2258   0.2212
 beta  -0.2407   0.3023   -0.7963   0.4265
sigma   0.0841   0.0066   12.7661   0.0000

Test of Overidentification:
 J-stat Df P.value
 3.7977 1  0.0513


Optimization Info:
Number of Iterative Steps: 13
```

The low p-value on the $J-$statistic indicates a potentially mis-specified model.

1.8 References

[1] Andersen, T.G. and B.E. Sorensen (1996). “GMM Estimation of a Stochastic Volatility Model: A Monte Carlo Study,” *Journal of Business and Economic Statistics*, 14, 328-352.

[2] Brenner, R.J., R.H. Harjes, and K.F. Kroner (1996). “Another Look at Models of the Short-Term Interest Rate,” *Journal of Financial and Quantitative Analysis*, 31, 85-107.

[3] Campbell, J. and G. Mankiw (1990), “Permanent Income, Current Income and Consumption,” *Journal of Business and Economic Statistics*, 8, 265-279.

[4] Chan, K.C., G.A. Karolyi, F.A. Longstaff, and A.B. Sanders (1992). “An Empirical Comparison of Alternative Models of the Term Structure of Interest Rates,” *Journal of Finance*, 47, 1209-1227.

[5] Cliff, M.T. (2003). "GMM and MINZ Program Libraries for MATLAB," unpublished manuscript, Krannert Graduate School of Management, Purdue University.

[6] Cochrane, J.H. (1996). "A Cross-Sectional Test of an Investment-Based Asset Pricing Model," *Journal of Political Economy*, 104, 572-561.

[7] Cochrane, J.H. (2001). *Asset Pricing*. Princeton University Press.

[8] Davidson, R. and J. MacKinnon (2004). *Econometric Theory and Methods*, Oxford University Press.

[9] den Haan, W.J. and A. Levin (1997). "A Practioner's Guide to Robust Covariance Matrix Estimation," in G. Maddala and C. Rao (eds.) *Handbook of Statistics, Volume 15*, 309-327. Elsevier, Amsterdam, The Netherlands.

[10] Ferson, W. E. (1995). "Theory and Empirical Testing of Asset Pricing Models," in R.A. Jarrow, V. Maksimovic, and W.T. Ziemba (eds.), *Handbooks in OR & MS, Volume 9, Finance*, Elsevier Science B.V., Amsterdam.

[11] Hall, A. (2005). *Generalized Method of Moments*, Oxford University Press.

[12] Harris, D. (1999). "GMM Estimation of Time Series Models," chapter 6 in *Generalized Method of Moments*, Mátyás, L. (ed.). Cambridge University Press, Cambridge, UK.

[13] Hamilton, J.D. (1994). *Time Series Analysis*, Princeton University Press.

[14] Hansen, L.P. (1982). "Large Sample Properties of Generalized Method of Moments Estimators," *Econometrica*, 50, 1029-1054.

[15] Hansen, L.P. and K.J. Singleton (1982). "Generalized Instrumental Variables Estimation of Nonlinear Rational Expectations Models," *Econometrica*, 50, 1269-1286.

[16] Hansen, L.P., J. Heaton, and A. Yaron (1996). "Finite-Sample Properties of Some Alternative GMM Estimators," *Journal of Business and Economic Statistics*, 14, 262-280.

[17] Hayashi, F. (2000). *Econometrics*. Princeton University Press, New Jersey.

[18] Imbens, G.W. (2002). "Generalized Method of Moments and Empirical Likelihood," *Journal of Business and Economic Statistics*, 20, 493-506.

[19] Jacquier, E., N.G. Polson, and P.E. Rossi (1994). "Bayesian Analysis of Stochastic Volatility Models," *Journal of Business and Economic Statistics*, 12, 371-389.

[20] Jagannathan, R. and G. Skoulakis (2002). "Generalized Method of Moments: Applications in Finance," *Journal of Business and Economic Statistics*, 20, 470-482.

[21] James, J. and N. Webber (2000). *Interest Rate Models*, John Wiley & Sons, Chichester, England.

[22] Mankiw, G., J. Rotemberg, and L.Summers (1986). "Intertemporal Substitution in Macroeconomics," *Quarterly Journal of Economics.*

[23] Newey, W. (1985). "Generalized Method of Moments Specification Testing," *Journal of Econometrics*, 29, 229-256.

[24] Newey, W. and K. West (1987). "Hypothesis Testing with Efficient Method of Moments," *International Economic Review*, 28, 777-787.

[25] Newey, W. and R Smith (2004). "Higher Order Properties of GMM and Generalized Empirical Likelihood," *Econometrica,* 72, 219-255.

[26] Ogaki, M. (1992). "Generalized Method of Moments: Econometric Applications," in G. Maddala, C. Rao, and H. Vinod (eds.), *Handbook of Statistics, Volume 11: Econometrics*, North-Holland, Amsterdam.

[27] Ruud, P. A. (2000). *An Introduction to Classical Econometric Theory*, Oxford University Press, Oxford.

[28] Sargan, D. (1958). "The Estimation of Economic Relationships Using Instrumental Variables," *Econometrica,* 26, 393-415.

[29] Shephard, N. G., (1996), "Statistical Aspects of ARCH and Stochastic Volatility," in D.R. Cox, D.V. Hinkley and O.E. Barndorff-Nielsen (eds.), *Time Series Models in Econometrics, Finance and Other Fields*, Chapman & Hall, London, England.

[30] Stock, J.H., and J.H. Wright (2000). "GMM with Weak Identification," *Econometrica*, 68, 1055-96.

[31] Stock, J.H., and J.H. Wright and M. Yogo (2002). "GMM, Weak Instruments, and Weak Identification," *Journal of Business and Economic Statistics*, 20.

[32] Taylor, S.J. (1986). *Modelling Financial Time Series*, John Wiley & Sons, Chichester, England.

[33] Verbeek, M. (2000). *A Guide to Modern Econometrics*, John Wiley & Sons, Chichester, England.

[34] Wooldrige, J. (2002). *Introduction to Econometrics*, Southwest.