readme_proc_vbias   August 2016

Kenneth D. West, 2016, “Approximate Bias in Time Series Regression”

This write-up describes some RATS and MATLAB procedures that can be used to compute bias to order T^{-1} in certain least squares regressions. Much much simpler but more restrictive procedures are described in the document “readme_longhor”.

Section 1, section 2 up to “alternative procedures” and section 3 provide (I hope) a self contained description. Additional information is in other sections.

If you want to change the code, see the paper, which writes out the formulas implemented in this code.

1. Overview
2. Procedures:
    2.1 **proc_vb_ma0** (η_t ~ iid)
    2.2 **proc_vb_maq** (η_t ~ MA(q))
  Alternative procedures:
    2.3 **proc_vb_d**
    2.4 **proc_vbias**
3. Examples using 2.1 and 2.2
4. Discussion
5. Population counterparts to variables passed to the procedures
6. A schematic of the hierarchy in the code
7. Code illustrating the procedures

Ken West
kdwest@wisc.edu

1. Overview

The equation of interest is

$$(*) \qquad y_t = \text{const.} + X_{t-1}'\beta + \eta_t,$$

where X_{t-1} is $k\times1$, y_t and η_t are scalars, the unobservable disturbance $\eta_t \sim \mathrm{MA}(q)$ and the dating of y_t and η_t is arbitrary. In terms of dating:

- The lhs variable y_t may be a cumulated sum–say, $y_t = x_t+...+x_{t+q}$ because one is forecasting a cumulated sum using the direct method. Or perhaps $y_t = x_{t+q}$ (again, direct method of forecasting). An allowable special case is q=0 (one step ahead forecasting). Or perhaps y_t is the period $t+q$ realization of a variable other than x_t. See next bullet point.

- In the empirical example described in the final section of this document, the regression is the familiar interest parity regression: the q+1 period change in the exchange rate is regressed on the corresponding q+1 period cross-country interest differential. Thus k=1 and

  - y_t is the q+1 period percentage change in the exchange rate, $y_t = s_{t+q}-s_{t-1} = \Delta s_{t+q}+...+\Delta s_t$; Δs_t is the percentage change in the exchange rate;
  - x_{t-1} is the corresponding q+1 period cross-country interest differential.

Let b be bias to order T^{-1} in the least squares estimator of β,

$$E\hat{\beta} \approx \beta + \frac{b}{T}.$$

The code provided here returns a $k\times1$ estimate $\hat{b}$. The user must divide by sample size T, to obtain

$$\text{bias adjusted estimate} = \hat{\beta} - \frac{\hat{b}}{T}.$$

The user will estimate (*), to obtain $\hat{\beta}$. From West (2016), b depends on certain second moments, such as autocovariances of X_t. Prior to invoking the code described here, the user needs to estimate an AR or VAR that can be used to compute these moments. The lag length of this AR or VAR should be such that the residuals are white noise. The user will pass the estimates of the AR or VAR to the routines described here (details below). As well, the user will need to construct and pass certain moments or parameters related to the cross-covariances between X_t and η_t (again, details below).

The code supplied here assumes that

- η_t follows a moving average process of known order q (serially uncorrelated [q=0] a special case), with $E\eta_t X_{t-j}=0_{k\times1}$ for all $j\geq1$;
- fourth cumulants are zero (roughly, no conditional heteroskedasticity or conditional skew).

Section 3 of the West (2016) provides formulas when η_t has an autoregressive component, when $E\eta_t X_{t-j}\neq0_{k\times1}$ for $j\geq1$ and when cumulants are not zero.

When $X_{t-1} = (x_{t-1}, ..., x_{t-k})'$ consists simply of lags of single variable, the code here is unnecessarily complicated to invoke. See "readme_longhor" for much simpler code to compute bias in that case.

2. Procedures **proc_vb_ma0** and **proc_vb_maq**

The user must estimate and pass (a)estimates of parameters of an AR or VAR used by the routines supplied here to compute autocovariances of X_t; (b)estimate of certain cross-moments between X_t and η_t.

(a)AR/VAR parameters: Consider a simple example one regressor (k=1). In this example, use lower case x_t instead of upper case X_t, so that the regression of interest is

$$y_t = \text{const.} + \beta x_{t-1} + \eta_t,$$

with β a scalar. (This application is simple enough that it could be handled by the procedures described in the document "readme_longhor". I use it for simple illustration.) The user needs to specify a model for x_t that **proc_vb_ma0** and **proc_vb_maq** can use to deliver accurate estimates of the autocovariances of x_t. An autoregression of suitable order will often suffice. The user needs to estimate this autoregression. Suppose an AR(3) produces a white noise residual,

$$x_t = \text{const.} + \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \varphi_3 x_{t-3} + u_t,\ u_t \sim \text{iid}.$$

Write this in companion form as

$$\begin{pmatrix} x_t \\ x_{t-1} \\ x_{t-2} \end{pmatrix} = \text{const.} + \begin{pmatrix} \varphi_1 & \varphi_2 & \varphi_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ x_{t-2} \\ x_{t-3} \end{pmatrix} + \begin{pmatrix} u_t \\ 0 \\ 0 \end{pmatrix}, \text{ written compactly as}$$

$$\underset{3\times 1}{Z_t} = \text{const.} + \underset{3\times 3}{\Phi} \quad \underset{3\times 1}{Z_{t-1}} + \underset{3\times 1}{U_t}.$$

The user must estimate the AR(3), and, among other parameters, the user will pass estimates of $\hat{\Phi}$ and of $\hat{\sigma}_u^2$. The code supplied here will use those estimates to compute autocovariances of x_t.

The general setup: let Z_t be the $n_Z\times 1$ vector of variables in the VAR used to compute moments related to X_t, with the VAR written in companion form. The setup is

(2.0) $$\underset{n_Z\times 1}{Z_t\text{-}EZ_t} = \underset{n_Z\times n_Z}{\Phi}(Z_{t-1}\text{-}EZ_{t-1}) + \underset{n_Z\times 1}{U_t}, \quad \underset{n_Z\times n_Z}{\Omega_U}=EU_tU_t', \qquad \underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\, \underset{n_Z\times 1}{Z_t}.$$

Note that the elements of X_t must also be elements of Z_t (unsurprisingly).

The user must compute and pass to the code: the dimension k of X_t (called **nk** in the code) the dimension of n_Z of Z_t (called **nZtwid** in the code), P_X and estimates $\hat{\Phi}$ and $\hat{\Omega}_U$ (called **PX**, **phitwid**, and **omegaUtwid** in the code). In the example just given, **nk**=1, **nZtwid**=3, **PX** = (1 0 0), and

$$\textbf{phitwid} = \begin{pmatrix} \hat{\varphi}_1 & \hat{\varphi}_2 & \hat{\varphi}_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \textbf{omegaUtwid} = \begin{pmatrix} \hat{\sigma}_u^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \hat{\sigma}_u^2 = T^{-1}\Sigma\hat{u}_t^2.$$

<u>(b)</u>: In terms of cross-covariances between X_t and η_t: I supply separate procedure calls for (1)η_t iid, and (2)$\eta_t \sim$ MA(q). The first is a special case of the second. Let $\hat{\eta}_t$ be the least squares residuals. Let Z_t be

the vector of variables in the companion form VAR.

η_t ~ iid:

The user needs to compute and pass an estimate of $E\eta_t Z_t'$, call it **EetaZtwid0**. In the k=1example just given, in which $Z_t' = (x_t, x_{t-1}, x_{t-2})$,

$$\textbf{EetaZtwid0} = (T^{-1}\sum\hat{\eta}_t x_t,\ T^{-1}\sum\hat{\eta}_t x_{t-1},\ T^{-1}\sum\hat{\eta}_t x_{t-2}).$$

η_t ~ MA(q):

The user passes the integer parameter **nq** (the value of q) as well as

(i)estimates of $E\eta_t Z_t'$, $E\eta_t Z_{t+1}'$, ..., $E\eta_t Z_{t+q}'$, where

$$\text{estimate of } E\eta_t Z_{t+i} = T^{-1}\Sigma\hat{\eta}_t Z_{t+i}'.$$

Let **EetaZtwid** denote this set of q+1 estimates, with each estimate of dimension 1×**nZtwid**. The structure of **EetaZtwid** is given below.

(ii)an estimate of the **nk**×1 vector $E(X_t+X_{t+1}+...+X_{t+q-1})\eta_t$, called **EXeta**, with

$$\textbf{EXeta} = T^{-1}\Sigma(X_t+X_{t+1}+...+X_{t+q-1})\hat{\eta}_t.$$

Here is the RATS and MATLAB syntax to invoke the procedures. The procedures return the **nk**×1 estimate of $\hat{b}$, called **vbias**.

η_t~i.i.d:

(2.1) RATS: execute **proc_vb_ma0** *nZtwid phitwid omegaUtwid nk PX EetaZtwid0 vbias*
MATLAB: [*vbias*] = **proc_vb_ma0**(*nZtwid*, *phitwid*, *omegaUtwid*, *nk*, *PX*, *EetaZtwid0*);

η_t~MA(q):

(2.2) RATS: execute **proc_vb_maq** *nZtwid phitwid omegaUtwid nk PX EetaZtwid EXeta nq vbias*
MATLAB: [*vbias*] = **proc_vb_maq**(*nZtwid*, *phitwid*, *omegaUtwid*, *nk*, *PX*, *EetaZtwid0*, *EXeta*, *nq*);

Relative to the (2.1) procedure used when η_t~i.i.d, the (2.2) procedure for η_t~MA(q) requires that the user change one parameter (**EetaZtwid0** → **EetaZtwid**) and include two additional parameters (**EXeta** and **nq**).

After obtaining **vbias** ≡ $\hat{b}$ from either procedure, the user must divide by sample size T, to obtain

$$\text{bias adjusted estimate} = \hat{\beta} - \frac{\hat{b}}{T}.$$

Here is a list, in alphabetical order, of parameters used in either of the two procedures (2.1) or (2.2). "Matrix" is used to cover data types that in RATS can be declared vector or rectangular. See next page for the RATS data types for each variable.

EXeta	matrix	when $q>0$, $k\times1$ estimate of $E(X_t+X_{t+1}+...+X_{t+q-1})\eta_t$

EetaZtwid RATS: vector of matrices. **EetaZtwid** is a vector of dimension q+1. The i'th entry in **EetaZtwid** is a vector, and contains an estimate of the $1\times n_Z$ matrix $E\eta_t Z_{t+i-1}{}'$:

•Note that the first element of **EetaZtwid** is $T^{-1}\Sigma\hat{\eta}_t Z_t{}'$ and not $T^{-1}\Sigma\hat{\eta}_t Z_{t+1}{}'$, i.e., **EetaZtwid**(1) = $T^{-1}\Sigma\hat{\eta}_t Z_t{}'$, ..., **EetaZtwid**($q$+1) = $T^{-1}\Sigma\hat{\eta}_t Z_{t+q}{}'$.

MATLAB: two dimensional matrix, of dimension $(q+1)\times n_Z$. Row i has the estimate of $E\eta_t Z_{t+i-1}{}'$.

EetaZtwid0	matrix	$1\times n_Z$ estimate of $E\eta_t Z_t{}'$
nk	integer	k = dimension of X_{t-1}: number of regressors, excluding constant term
nq	integer	q = order of MA of η_t
nZtwid	integer	n_Z = dimension of Z_t, where $Z_t - EZ_t = \Phi(Z_{t-1}-EZ_{t-1})+U_t$ and $X_t = P_X Z_t$
omegaUtwid	matrix	$\hat{\Omega}_U = n_Z\times n_Z$ estimate of $\Omega_U = EU_tU_t{}'$
phitwid	matrix	$\hat{\Phi} = n_Z\times n_Z$ estimate of Φ
PX	matrix	$k\times n_Z$; see definition of **nZtwid**
vbias	matrix	$\hat{b} = k\times1$ estimate of bias

Variable	Data type, RATS	Data type, MATLAB	Dimension
EXeta	rectangular	matrix	$1{\times}k$
EetaZtwid	vector[rectangular], **EetaZtwid** has q+1 rows; each row holds a rectangular $1{\times}n_Z$ matrix	matrix, $(q+1){\times}n_Z$	see “Data type” columns
EetaZtwid0	rectangular	matrix	$1{\times}n_Z$
nk	integer	integer	n.a.
nq	integer	integer	n.a.
nZtwid	integer	integer	n.a.
omegaUtwid	rectangular	matrix	$n_Z{\times}n_Z$
phitwid	rectangular	matrix	$n_Z{\times}n_Z$
PX	rectangular	matrix	$k{\times}n_Z$
vbias	vector	matrix	$k{\times}1$

Alternative procedures:

•In writing the code for numerical results in the paper, I found it convenient to allow a second way for the user to pass information necessary to compute cross-covariances between X_t and η_t. I wrote the code to allow this second way because it was (to me) more convenient for calculations with population quantities, while methods 2.1 and 2.2 were sometimes more convenient for calculations using simulated or real data.

Like the earlier procedures, the procedure using the second method invokes procedure **proc_vbias** (file proc_vbias.src (RATS) or proc_vbias.m (MATLAB)) and so **proc_vbias** must be accessible when the procedure is invoked. This alternative procedure

(a)drops **EetaZtwid0**/**EetaZtwid** and **EXeta**,

(b)adds a parameter called **delta**;

(c)does not have distinct procedure calls for η_t~iid and η_t~ MA(q).

For η_t~MA(q), with η_t~iid as a special case accomplished by setting **nq**=0, the call is

(2.3) RATS: execute **proc_vb_d** *nZtwid phitwid omegaUtwid nk PX delta nq vbias*
MATLAB: *vbias* = **proc_vb_d**(*nZtwid*, *phitwid*, *omegaUtwid*, *nk*, *PX*, *delta*, *nq*)

(proc_vb_**d** with “d” as in delta.) The new parameter:

delta RATS: data type vector[rectangular], i.e., a vector of matrices. **delta** is a vector of dimension q+1. The i’th entry is **delta**(i) =$\hat{\delta}_{i\text{-}1}$, for $n_Z\times 1$ $\hat{\delta}_{i\text{-}1}$ defined below. Each of the rectangular matrices is $n_Z\times 1$.
MATLAB: matrix, of dimension $n_Z\times(q+1)$. Column i holds $\hat{\delta}_{i\text{-}1}$, for $n_Z\times 1$ $\hat{\delta}_{i\text{-}1}$ defined below. Data type: matrix.

RATS and MATLAB: For U_t defined in (2.0), let

$$E(\eta_t|U_{t+i}) = \delta_i' U_{t+i}$$

Then

$$\textbf{delta}(i) = \hat{\delta}_{i\text{-}1}.$$

For example, in the k=1, AR(3) example just given,

$$\textbf{delta}(1)=\hat{\delta}_0 = \begin{pmatrix} \hat{\sigma}_{\eta u}/\hat{\sigma}_u^2 \\ 0 \\ 0 \end{pmatrix}; \ \hat{\sigma}_{\eta u} = T^{-1}\Sigma\hat{\eta}_t\hat{u}_t \text{ and } \hat{\sigma}_u^2 = T^{-1}\Sigma\hat{u}_t^2,$$

where $\hat{u}_t$ and $\hat{\eta}_t$ are least squares residuals.

•A fourth and final procedure is **proc_vbias**, invoked as described below. The procedures listed to this point all invoke **proc_vbias** to do the actual work. **proc_vbias** requires the user to flag whether the baseline method in (2.1) and (2.2) is used (**iflag**=1) or the alternative (2.3) (**iflag**=0).

The MATLAB (but <u>not</u> RATS) code for **proc_vbias** requires that **EetaZtwid** and **delta** be passed as three dimensional rather than two dimensional matrices:

> •In proc_vbias.m, **delta** is $n_Z \times 1 \times (q+1)$and **EetaZtwid** is $1 \times n_Z \times (q+1)$. This additional dimension is not used.
> •The information in the first and third columns of **delta** is what is passed to **proc_vb_d** in the $n_Z \times (q+1)$ variable of the same name; the information in second and third columns of **EetaZtwid** are the *transpose* of what is passed to **proc_vb_maq** in the $(q+1) \times n_Z$ variable of the same name. Sorry about this confusion.

Again, the RATS version proc_vbias.src is consistent across routines in definition of **EetaZtwid** and **delta**; it is only the MATLAB version that differs across routines in these definitions.

As well **proc_vbias** returns the following four intermediate quantities computed in the course of computing $\hat{b}$=**vbias**:

Variable	Data type, RATS/MATLAB	Description
bigD	rectangular/matrix	$\hat{D} = k \times k$ estimate of variance-covariance matrix of X_t
gamma0	rectangular/matrix	$\hat{\Gamma}_0 = n_Z \times n_Z$ estimate of variance-covariance matrix of Z_t
vbias1	vector/matrix	$\hat{b}_1 = k \times 1$ estimate of bias associated with first order term in expansion
vbias2	vector/matrix	$\hat{b}_2 = k \times 1$estimate of bias associated with second order term in expansion

These intermediate quantities are probably not of interest in the typical application.

RATS and then MATLAB syntax for calling **proc_vbias**.

(2.4) RATS:

$$\text{execute } \textbf{proc\_vbias} \overbrace{iflag\ \ nZtwid\ \ phitwid\ \ omegaUtwid\ \ nq\ \ nk\ \ nN\ \ PX\ \ EetaZtwid\ \ EXeta\ \ delta}^{passed\ to\ proc\_vbias}\ \$$$

$$\underbrace{bigD\ \ gamma0\ \ vbias1\ \ vbias2\ \ vbias}_{returned\ from\ proc\_vbias}$$

MATLAB:
[*bigD*,*gamma*0,*vbias*1,*vbias*2,*vbias*]=...
proc_vbias(*iflag*, *nZtwid*, *phitwid*, *omegaUtwid*, *nq*, *nk*, *nN*, *PX*, *Eetaztwid*, *EXeta*, *delta*, ...
bigD, *gamma*0, *vbias*1, *vbias*2, *vbias*);

The list of parameters is explained on the next page.

proc_vbias explanation of symbols, with y_t = const. + $X_{t-1}'\beta + \eta_t$, $\eta_t \sim \text{MA}(q)$

iflag	integer	0 (moments related to η_t parameterized in terms of iid driving shocks) or 1 (moments related to η_t not parameterized via iid shocks) see discussion below
nZtwid	integer	n_Z dimension of $\underset{n_Z\times 1}{Z_t}$, where $Z_t\text{-}EZ_t = \Phi(Z_{t-1}\text{-}EZ_{t-1})+U_t$ and $\underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\,\underset{n_Z\times 1}{Z_t}$
phitwid	matrix	$\underset{n_Z\times n_Z}{\hat{\Phi}}$ estimate of Φ
omegaUtwid	matrix	$\underset{n_Z\times n_Z}{\hat{\Omega}_U}$ estimate of $\Omega_U = EU_tU_t'$
nq	integer	q order of MA of η_t
nk	integer	k dimension of X_{t-1}: number of regressors, excluding constant term
nN	integer	1
PX	matrix	P_X see definition of **nZtwid**
EetaZtwid	**iflag**=0 **iflag**=1	n.a. RATS: vector of matrices. **EetaZtwid** is a vector of dimension q+1. The i'th entry in **EetaZtwid** has an estimate of the $1\times n_Z$ matrix $E\eta_tZ_{t+i-1}'$: **EetaZtwid** (1)=estimate of $E\eta_tZ_t'$, ..., **EetaZtwid**(q+1)=estimate of $E\eta_tZ_{t+q}'$. MATLAB: three dimensional matrix. See description above.
EXeta	**iflag**=0 **iflag**=1	n.a. matrix; when q>0, estimate of $E(X_t+X_{t+1}+...+X_{t+q-1})\eta_t$; when q=0: n.a.
delta	**iflag**=0:	RATS: vector of matrices. **delta** is a vector of dimension q+1. The i'th entry is **delta**(i) $=\hat{\delta}_{i-1}$, for $n_Z\times 1$ $\hat{\delta}_{i-1}$ defined above. MATLAB: three dimensional matrix. See description above.
	iflag=1:	n.a.
bigD	matrix	$\underset{k\times k}{\hat{D}}$ estimate of variance-covariance matrix of X_t
gamma0	matrix	$\underset{n_Z\times n_Z}{\hat{\Gamma}_0}$ estimate of variance-covariance matrix of Z_t
vbias1	matrix	$\underset{k\times 1}{\hat{b}_1}$ estimate of bias associated with first order term in expansion
vbias2	matrix	$\underset{k\times 1}{\hat{b}_2}$ estimate of bias associated with second order term in expansion
vbias	matrix	$\underset{k\times 1}{\hat{b}}=\hat{b}_1+\hat{b}_2$ estimate of bias

3. Examples

Examples (3.1) and (3.2) on the following pages both use a scalar (**nk**=k=1) right hand side variable $x_{t\text{-}1}$,

$$y_t = \text{const.} + \beta x_{t\text{-}1} + \eta_t.$$

To keep equations relatively uncluttered, none of the formulas below use degrees of freedom adjustments when computing moments from least squares residuals. It is perfectly fine to do so.

A "^" over a variable indicates a least squares estimate or residual, e. g., $\hat{\beta}$ or $\hat{u}_t$.

3.1 Univariate AR(3) model for x_t used to compute the relevant moments

$$y_t = \text{const.} + \beta x_{t\text{-}1} + \eta_t,$$

$$x_t \sim \text{AR(3)},\ x_t = \text{const.} + \varphi_1 x_{t\text{-}1} + \varphi_2 x_{t\text{-}2} + \varphi_3 x_{t\text{-}3} + u_t,\ u_t \sim \text{iid}$$

This can be written as a companion form VAR as

$$\begin{pmatrix} x_t \\ x_{t-1} \\ x_{t-2} \end{pmatrix} = \text{const.} + \begin{pmatrix} \varphi_1 & \varphi_2 & \varphi_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ x_{t-2} \\ x_{t-3} \end{pmatrix} + \begin{pmatrix} u_t \\ 0 \\ 0 \end{pmatrix}, \text{ written compactly as}$$

$$\underset{3\times 1}{Z_t} = \text{const.} + \underset{3\times 3}{\Phi}\ \underset{3\times 1}{Z_{t\text{-}1}} + \underset{3\times 1}{U_t}$$

Then **nZtwid**=n_Z=3. In terms of other variables passed to **proc_vbias**:

$$\textbf{PX} = P_X = (1\ 0\ 0),\ \textbf{phitwid} = \begin{pmatrix} \hat{\varphi}_1 & \hat{\varphi}_2 & \hat{\varphi}_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix},\ \textbf{omegaUtwid} = \begin{pmatrix} \hat{\sigma}_u^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},\ \hat{\sigma}_u^2 = T^{-1}\Sigma \hat{u}_t^2.$$

A. η_t, u_t ~ jointly iid. Then **nq**=0 and

equation (2.1): $\textbf{EetaZtwid0} = T^{-1}\Sigma\hat{\eta}_t Z_t' = (T^{-1}\sum\hat{\eta}_t x_t,\ T^{-1}\sum\hat{\eta}_t x_{t\text{-}1},\ T^{-1}\sum\hat{\eta}_t x_{t\text{-}2})$.

•Technical note: even though the iid variable η_t is assumed uncorrelated with $x_{t\text{-}2}$ in population, in the sample $T^{-1}\Sigma\hat{\eta}_t x_{t\text{-}2}$ will in general be nonzero. Of course it is okay to set this term to zero, though of course under correct specification the sample estimate $T^{-1}\Sigma\hat{\eta}_t x_{t\text{-}2}$ will be near zero anyway.

B. $\eta_t \sim$ MA(q). Then **nq**=q and **EXeta** and **EetaZtwid** (equation 2.2) are constructed as follows

equation (2.2) **EXeta** = $T^{-1}\Sigma(x_t+x_{t+1}+...+x_{t+q-1})\hat{\eta}_t$, or perhaps
EXeta = $(T\text{-}q)^{-1}\Sigma(x_t+x_{t+1}+...+x_{t+q-1})\hat{\eta}_t$,

the latter used if lack of data on leads of x at the end of the sample leads to there being only T-q observations in the sum.

•RATS: **EetaZtwid** is a $(q+1)\times 1$ vector, whose i'th element is a 1×3 estimate of $E\eta_t Z_{t+i-1}'$

EetaZtwid(i) = $T^{-1}\Sigma\hat{\eta}_t Z_{t+i-1}'$,

or perhaps **EetaZtwid**(i) = $(T\text{-}i)^{-1}\Sigma\hat{\eta}_t Z_{t+i-1}'$ if only T-i observations are available.

•As well, and as illustrated in example 3.1A, one might impose zeroes when a population covariance is known to be zero.
•Note that the first element of **EetaZtwid** is **EetaZtwid**(1)=$T^{-1}\Sigma\hat{\eta}_t Z_t'$, i.e., $T^{-1}\Sigma\hat{\eta}_t Z_{t+i}'$ for i=0 (not i=1), ..., the last (the $(q+1)^{st}$ element) is **EetaZtwid**(q+1)=$T^{-1}\Sigma\hat{\eta}_t Z_{t+q}'$.

•MATLAB: **EetaZtwid** is $(q+1)\times 3$. The i'th row is $T^{-1}\Sigma\hat{\eta}_t Z_{t+i-1}'$, or perhaps $(T\text{-}i)^{-1}\Sigma\hat{\eta}_t Z_{t+i-1}'$.

3.2 Long horizon regression, bivariate information set to compute the relevant moments.

$y_t =$ const. $+ \beta x_{t-1} + \eta_t$,

$y_t = \Delta s_{t+q} + \Delta s_{t+q-1} + \Delta s_t$, OR $y_t = \Delta s_{t+q}$,

bivariate VAR(2) in $(x_t, \Delta s_t)$ used to compute moments.

The companion form first order VAR is

$$Z_t \equiv \begin{pmatrix} x_t \\ \Delta s_t \\ x_{t-1} \\ \Delta s_{t-1} \end{pmatrix} = \text{const.} + \Phi Z_{t-1} + U_t = \text{const.} + \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} & \phi_{14} \\ \phi_{21} & \phi_{22} & \phi_{23} & \phi_{24} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ \Delta s_{t-1} \\ x_{t-2} \\ \Delta s_{t-2} \end{pmatrix} + \begin{pmatrix} u_{1t} \\ u_{2t} \\ 0 \\ 0 \end{pmatrix}$$

Let

$$U_{1t} \equiv \begin{pmatrix} u_{1t} \\ u_{2t} \end{pmatrix}, \text{ so that } U_t = \begin{pmatrix} U_{1t} \\ 0_{2\times 1} \end{pmatrix}; \text{ let } \Omega_1 \equiv EU_{1t}U_{1t}', \ \Omega_U = EU_tU_t' = \begin{pmatrix} \Omega_1 & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} \end{pmatrix}$$

Thus, U_{1t} is the 2×1 innovation in $(x_t\ \Delta s_t)$, with 2×2 VCV Ω_1. Then **nZtwid**=n_Z=4. Under the null that x_{t-1} is an efficient predictor of y_t, $\eta_t \sim$ MA (q) and **nq**=q. Other parameters passed include

$$\textbf{PX} = (1\ 0\ 0\ 0), \ \textbf{phitwid} = \begin{pmatrix} \hat\phi_{11} & \hat\phi_{12} & \hat\phi_{13} & \hat\phi_{14} \\ \hat\phi_{21} & \hat\phi_{22} & \hat\phi_{23} & \hat\phi_{24} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \ \textbf{omegaUtwid} = \begin{pmatrix} \hat\Omega_1 & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} \end{pmatrix}, \ \hat\Omega_1 = T^{-1}\Sigma \hat U_{1t}\hat U_{1t}'.$$

equation (2.2): **EXeta** and, with redefinition of Z_t, **EetaZtwid** are constructed as described in the previous example, **EXeta** = $T^{-1}\Sigma(x_t + x_{t+1} + ... + x_{t+q-1})\hat\eta_t$ and **EetaZtwid**(i) = $T^{-1}\Sigma\hat\eta_t Z_{t+i-1}'$ (RATS) or i'th row of **EetaZtwid** is $T^{-1}\Sigma\hat\eta_t Z_{t+i-1}'$ (MATLAB). In either RATS or MATLAB, T-q or T-i might replacing T.

•Of course $\hat\eta_t$, which is used to construct **EXeta** and **EetaZtwid**, will be different for $y_t = \Delta s_{t+q} + \Delta s_{t+q-1} + \Delta s_t$ than for $y_t = \Delta s_{t+q}$. So even though the same formulas apply, the resulting values will be different.

4. Discussion

4.1 **proc_vbias** is the only routine to do any computations. The other three procedures (**proc_vb_ma0**, **proc_vb_maq**, **proc_vb_d**) are front ends that set up parameters and then invoke **proc_vbias**.

4.2 **proc_vb_ma0** is a special case of **proc_vb_maq**. I put together the separate procedure (2.1) **proc_vb_ma0** solely to make a relatively simple procedure call available in the $\eta_t \sim$ i.i.d. case.

4.3. I originally intended to allow the code to handle panel data with fixed effects. Panel data is the reason for the parameter **nN** in (2.4) **proc_vbias**: while **nN** is set to 1 in this write-up, it is set to the number of cross-sectional units in a panel data application. This explains why the code has various loops running from 1 to **nN**. Code to handle **nN**>1 is not well debugged.

5. Population counterparts to variables passed to the procedures

Equation of interest: $y_t = \text{const.} + X_{t-1}'\beta + \eta_t$, $\eta_t \sim \text{MA}(q)$

(5.1) $\underset{n_Z\times 1}{Z_t - EZ_t} = \underset{n_Z\times n_Z}{\Phi}(Z_{t-1} - EZ_{t-1}) + \underset{n_Z\times 1}{U_t}$, **Ztwid**, **phitwid**

(5.2) $\underset{n_Z\times n_Z}{\Omega_U} = EU_tU_t'$, **omegaUtwid**

(5.3) $\underset{k\times 1}{X_t} = \underset{k\times n_Z}{P_X}\, \underset{n_Z\times 1}{Z_t}$, **PX**

(5.4a) $P_X = \begin{pmatrix} P_1' \\ P_2' \\ \cdots \\ P_k' \end{pmatrix}$,

(5.4b) $\underset{1\times n_Z}{P_i'}$ = i'th row of P_X, $X_{it} = P_i'Z_t$

(5.5) $E(\underset{1\times 1}{\eta_t} \mid \underset{n_Z\times 1}{U_t}, \underset{n_Z\times 1}{U_{t+1}}, \ldots, \underset{n_Z\times 1}{U_{t+q}}) = \underset{1\times n_Z}{\delta_0'}\underset{n_Z\times 1}{U_t} + \ldots + \underset{1\times n_Z}{\delta_q'}\underset{n_Z\times 1}{U_{t+q}}$. **delta**

(5.6) k=dimension of X_t, q = order of MA of η_t, n_Z = dimension of Z_t **nk**, **nq**, **nZtwid**

(5.7) $\underset{1\times n_Z}{E\eta_t Z_t'}$ **EetaZtwid0**

(5.8) $\underset{1\times n_Z}{E\eta_t Z_t'}, \underset{1\times n_Z}{E\eta_t Z_{t+1}'}, \ldots, \underset{1\times n_Z}{E\eta_t Z_{t+q}'}$ **EetaZtwid**

(5.9) $E\underset{k\times 1}{(X_t + X_{t+1} + \ldots + X_{t+q-1})}\eta_t$ **EXeta**

6. A schematic of the hierarchy in the code

Applications in which $X_{t-1} = (x_{t-1},\ldots,x_{t-k})'$–the only regressors are a set of lags of a single variable. User does not need to estimate the regression or compute moments. User calls **longhor1** or **longhor** to get $\hat{\beta}$, $\hat{b}$ and bias adjusted $\hat{\beta}$-$\hat{b}/T$. See the separate document "readme_longhor".

Applications involving regressors other than a set of lags of a single variable. User estimates $\hat{\beta}$ and computes certain moments. User calls **proc_vb_ma0** or **proc_vb_maq** to get $\hat{b}$. user then computes bias adjusted $\hat{\beta}$-$\hat{b}/T$. See the description above.

longhor and **longhor1** invoke **proc_vb_ma0** or **proc_vb_maq** to compute $\hat{b}$. They return $\hat{\beta}$, $\hat{b}$ and bias adjusted $\hat{\beta}$-$\hat{b}/T$.

proc_vb_ma0 and **proc_vb_maq**: take as input certain moments, invoke **proc_vbias**, return $\hat{b}$.

proc_vbias: code that computes $\hat{b}$. This code must be accessible when the user invokes any of the routines above.

7. Code illustrating the procedures

A. Code that uses these procedures to produce some of the entries in the tables in West (2016) is as follows:

- •Table 1 (except for GARCH entries): Table1.prg (RATS) and Table 1.m (MATLAB), the entries are computed using (2.1) **proc_vb_ma0**. They are also computed a second time using (2.3) **proc_vb_d**.
- •Table 2 (except for the b_1 and b_2 entries, which can be printed by inserting print statements in the **proc_vb_d** code): Table2A.prg, Table2B.prg, Table 2C.prg (RATS)and Table2A.m, Table2B.m, Table 2C.m (MATLAB), using (2.3) **proc_vb_d**.
- •RATS output is in .LIS files, MATLAB output in .PDF files.

There is a bit of extraneous code in some of these. Per the hierarchy of the code depicted above, **proc_vbias.src** (RATS) or **proc_vbias.m** (MATLAB) must be accessible from the program.

B. Separately, an empirical application: Let

s_t = 100×log exchange rate, US dollars per foreign currency unit
Δs_t = quarterly percentage change in bilateral U.S. dollar exchange rate
x_t = q+1 period interest differential, U.S. minus foreign, safe debt

Strictly speaking, x_t should be indexed by q but is not for simplicity.

The empirical example estimates the interest parity regression, regressing a q+1 period change in the exchange rate on the corresponding interest differential:

$$y_t \equiv s_{t+q}-s_{t-1} = \text{const.} + \beta x_{t-1} + \eta_t$$

for q=0 (one quarter), q=1 (6 month), q=3 (one year), q=19 (five year) and q=39 (10 years), for the U.S. dollar vs. the Canadian dollar, Japanese yen and U.K. pound, 1979-2011. Thus there are 15 sets of estimates: 5 horizons × 3 currency.

•Data are in **empirical_data.xlsx**. Kindly supplied by Menzie Chinn. A subset of the data used in Chinn and Quayyam (2012). The spreadsheet includes some data prior to 1979, but such data are not used in the example.

•The programs to produce the results in the table on the next page are **empirical_longhor.prg** (RATS) and **empirical_longhor.m** (MATLAB). These invoke a routine called longhor (described in “readme_longhor”). The longhor routine invokes (2.1) and (2.2) and thus illustrates use of these routines. These programs produce estimates for a given horizon and currency. Comments within the programs describe how to select horizon and currency.

•In addition to **empirical_data.xlsx**, the following files need to be accessible from the directory in which these programs are run:

RATS: **proc_vb_ma0.src**, **proc_vb_maq.src**, **proc_vbias.src**, **nwbandwidth.src**; a ***.inp** file as described in the comments in **empirical_longhor.prg**
MATLAB: **proc_vb_ma0.m, proc_vb_maq.m, proc_vbias.m**, **nwbandwidth.m**;

The next page has results.

Empirical Estimates

q+1	Canada			Japan			U.K.		
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
	$\hat{\beta}$	$\frac{\hat{b}}{T}$	$\hat{\beta}-\frac{\hat{b}}{T}$	$\hat{\beta}$	$\frac{\hat{b}}{T}$	$\hat{\beta}-\frac{\hat{b}}{T}$	$\hat{\beta}$	$\frac{\hat{b}}{T}$	$\hat{\beta}-\frac{\hat{b}}{T}$
1	-0.29	-0.02	-0.27	-2.50	-0.01	-2.49	-1.51	0.04	-1.55
	(0.59)			(0.74)			(0.97)		
2	-0.15	0.00	-0.15	-2.66	-0.00	-2.66	-1.31	0.09	-1.40
	(0.57)			(0.54)			(0.95)		
4	-0.06	0.00	-0.06	-2.33	0.07	-2.39	-0.87	0.17	-1.04
	(0.68)			(0.45)			(0.82)		
20	0.74	-0.14	0.89	0.48	0.27	0.21	0.70	0.31	0.39
	(0.45)			(0.58)			(0.51)		
40	1.50	-0.47	1.98	0.79	0.10	0.73	0.48	-0.05	0.53
	(0.61)			(0.41)			(0.12)		

Notes:

1. The least squares regression is y_t = const. + $\beta x_{t-1}+\eta_t$, where y_t is change in US dollar exchange rate from t-1 to $t+q$ and x_{t-1} is the interest rate differential on q+1 quarter government debt. Sample period is 1979:2-(2011:2-q) for Canada, 1979:2-(2011:3-q) for Japan and the U.K.. This implies T=129, 128, 126, 110 and 90 for q=0,1,3,19 and 39 for Canada, with T one observation greater for Japan and the U.K.. HAC standard errors in parentheses.

2. The variable b is the bias, $\lim_{T\to\infty} T(E\hat{\beta}-\beta) = b$. Computation of $\hat{b}$ relies in part on moments computed from a VAR(2) in x_t and the one quarter change in the exchange rate.

3. Columns (3), (6) and (9) presented bias adjusted estimates of $\hat{\beta}$, with $\hat{\beta}$ itself reported in columns (1), (4) and (7) and the bias $\hat{b}/T$ reported in columns (2), (5) and (8).