

Replication code for Lazarus, Lewis, Stock (2021), “The Size-Power Tradeoff in HAR Inference”

This replication folder provides the code necessary to compute numerical values found throughout the paper and to produce Table 1 and Figure 1 in main text and Figures S1-S3 in the Online Supplement, as well as additional figures appearing in previous working paper versions.

Numerical Results

`Table1FigureS1.m`: computes the maximum power losses of EWP relative to QS reported in Table 1 and the curves in Figure S1.

`Figure1.m`: computes the size-power frontiers and plots them, producing Figure 1.

`Figure_kernels.m`: computes the implied mean kernel for Fourier, cosine, and split sample basis functions in both time and frequency domains and plots them (figures included in WP Supplement).

`Figure_ARMAcabibration.m`: computes and plots the spectral density of an ARMA(2,1) process for given parameter values and confirms that those parameter values yield $\omega^{(2)}=4$. (figure included in WP supplement) The code then confirms these analytical results numerically.

`generateomegaderivative.m`: calculates $\omega^{(1)}$, $\omega^{(2)}$ values (normalized curvature of spectral density) used throughout the paper and this replication code.

`genconstants.m`: analytically computes the constants appearing in Theorem 5 and used throughout the paper and this replication code.

Monte Carlo Results

`RunMC.m`: master file, runs all Monte Carlo specifications, records Monte Carlo size and worst-case power loss for each estimator by calling `MCsizepower.m`.

`MCsizepower.m`: calculates size and worst-case power for various estimators for a specified DGP for all LRV estimators.

`Constructfigures.m`: produces all Monte Carlo figures (frontiers and scatters) found in the Supplement, in particular Figures S2-S3 (and many extra figures in WP Supplement).

Auxiliary Code

DGPs:

`ARmean.m`: simulates data and estimates mean for 1-dimensional location model with AR errors.

`loc2.m`: simulates data and estimates means for 2-dimensional location model with AR errors.

`ARMA21.m`: simulates data and estimates means for 1-dimensional location model with ARMA(2,1) errors.

`stochasAR.m`: simulates data and estimates coefficients for stochastic regressor model where both regressor(s) and errors follow AR processes.

Kernels and basis functions:

`wpbasis.m`: computes Fourier basis functions.

`cosbasis.m`: computes type-2 cosine basis functions.

`imbasis.m`: computes split-sample basis functions.

`QSfreq.m`: computes the QS kernel in the frequency domain.
`kvbkernel.m`: computes the KVB kernel in the time domain.
`nwkernel.m`: computes the NW kernel in the time domain.

Other:

`QScrits.m`: computes asymptotic fixed- b critical values for QS in the frequency domain, with results hard-code into `MCsizepower.m`.
`NWcrits.m`: computes asymptotic fixed- b critical values for Newey-West in the time domain, with results hard-code into `MCsizepower.m`.
`impliedkernel.m`: computes the implied mean kernel in both the time and frequency domains for a given basis function.
`densplot.m`: plots the spectral density of an ARMA(2,1) process, called by `Figure_ARMACalibration`.
`gramschmidt`: performs Gram-Schmidt orthonormalization on a set of basis functions.