Empirical Dynamic Quantile for Visualization of High-Dimensional Time Series

Daniel Peña — UC3M

Ruey S. Tsay — Chicago Booth

Ruben Zamar — UBC

January 16, 2019

This vignette describes the commands available in the program **EDQts** for computing empirical dynamic quantiles of a high-dimensional time series and for simulating high-dimensional series of vector autoregressive (VAR) models of order 1, vector moving-average (VMA) models of order 1, and vector generalized autoregressive conditional heteroscedastic (GARCH) models of order (1,1). In addition, it also produces a time plot showing the observed time series and some selected empirical quantile series.

There are four main commands in the **EDQts** program. They are **EDSts**, **HDgen**, **HDgarch** and **EDQplot**. We briefly describe these four functions below. The description provides input variables and output variables. Some simple illustrations are given in the next section. The program has another command **sqrtMtx**, which uses the eigen-decomposition to compute the square-root matrix of a positive-definite matrix. This command is used internally to generate high-dimensional time series.

1 Main functions

The five commands are given below with details.

1. **EDQts**: Computes an empirical dynamic quantile for a given dataset of high-dimensional time series. The probability and the number of weighted series used in the algorithm must be given.

- Form:

```
i <- EDQts(x,p=0.5,h=30)
```

- Input:
  - x: A $T \times m$ matrix containing m time series each with T data points. Each column contains an individual time series.
  - p: The probability between (0,1). The default is p = 0.5.
  - h: The number of series used in finding the p-th empirical dynamic quantile based on the search algorithm proposed in Peña, Tsay and Zamar (2019). The default is h = 30. The higher h is, the more accurate the result, but the longer to compute.
- Output: An integer identifying the column of the p-th empirical dynamic quantile series.

2. **HDgen**: Generates m Gaussian time series each of length T from either a VAR(1) or a VMA(1) model:

$$X_t = c + GX_{t-1} + E_t \quad \text{or} \quad X_t = c + E_t - GE_{t-1},$$

where $c = [c_i]$ with c_i being random draw from $u[-1, 1]$ and $E_t = \Sigma^{1/2} A_t$ with $\{A_t\}$ being *iid* m-dimensional random vector with mean zero and identity covariance matrix, and $\Sigma^{1/2}$ is the square-root matrix of $\Sigma = [\sigma_{ij}]$, where $\sigma_{ij} = \text{rho}^{|i-j|}$. The coefficient matrix is an upper-banded matrix. The width of the band matrix must be given with a default being 1. Specifically, the coefficient matrix $G = [g_{ij}]$ is given as (a) g_{ii}: random draw from $U[-0.7, 0.7]$ and g_{ij}: random draw from $U[-0.5, 0.5]$ for $j - i \leq$ band.

- Form:

```
m1 <- HDgen(m=100,T=100,rho=0.5,AR=TRUE,band=1)
```

- Input:
  - m: The dimension or number of time series. The default is 100 and m must be greater than 1.
  - T: The number of observations, i.e., sample size. The default is 100.
  - rho: The maximum correlation between the innovation series. The default is 0.5.

- AR: A switch controlling VAR(1) or VMA(1) series are generated. AR=TRUE for generating VAR(1) series. AR=FALSE for generating VMA(1) series.
- band: The width of the upper-banded coefficient matrix. The default is 1.

- Output:
  - et: The innovation series E_t. A $T \times m$ matrix.
  - Xt: The generated time series. Also a $T \times m$ matrix.
  - Sigma: The Σ matrix used in the simulation.
  - Coef: The coefficient matrix used in the simulation.

3. **HFgarch**: generates m-dimensional vector GARCH(1,1) time series. The program generates m independent univariate GARCH(1,1) model $x_t = \sigma_t a_t$ with a_t being *iid* $N(0,1)$ and $\sigma_t^2 = \alpha_0 + \alpha x_{t-1} + \beta\sigma_{t-1}^2$ with $\sigma_0 = \alpha_0/(1-\alpha-\beta)$. The coefficient α_0 is a random draw from $U[0.02, 0.15]$. For stationarity, let $\eta = \alpha + \beta$ and η is a random draw from $U[0.75, 0.98]$ and β is a random draw from $U[\eta/2, \eta]$ so that $\beta >= \alpha$ but $\alpha + \beta < 1$. The output series Y_t is defined as $Y_t = c + \Sigma^{1/2}X_t$, where $c = [c_i]$ with c_i being a random draw from $U[-1, 1]$, columns of X_t are univariate GARCH(1,1) series and $\Sigma = [\sigma_{ij}]$ is a banded matrix with $\sigma_{ij} = \text{rho}^{|i-j|}$ if $|i-j| <=$ band. Note that to mitigate the effect of initial values used in data generation, the program generate $T + 50$ observations, but discards the first 50 data points at the output.

- Form:

```
m2 <-  HDgarch(m=100,T=100,rho=0.5,band=1)
```

- Input:
  - m: The number of time series, i.e., the dimension. Default is 100.
  - T: The number of data points, i.e., the sample size. Default is 100.
  - rho: The maximum correlation coefficient between the series. Default is 0.5.
  - band: The width of banded matrix for creating cross correlation between the series. Default is 1.

- Output:
  - et: The m-dimensional Gaussian innovation series. A $(T + 50) \times m$ matrix.
  - Yt: The vector GARCH(1,1) series, which is a $T \times m$ matrix.
  - Xt: The independent individual GARCH(1,1) series, which is a $(T + 50) \times m$ matrix.

– par: The parameters used in generating the univariate GARCH(1,1) models. Each row contains $(\alpha_0, \alpha, \beta, \eta)$.

4. **EDQplot**: Obtain a time plot of the observed time series and some selected EDQ series. The EDQ series are shown in different colors.

   - Form:

     ```
     EDQplot(x,prob=c(0.05,0.5,0.95),h=30,loc=NULL,
     color=c(‘‘yellow’’,‘‘red’’,‘‘green’’,‘‘blue’’))
     ```

   - Input:
     – x: The data matrix, a $T \times m$ matrix of the time series data.
     – prob: The probability vector used to compute the EDQ of the data. The default is (0.05,0.50,0.95).
     – h: The number of time series used in the algorithm to compute EDQ. See **EDQts**.
     – loc: A vector containing the columns corresponding to EDQ. If loc=NULL, the program **EDQts** is used to find the EDQ series. If loc is given, then prob and h are not used.
     – color: The color vector used to plot the EDQ. If length of prob is greater than that of color, the color is recursively used.
   - Output: A time series plot. The data are in black line and EDQ in color.

5. **sqrtMtx**: Compute the square-root matrix of a positive definite matrix.

   - Form:

     ```
     s <- sqrtMtx(Sigma)
     ```

   - Input: Sigma is a positive definite square matrix
   - Output: the square-root matrix of Sigma.

2 Demonstration

We demonstrate the use of the commands with some simulated examples.

Example 1: In this example, we generated 1000 time series of length 100 from a VAR(1) model and use the data to obtain EDQ with probabilities $p = 0.05$, 0.5, and 0.95. That is, we have $m = 1000$, $T = 100$, and prob = c(0.05,0.50,0.95). The resulting plot of **EDQplot** is shown in Figure 1.

```
> source("EDQts.R")
> set.seed(11)
> m1 <- HDgen(m=1000) ## Generating VAR(1) process: 1000 series and 100 observations
> Xt <- m1$Xt
> i50 <- EDQts(Xt)  ## using default options
> i50
[1] 703
> i05 <- EDQts(Xt,p=0.05)
> i05
[1] 192
> i95 <- EDQts(Xt,p=0.95)
> i95
[1] 438
> EDQplot(Xt,loc=c(192,703,438))
> EDQplot(Xt) ## not run, but it produces exactly the same plot.
```

Example 2. In this example, we generate a vector GARCH(1,1) process with dimension $m = 1000$ and sample size $T = 100$. Then, we compute some selected EDQ series and obtain a time plot. Since volatility process has higher excess kurtosis, one might want to use a larger h to compute EDQ for extreme probabilities. The time plot of the data and three selected EDQ are shown in Figure 2. We also obtain a time plot of the data with four selected EDQ series in Figure 3, where the EDQ series are (a) with probability 0.01 (yellow), (b) with probability 0.25 (red), (c) with probability 0.75 (green), and (d) with probability 0.99 (blue).

```
> source(‘‘EDQts.R’’)
> set.seed(15)
> m2 <- HDgarch(m=1000)
> names(m2)
```

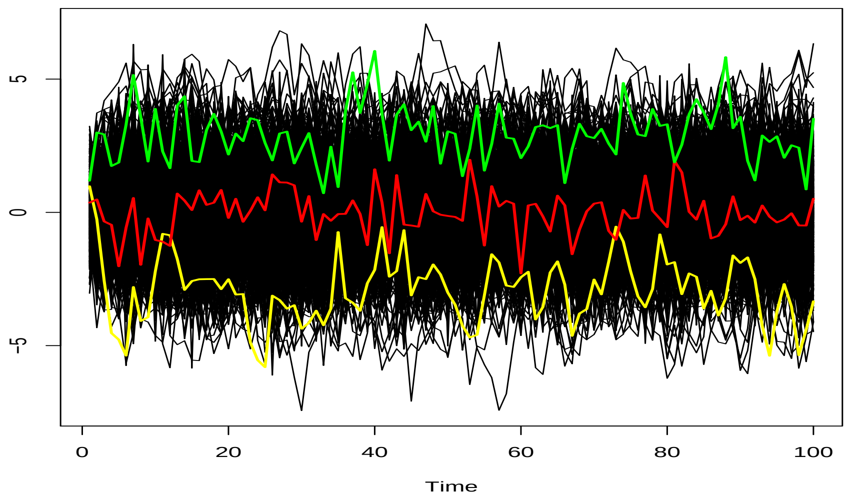

Figure 1: Time plot of the VAR(1) process with 1000 series and 100 observations. Also shown are three EDQ series with probability 0.05 (yellow), 0.5(red) and 0.95(green), respectively.

```
[1] "et"  "Yt"  "Xt"  "par"
> Yt <- m2$Yt
> dim(Yt)
[1]  100 1000
> j50 <- EDQts(Yt)
> j50
[1] 854
> j05 <- EDQts(Yt,p=0.05,h=50)
> j05
[1] 551
> j95 <- EDQts(Yt,p=0.95,h=50)
> j95
[1] 596
> EDQplot(Yt,loc=c(551,854,596))
> EDQplot(Yt,prob=c(0.05,0.50,0.95) # Not run, it produces the same plot.
> EDQplot(Yt,prob=c(0.01,0.25,0.75,0.99),h=50)
> q()
```

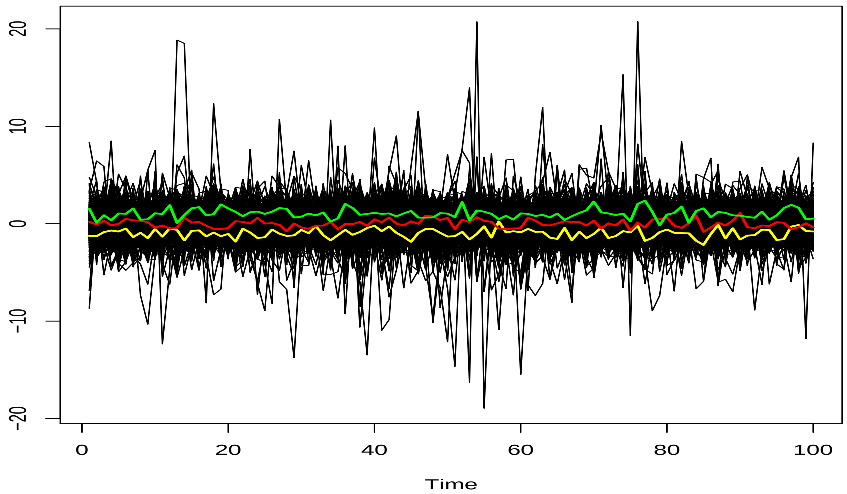

Figure 2: Time plot of the VGARCH(1,1) process with 1000 series and 100 observations. Also shown are three EDQ series with probability 0.05 (yellow), 0.5(red) and 0.95(green), respectively.

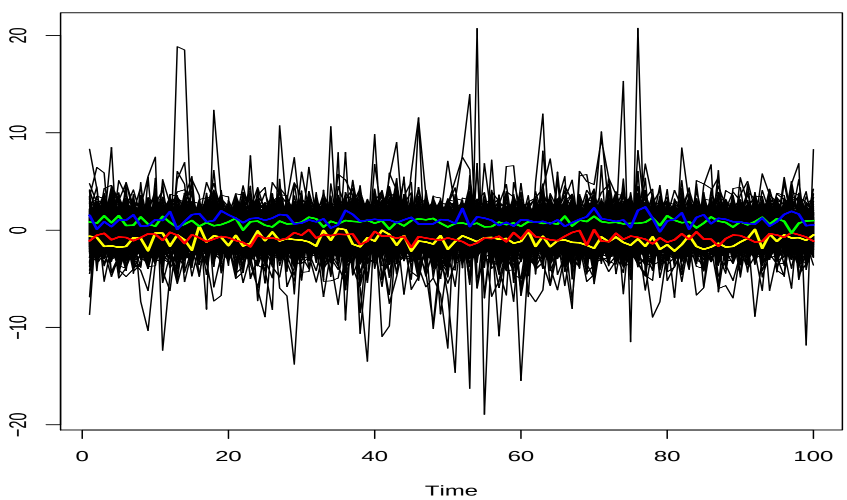

Figure 3: Time plot of the VGARCH(1,1) process with 1000 series and 100 observations. Also shown are three EDQ series with probability 0.01 (yellow), 0.25(red), 0.75(green) and 0.99 (blue), respectively.