BFAST

Notation and definitions: For an $m \times n$ matrix A, an n-vector x, $I \subset \{1, \ldots, m\}$, $J \subset \{1, \ldots, n\}$, let A_{IJ} denote the submatrix of A formed from the rows indexed by I and the columns indexed by J, and x_J denote the subvector of x indexed by J. $A_{I\cdot}$ ($A_{\cdot J}$) are the rows (columns) of A indexed by I (J), respectively. An image is an $R \times C$ matrix D, where each D_{rc} (pixel) is an $S \times B$ matrix, whose (s, b) element $(D_{rc})_{sb}$ is the signal value at time index s and frequency band index b.

Algorithm BFAST.

for band $b = 1 : B$

for row $r = 1 : R$

for col $c = 1 : C$ **do**

Let $T = (t_1, \ldots, t_S)$ be the sequence of given time points and the S-vector u denote the time series data in the column $(D_{rc})_{\cdot b}$, i.e., $u = (D_{rc})_{\cdot b}$. Assume that the general model is of the form

$$u = \mathcal{V} + \mathcal{W} + \epsilon,$$

where $\mathcal{V}$ and $\mathcal{W}$ denote the iteratively computed trend and seasonal components, respectively, present in the data and ϵ is the noise. The trend $\mathcal{V}$ may be piecewise linear and the seasonal component $\mathcal{W}$ may be piecewise harmonic. Let N be the maximum number of iterations, n be the iteration number, and $\mathcal{V}^n$ and $\mathcal{W}^n$ be the trend and seasonal components, respectively, computed at the nth iteration. Let $h \in (0, 1)$ denote the proportion of data points by which two consecutive breakpoints t_i and t_j (including t_1 and t_S) must be separated. Thus $\lceil Sh \rceil \leq j - i - 1$. Take the length of moving windows to be $\lceil Sh \rceil$, initialize the iteration number $n := 1$, and initialize the seasonal component as $\mathcal{W}^0(T) = (w_1^0, \ldots, w_S^0)$.

Step 1.1: Determine the possibility of breakpoints in trend.

Eliminate the seasonal component from the data

$$u^n = u - \mathcal{W}^{n-1}(T).$$

The ordinary least squares (OLS) estimator for the trend is given as

$$\alpha = (X^t X)^{-1} X^t u^n$$

where X is the Gram matrix for linear regression given by

$$X = \begin{pmatrix} 1 & t_1 \\ \vdots & \vdots \\ 1 & t_S \end{pmatrix}.$$

The prediction error (or residual vector or the OLS residual) is defined as

$$E^o = u^n - X\alpha,$$

where the superscript 'o' is used to signify the fact that these residuals are OLS regression based. Consider the process defined by the moving sums (MOSUM) of these OLS residuals

$$Q^o = \left\{ \frac{1}{\sigma\sqrt{\lceil Sh \rceil}} \sum_{i=k-\lceil Sh \rceil+1}^{k} E_i^o \right\}_{k=\lceil Sh \rceil}^{S},$$

where σ is the sample standard deviation of all the OLS residuals.

Compute the OLS-MOSUM test statistic

$$\hat{f}^o = \max_{1 \leq k \leq S-\lceil Sh \rceil+1} |Q_k^o|$$

as the maximum absolute value of this process, then compute the asymptotic critical value of the OLS-MOSUM test using the two-sided boundary-crossing probability

$$p_T = P[f^o > \hat{f}^o],$$

where p_T is read from the Brownian Bridge table.

A p-value less than a user defined parameter $\tau_{\mathcal{V}} \in (0,1)$ indicates the presence of breakpoints.

Remark 1: As discussed in cite, under the null hypothesis, the OLS-MOSUM process converges in distribution to the increments of a Brownian Bridge process.

Step 1.2: Locate trend breakpoints.

Suppose $p_T \leq \tau_{\mathcal{V}}$. To locate the breakpoints, consider all possible partitions of the domain, compute OLS fits for each partition, and settle with a partition that yields minimum squared error.

Let $X_{[i,j]}$ denote the matrix formed from rows i through j of the matrix X, and $\alpha_{[i,j]}$ denote the least squares coefficients computed using the matrix $X_{[i,j]}$ with time points $t_i, \ldots, t_j$, and data $u^n_{[i,j]} = u^n_{\{i,\ldots,j\}}$. For $i = 1, \ldots, S - \lceil Sh \rceil + 1$, consider each window $[t_i, \ldots, t_{j-1}]$, $i+2 \leq j \leq S$, and the linear fit in this window. The recursive residual at point t_j is then defined as the weighted prediction error

$$E_{ij}^r = \frac{u^n_{[j,j]} - X_{[j,j]}\alpha_{[i,j-1]}}{\sqrt{1 + X_{[j,j]}\left(X^t_{[i,j-1]}X_{[i,j-1]}\right)^{-1}X^t_{[j,j]}}}.$$

The superscript 'r' is used to signify the fact that the process/statistic is recursive residual based.

Suppose a breakpoint has been found at t_i. Then the cost of placing the next breakpoint at t_k is calculated as the accumulated sum of squared recursive residuals in the interval $[t_i, t_{k-1}]$, i.e.,

$$\rho_{ik} = \sum_{j=i+2}^{k-1} \left(E_{ij}^r\right)^2.$$

All possible positions for the breakpoints can thus be calculated by considering the moving sums of squared recursive residuals, i.e., the process defined by

$$Q^r = \left\{ \left\{ \sum_{j=i+2}^{k} \left(E_{ij}^r\right)^2 \right\}_{k=i+2}^{S} \right\}_{i=1}^{S-\lceil Sh \rceil+1}.$$

Given the number μ of desired interior breakpoints, let $k_1, \ldots, k_\mu$ be integers such that $k_{i+1} - k_i > \lceil Sh \rceil$, $k_1 > \lceil Sh \rceil + 1$, and $k_\mu < S - \lceil Sh \rceil$. Determine $K = (1, k_1, \ldots, k_\mu, S)$ to minimize the moving sums of squared recursive residuals

$$\sum_{i=3}^{k_1-1} \left(E_{1,i}^r\right)^2 + \sum_{i=k_1+2}^{k_2-1} \left(E_{k_1,i}^r\right)^2 + \sum_{i=k_2+2}^{k_3-1} \left(E_{k_2,i}^r\right)^2 + \cdots + \sum_{i=k_\mu+2}^{S} \left(E_{k_\mu,i}^r\right)^2.$$

Then $(t_{k_1}, \ldots, t_{k_\mu})$ are the interior breakpoints in the trend component.

Remark 2: The breakpoints $t_1, t_{k_1}, \ldots, t_{k_\mu}, t_S$ are optimal in the sense of the above moving sums of squared recursive residuals criterion.

Remark 3: If $p_T > \tau_{\mathcal{V}}$, then there are only two breakpoints (t_1 and t_S) and no interior breakpoints. So this step is skipped and there is simply one linear fit over the entire domain $[t_1, t_S]$ (Step 1.3).

Step 1.3: Let $k_0 = 1$, $k_{\mu+1} = S$, and $I_0 = [t_{k_0}, t_{k_1})$, $I_1 = [t_{k_1}, t_{k_2})$, $\ldots$, $I_\mu = [t_{k_\mu}, t_{k_{\mu+1}}]$. For each interval I_i, determine the linear regression coefficients

$$\gamma^i = \left(X_{[k_i,k_{i+1}]}^t X_{[k_i,k_{i+1}]}\right)^{-1} X_{[k_i,k_{i+1}]} u_{[k_i,k_{i+1}]}^n$$

and construct the (discontinuous) piecewise linear fit

$$\mathcal{V}^n(t) = \sum_{i=0}^{\mu} \Gamma^i(t), \text{ where } \Gamma^i(t) = \begin{cases} \gamma_0^i + \gamma_1^i t, & t \in I_i, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathcal{V}^n(T) = (v_1^n, \ldots, v_S^n)$ be the sequence of values estimated at $t_1, \ldots, t_S$ using this piecewise linear fit.

Step 2.1: Determine the possibility of breakpoints in seasons.
Eliminate the estimated trend component from the observed data

$$\tilde{u}^n = u - \mathcal{V}^n(T).$$

The Gram matrix for the seasonal (harmonic) component is given by

$$Y = \begin{pmatrix} 1 & \sin t_1 & \cos t_1 & \cdots & \sin Kt_1 & \cos Kt_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \sin t_S & \cos t_S & \cdots & \sin Kt_S & \cos Kt_S \end{pmatrix},$$

where K is the degree of the trigonometric polynomial used for regression. The trigonometric regression coefficients for the seasonal component are computed as

$$\beta = (Y^t Y)^{-1} Y^t \tilde{u}^n.$$

The prediction error for this fit is defined as

$$E^o = \tilde{u}^n - Y\beta.$$

The OLS-MOSUM process for these errors is given by

$$Q^o = \left\{ \frac{1}{\sigma\sqrt{\lceil Sh \rceil}} \sum_{i=k-\lceil Sh \rceil+1}^{k} E_i^o \right\}_{k=\lceil Sh \rceil}^{S},$$

and the OLS-MOSUM test statistic is

$$\hat{g}^o = \max_{1 \le j \le S-\lceil Sh \rceil+1} |Q_j^o|.$$

The two-sided boundary-crossing probability

$$p_S = P[g^o > \hat{g}^o]$$

is read from the Brownian Bridge table.

A p-value less than a user defined parameter $\tau_{\mathcal{W}} \in (0,1)$ indicates the presence of seasonal breakpoints.

Step 2.2: Locate seasonal breakpoints.

Suppose $p_S \le \tau_{\mathcal{W}}$. Using the same notation as for the trend breakpoints,

$$E_{ij}^r = \frac{\tilde{u}_{[j,j]}^n - Y_{[j,j]}\beta_{[i,j-1]}}{\sqrt{1 + Y_{[j,j]}\left(Y_{[i,j-1]}^t Y_{[i,j-1]}\right)^{-1} Y_{[j,j]}^t}}$$

is the recursive residual at time t_j, obtained by trigonometric regression in the time window $[t_i, t_{j-1}]$.

Given the number ν of desired seasonal interior breakpoints and a minimum number of data points separating breakpoints (as for the trend), let $l_1, \ldots, l_\nu$ be integers such that $l_{i+1} - l_i > \lceil Sh \rceil$, $l_1 > \lceil Sh \rceil + 1$, and $l_\nu < S - \lceil Sh \rceil$. Determine $L = (1, l_1, \ldots, l_\nu, S)$ to minimize the moving sums of squared recursive residuals

$$\sum_{i=3}^{l_1-1} \left(E_{1,i}^r\right)^2 + \sum_{i=l_1+2}^{l_2-1} \left(E_{l_1,i}^r\right)^2 + \sum_{i=l_2+2}^{l_3-1} \left(E_{l_2,i}^r\right)^2 + \cdots + \sum_{i=l_\nu+2}^{S} \left(E_{l_\nu,i}^r\right)^2.$$

Then $(t_{l_1}, \ldots, t_{l_\nu})$ are the interior breakpoints in the seasonal component.

Remark 4: If $p_S > \tau_{\mathcal{W}}$, then there are only two breakpoints (t_1 and t_S) and no interior breakpoints. So this step is skipped and there is simply one trigonometric polynomial fit over the entire domain $[t_1, t_S]$ (Step 2.3).

Step 2.3: Let $l_0 = 1$, $l_{\nu+1} = S$, and $J_0 = [t_{l_0}, t_{l_1})$, $J_1 = [t_{l_1}, t_{l_2})$, $\ldots$, $J_\nu = [t_{l_\nu}, t_{l_{\nu+1}}]$. For each interval J_j determine the trigonometric polynomial regression coefficients

$$\delta^j = \left(Y^t_{[l_j, l_{j+1}]} Y_{[l_j, l_{j+1}]}\right)^{-1} Y_{[l_j, l_{j+1}]} \tilde{u}^n_{[l_j, l_{j+1}]}$$

and construct the (discontinuous) piecewise trigonometric polynomial

$$\mathcal{W}^n(t) = \sum_{j=0}^{\nu} \Delta^j(t), \text{ where } \Delta^j(t) = \begin{cases} \delta^j_1 + \sum_{k=1}^{K} \delta^j_{2k} \sin kt + \delta^j_{2k+1} \cos kt, & t \in J_j, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathcal{W}^n(T) = (w^n_1, \ldots, w^n_S)$ be the sequence of values estimated at $t_1, \ldots, t_S$ using this piecewise trigonometric polynomial approximation.

Step 3: Compare the breakpoints between iterations $n - 1$ and n.

If the Hamming distance between the two breakpoint vectors $(t_{k_1}, \ldots, t_{k_\mu}, t_{l_1}, \ldots, t_{l_\nu})$ at iterations $n - 1$ and n is less than some defined tolerance or the number of iterations has reached N, then exit. Otherwise, increment the iteration number n and repeat Steps 1.1 to 3.

end

end

end