Wavelet Methods for Time Series Analysis

Part VI: Matching Pursuit

- idea: approximate \( X \) using a few \# of ‘time/frequency’ vectors from large set of such vectors (cf. best basis)
- form ‘dictionary’ of vectors \( D \equiv \{ d_\gamma : \gamma \in \Gamma \} \)
  \( d_\gamma = [d_{\gamma,0}, d_{\gamma,1}, \ldots, d_{\gamma,N-1}]^T \), where \( \|d_\gamma\|^2 = 1 \)
  - \( \gamma \) is vector of parameters connecting \( d_\gamma \) to time/frequency; e.g., \( \gamma = [j, n, \ell]^T \) for WP table dictionary
  - \( \Gamma \) = finite set of possible values for \( \gamma \)
  - \( D \) contains basis for \( R^N \), but can be highly redundant (helps identify time/frequency content in \( X \))
- matching pursuit successively approximates \( X \) with orthogonal projections onto elements of \( D \)

Matching Pursuit Algorithm: I

- for \( d_{\gamma_0} \in D \), form \( \langle X, d_{\gamma_0} \rangle d_{\gamma_0} \), and define residual vector:
  \( R^{(1)} \equiv X - \langle X, d_{\gamma_0} \rangle d_{\gamma_0} \) so that \( X = \langle X, d_{\gamma_0} \rangle d_{\gamma_0} + R^{(1)} \)
- note that \( d_{\gamma_0} \) and \( R^{(1)} \) orthogonal:
  \( \langle d_{\gamma_0}, R^{(1)} \rangle = \langle d_{\gamma_0}, X - \langle X, d_{\gamma_0} \rangle d_{\gamma_0} \rangle 
  = \langle d_{\gamma_0}, X \rangle - \langle d_{\gamma_0}, \langle X, d_{\gamma_0} \rangle d_{\gamma_0} \rangle 
  = \langle d_{\gamma_0}, X \rangle - \langle X, d_{\gamma_0} \rangle = 0 \)
- hence \( \langle X, d_{\gamma_0} \rangle d_{\gamma_0} \& R^{(1)} \) are also orthogonal, showing that
  \( \|X\|^2 = \|\langle X, d_{\gamma_0} \rangle d_{\gamma_0}\|^2 + \|R^{(1)}\|^2 = \|\langle X, d_{\gamma_0} \rangle\|^2 + \|R^{(1)}\|^2 \)
- minimize energy in residuals by choosing \( \gamma_0 \in \Gamma \) such that
  \( \|\langle X, d_{\gamma_0} \rangle\| = \max_{\gamma \in \Gamma} \|\langle X, d_{\gamma} \rangle\| \)

Matching Pursuit Algorithm: II

- after first step of algorithm, second step is to treat the residuals in the same manner as \( X \) was treated in first step, yielding
  \( R^{(1)} = \langle R^{(1)}, d_{\gamma_1} \rangle d_{\gamma_1} + R^{(2)} \),
  with \( d_{\gamma_1} \) picked such that
  \( \|R^{(1)}, d_{\gamma_1}\| = \max_{\gamma \in \Gamma} \|R^{(1)}, d_{\gamma}\| \)
- letting \( R^{(0)} \equiv X \), after \( m \) such steps, have additive decomposition:
  \( X = \sum_{k=0}^{m-1} \langle R^{(k)}, d_{\gamma_k} \rangle d_{\gamma_k} + R^{(m)} \)
Matching Pursuit Algorithm: III

- also have an energy decomposition:

\[
\|X\|^2 = \sum_{k=0}^{m-1} \| \langle R^{(k)}, d_{\gamma_k} \rangle d_{\gamma_k} \|^2 + \|R^{(m)}\|^2 \\
= \sum_{k=0}^{m-1} |\langle R^{(k)}, d_{\gamma_k} \rangle|^2 + \|R^{(m)}\|^2
\]

- note: as \( m \) increases, \( \|R^{(m)}\|^2 \) must decrease (must reach zero under certain conditions)

Matching Pursuit Dictionaries: I

- key to matching pursuit is dictionary
- simplest dictionary: DWT dictionary
  - \( D \) contains \( d_{\gamma} \equiv W_{j*} \), \( j = 0, \ldots, N - 1 \)
  - \( \gamma = [j] \) associates \( W_{j*} \) with time/scale
  - \( \langle X, d_{\gamma} \rangle = W_j \) is \( j \)th DWT coefficient
  - 1st step picks \( W_j \) with largest magnitude:
    \[
    X = W_{(0)} W_{(0)} + R^{(1)} \quad \text{with} \quad R^{(1)} = \sum_{j \neq (0)} W_j W_{j*} 
    \]
  - 2nd step picks out \( W_j \) with 2nd largest \( |W_j| \)
  - for any orthonormal \( D \), matching pursuit approximates \( X \) using coefficients with largest magnitudes

Matching Pursuit Dictionaries: II

- larger dictionary: wavelet packet table dictionary (more flexible than best basis)
- even larger dictionary: above combined with basis vectors corresponding to a discrete Fourier transform (DFT)
- level \( J_0 \) MODWT dictionary
  - works for all \( N \), shift invariant, redundant
  - \( D \) contains vectors whose elements are either
    * normalized rows of \( \tilde{W}_j \), \( j = 1, \ldots, J_0 \), or
    * normalized rows of \( \tilde{V}_{J_0} \)

Example – Subtidal Sea Levels: I

- recall subtidal sea level series \( X \) for Crescent City, CA
Example – Subtidal Sea Levels: II

- use $J_0 = 10$ LA(8) MODWT dictionary (96,206 vectors in all)
- above shows first 10 vectors picked by matching pursuit ($\times \pm 1$)

Example – Subtidal Sea Levels: III

- next 10 vectors picked by matching pursuit ($\times \pm 1$)

Example – Subtidal Sea Levels: IV

- very first ($k = 0$) associated with overall increase in 1982–3
- first 10 are for $\tau_8 \Delta t = 64$ to $\lambda_{10} \Delta t = 512$ days
- 7 of first 20 are associated with $\tau_9 \Delta t = 128$ days (needed to account for seasonal variability)
- $k = 3$ has inverted sign & picks out gradual dip in Spring, 1984 (cf. 1981, 3, 5, 7 & 8); $k = 8$ also inverted, but is a boundary effect

Example – Subtidal Sea Levels: V

- matching pursuit approximations of orders $m = 20$, 50 and 200, along with residuals for $m = 200$
Example – Subtidal Sea Levels: VI

- matching pursuit approximations of orders $m = 20$, 50 and 200, but now using a dictionary augmented to include basis vectors corresponding to the DFT
- $k = 0$ choice same as before, but $k = 1$ choice is DFT vector with period close to one year
- for $2 \leq k < 200$, only $k = 65, 84$ and 192 are DFT vectors

Example – Subtidal Sea Levels: VII

- matching pursuit approximations of orders $m = 20$, 50 and 200, but now using a dictionary consisting of just the basis vectors corresponding to the DFT

Example – Subtidal Sea Levels: VIII

- normalized residual sum of squares $\| R^{(m)} \|^2 / \| X \|^2$ versus number of terms $m$ in matching pursuit approximation using the MODWT dictionary (thick curve), the DFT-based dictionary (dashed) and both dictionaries combined (thin)
- combined dictionary does best for small $m$, but MODWT dictionary by itself becomes competitive as $m$ increases