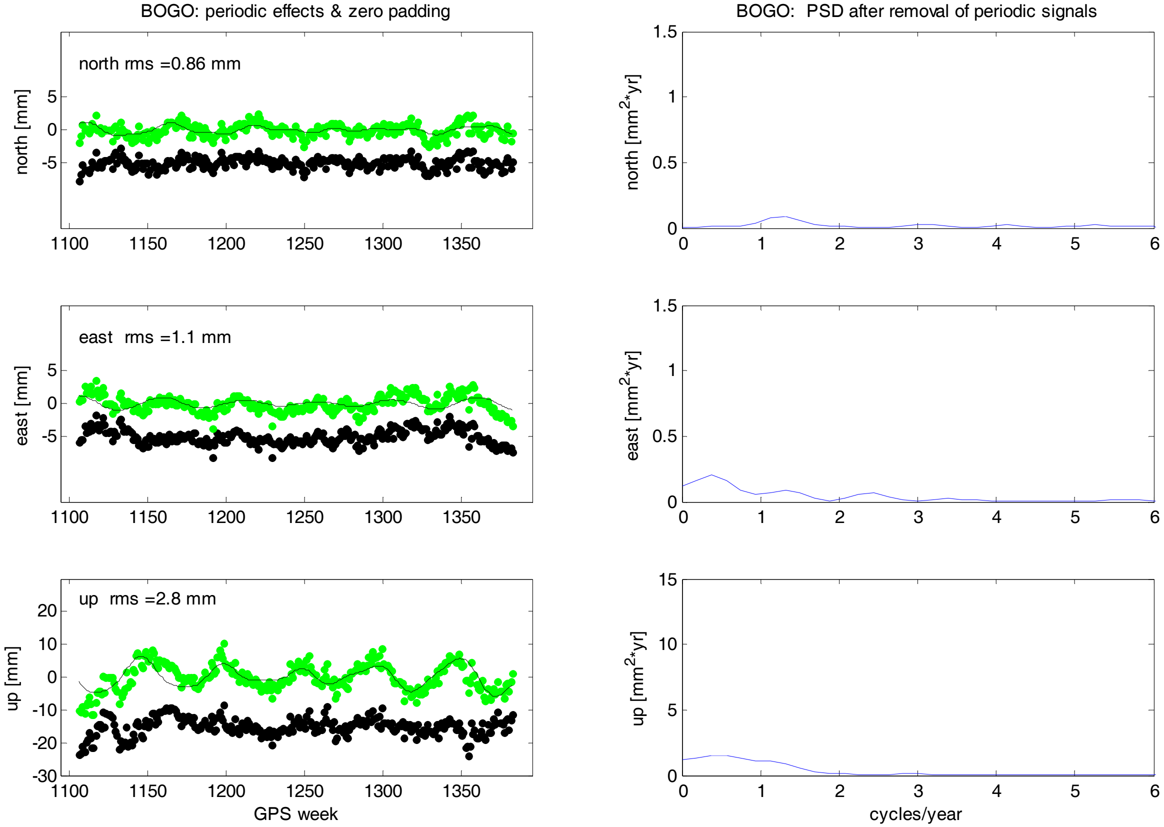
Predicting and filtering time series

Alessandro Caporali
University of Padova

Outlook

- Time series, after removal of empiric periodic terms (annual, typically) follow in most case a power law spectrum, with flicker phase (1/f) behavior at low (<2 cycles/yr) frequencies and white noise at higher frequencies
- This property has been recognized, but never exploited
- Here I report on attempts to
  - Filter the series, in order to investigate the existence of stochastic signals buried in the noise, which may be relevant at improving the TRF and/or detecting a geophysically interesting signal
  - Predict the series, to help analysts to understand when a jump occurs, in a most rigorous sense (i.e. not by ‘bare eye’)

Example: BOGO



Left

- Green dots: raw series
- Superimposed black line: best fitting sinusoid
- Black dots: green minus sinusoid

Right

- Resulting spectrum (PSD)

Noise description in detail (1/2)

- Wiener approach to filtering Time Series:
  - Is based on weighting function
  - Examples:
    - Gauss - Markov process
    - ARMA or digital filter (e.g. FILTER function in MATLAB
    - Special case: least squares collocation
- Wiener Theorem on predicting a Gauss Markov process:
  - The value of a random variable predicted Δt ahead equals the last observed value times the autocorrelation function evaluated at time Δt
  - Corollary: for large Δt, the predicted value of the r.v. tends to zero, which is the mean (and most probable) value of the r.v.

$$y(t) = \sum_{t'} a_{tt'} x(t')$$

$$R(\tau) = R(0) e^{-\frac{\tau}{T}}, \text{ with } \tau = \text{t - t'}$$

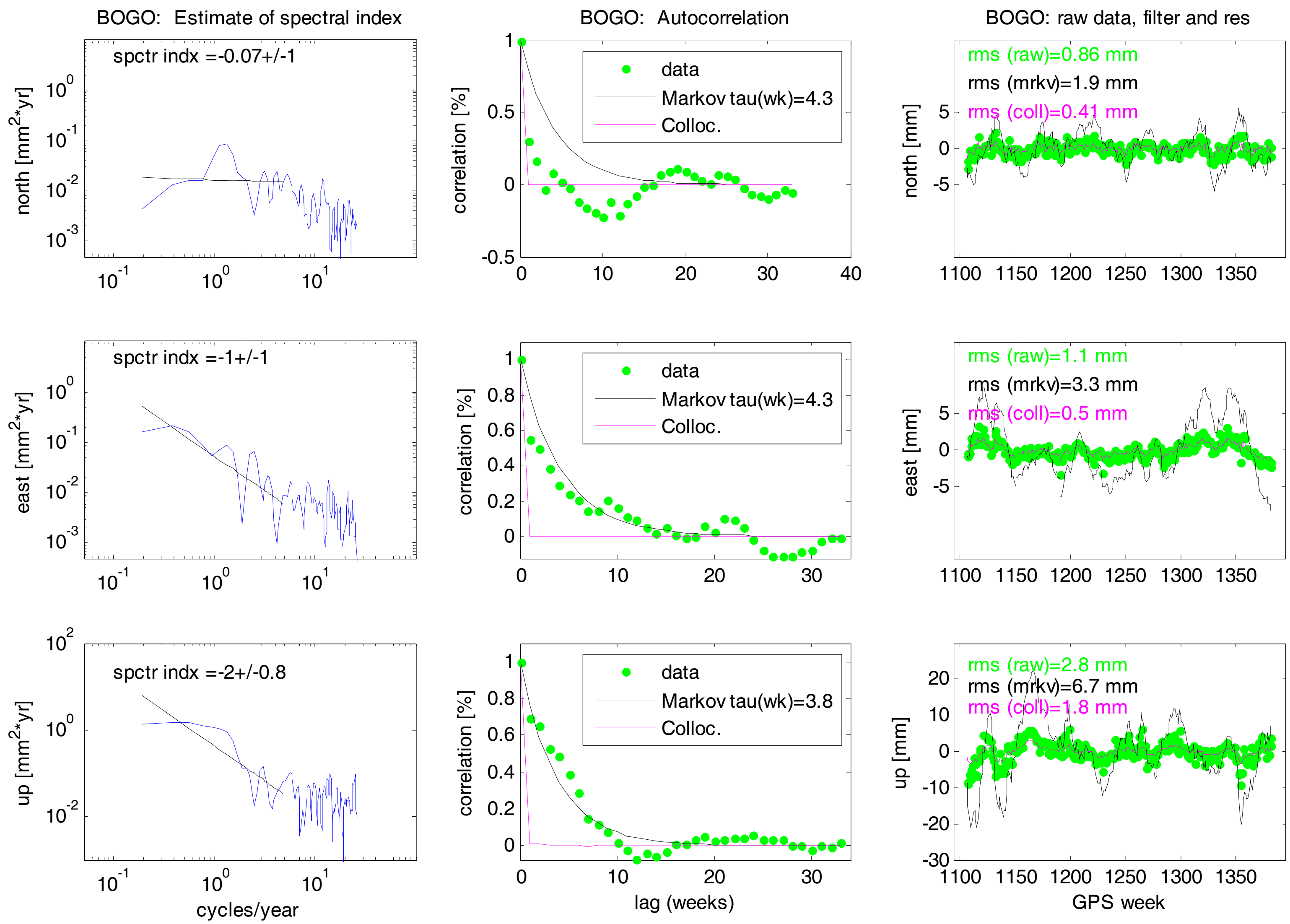
$$y(t) = FILTER(A, B, x)$$

$$a_{tt'} = \sum_{t''} \langle x(t) x(t'') \rangle \langle x(t') x(t'') + \sigma^2 \delta_{t't''} \rangle^{-1}$$

smoother

$$y_{-}(t + \Delta t) = R(0) e^{-\frac{\Delta t}{T}} x(t)$$

Noise description in detail (2/2)



Results from Wiener approach

- We have tested two options:
  - Gauss-Markov, with characteristic time T estimated from the autocorrelation function
  - Least squares collocation, with a smoother term of amplitude equal to the rms^2 of the raw data
- Results:
  - GM tends to amplify the input signal
  - LSQ is more effective in filtering the data, and is equivalent to force the data to be a GM process with small T
- Inference:
  - Time series may be considered Gauss Markov processes, because the autocorrelation looks like an exponential. However filtering the data with a Wiener filter built on a GM process leads to noise amplification. Hence
    - a) Wiener filter is unable to filter the noise

    Or
    - b) the time series is only apparently Gauss Markov
- Action in response to a) : try a Kalman filter in place of a Wiener Filter

Kalman approach

- Based on recursive relation rather than weighting function
- 4 independent variables, 2 initial values:
  - Covariance of the measurement noise R=cov(v,v) with y=ϕy_+ v
  - Covariance of process noise Q=cov(w,w) with z=Hy+w
  - Partial derivatives matrix H
  - Characteristic time of Gauss Markov process

  And
  - X_(0); P_(0) predicted initial value for the random variable and its Probability

$T = 4$ weeks (typical)

$R = \sigma_p^2$ variance of process noise (tbd : see later)

$Q = \left(1 - e^{-\frac{2}{T}}\right)\sigma_m^2$ where σ_m^2 is the variance of the measurement noise (tbd : see later)

$H = 1$

$\phi = e^{-\frac{1}{T}}$

$y_- = 0$

$P_- = 0$

Filter loop :

$K(i) = P\_(i)H\left[HP\_(i)H + R\right]$ Kalman gain

filtered variable at $t = i$:

$y(i) = y\_(i) + K(i)\left[x(i) - Hy\_(i)\right]$

$P(i) = \left[1 - K(i)H\right]P\_(i)$

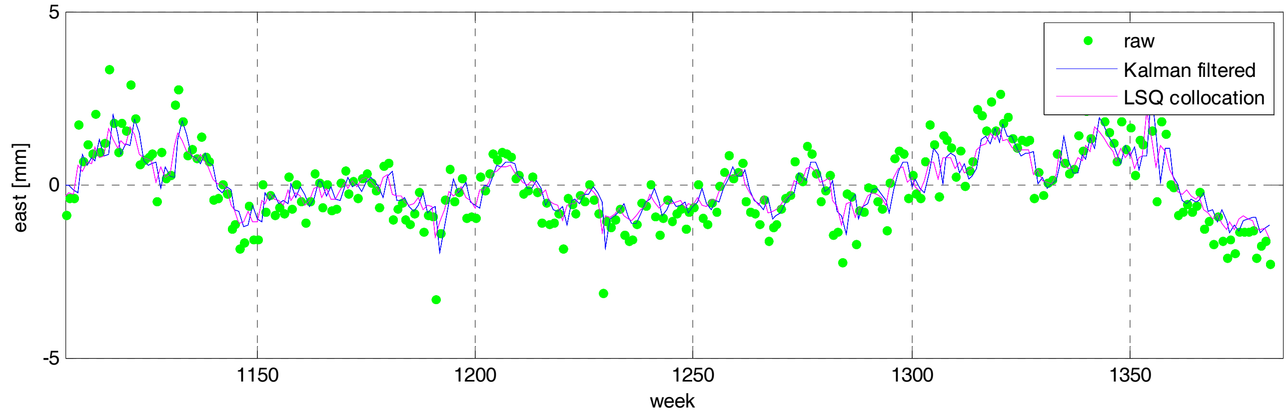
predicted variable at $t = i + 1$:

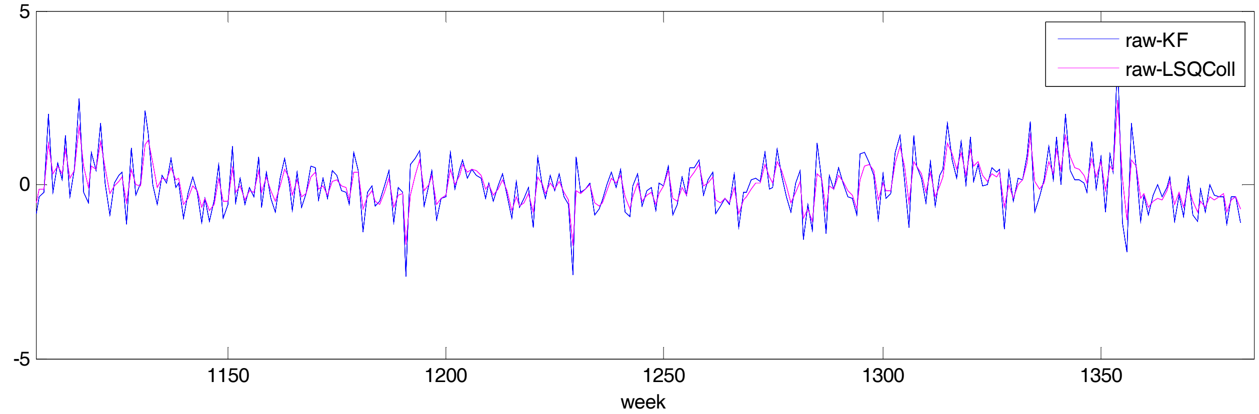
$y\_(i+1) = \phi y(i)$, as in Wiener filter

$P\_(i+1) = \phi P(i)\phi + Q$

Example: filter the east of BOGO

- Plot raw data (green) and filtered data
- Kalman Filter and LSQ collocation appear to be equally capable to filter the noise
- Rms=1.17 mm (raw)
- Rms=0.52 mm (filtered)



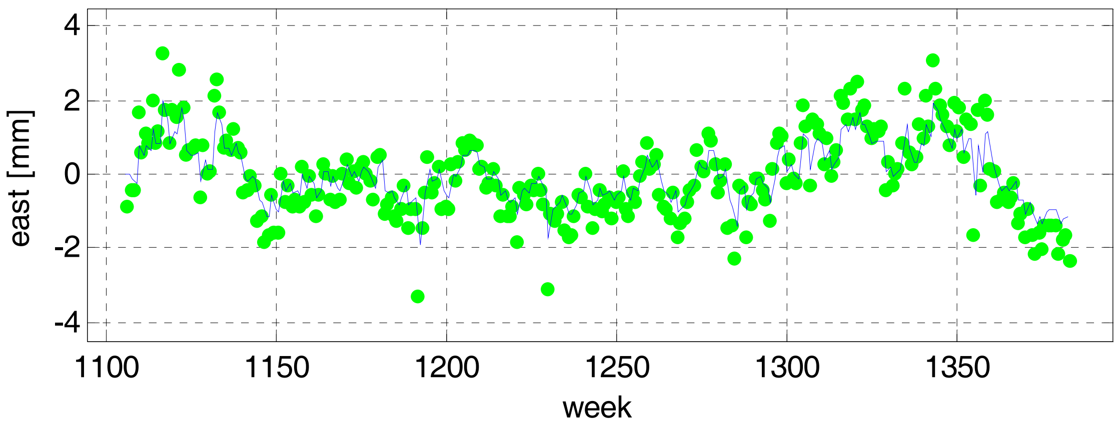


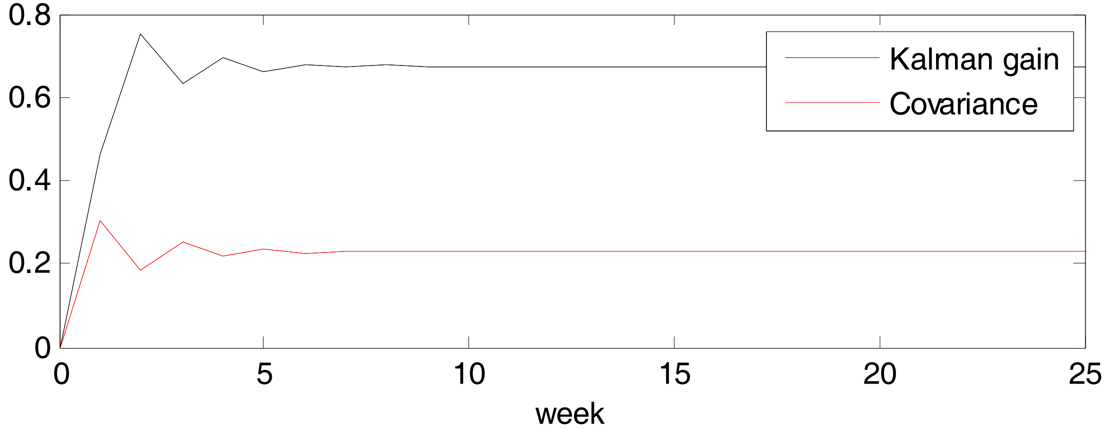
Example: predict the east of BOGO

- Plot raw data (green) and predicted one epoch ahead y_
- Estimate variances of measurement and process noise so that the rms of raw-y_ has a minimum
- We find for process and measurement noise respectively:

0.5 mm and 1.2 mm

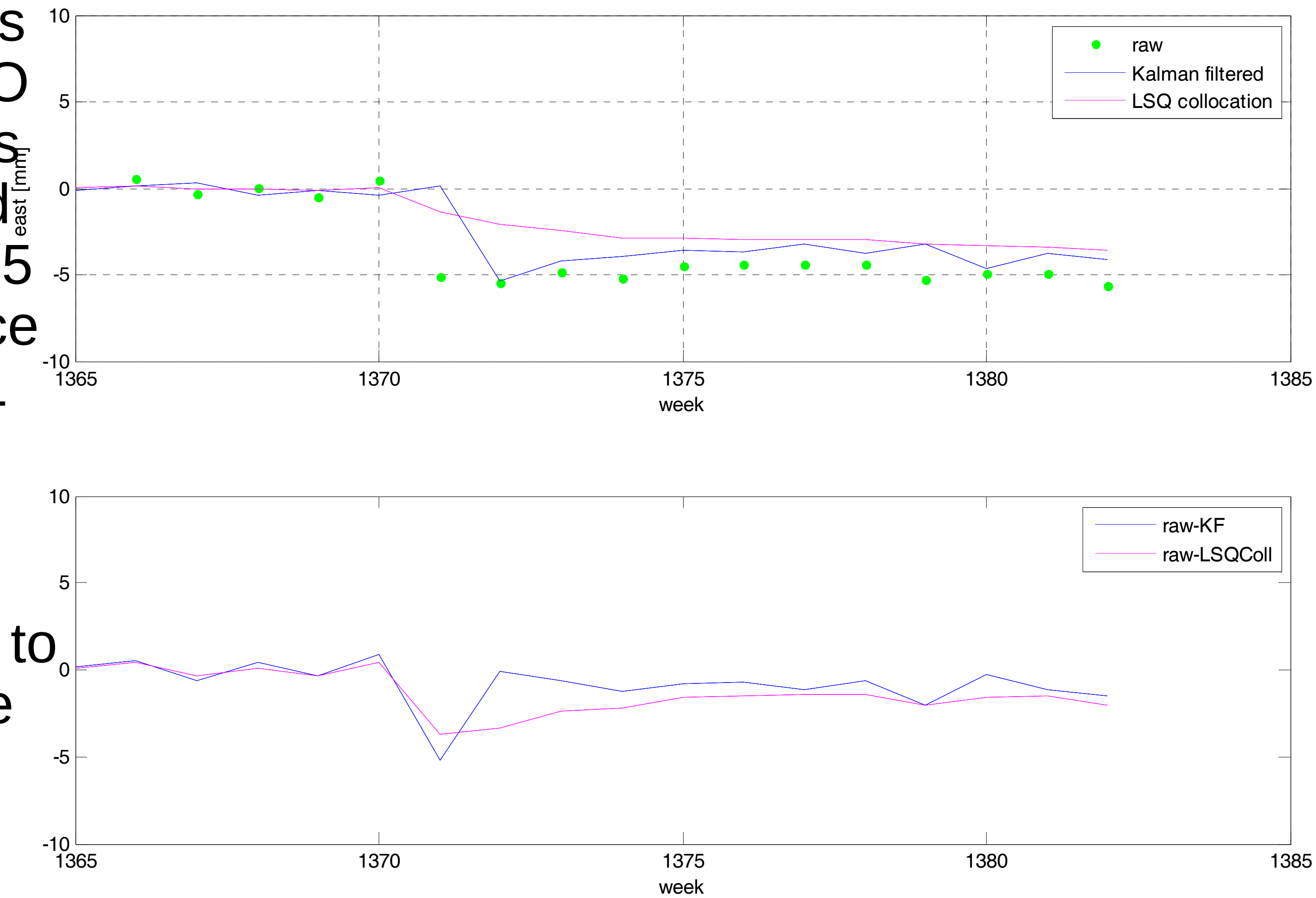
Note: the rms of the raw data was 1.13 mm





Example:detect jump at BOGO

- Time series of BOGO east was modified adding -5 mm since wk 1371
- LSQ coll appears slower than KF to track the change



Conclusion

- Preliminary indication that time series look like Gauss Markov processes (exponential autocorrelation), with time constants of the order of 3-5 weeks
- Filtering the time series by either Kalman or least squares collocation seems to significantly reduce the rms (e.g. 1.1 →0.5 mm).
- The significance of the filtered signal remains to be investigated by correlating to similar filtered signals from other stations
- Where KF seems superior to LSQC is in the detection of a jump (e.g. 5 mm jump)