Comparison of recurrent and vector forecasting

Andrey Pepelyshev

The University Of Sheffield.

Cardiff
September 20, 2010

Contents

- SSA framework
- SSA recurrent forecasting
- SSA vector forecasting
- Comparison of two procedures via simulation
- Comparison for China GDP time series

Embedding

f_1	f_2	f_3	f_4	f_5	f_6	f_7	$\cdots$	f_n	$= F$

f_1	f_2	f_3	$\cdots$	f_{n-L+1}
f_2	f_3	f_4	$\cdots$	f_{n-L+2}
f_3	f_4	f_5	$\cdots$	f_{n-L+3}
$\vdots$	$\vdots$	$\vdots$		$\vdots$
f_L	f_{L+1}	f_{L+2}	$\cdots$	f_n

$= \mathbf{X}$

$\mathbf{X}$ is called a trajectory matrix, $2 \leq L < n - L + 1$

Decomposition of series

$$\mathbf{X} = \sqrt{\lambda_1}U_1V_1^T + \sqrt{\lambda_2}U_2V_2^T + \sqrt{\lambda_3}U_3V_3^T + \ldots$$

$$\Uparrow \qquad \circlearrowright \qquad \Downarrow \qquad \Downarrow \qquad \Downarrow$$

$$F = F_1 + F_2 + F_3 + \ldots$$

- F_i is a trend component
- F_j is a harmonic component
- F_l is a noise component

Extraction of signal

Let F be modeled as a sum of Signal and Noise

$$
\begin{array}{ccccccc}
\mathbf{X} & = & \sum_{i\in I}\sqrt{\lambda_i}U_iV_i^T & + & \sum_{i\notin I}\sqrt{\lambda_i}U_iV_i^T \\
\Uparrow & \circlearrowright & \Downarrow & & \Downarrow \\
F & = & \hat{S} & + & \hat{E}
\end{array}
$$

- $\hat{S}$ is an estimate of Signal
- Usually, $I = \{1, 2, \ldots, r\}$

Forecast of signal

Let F be modeled as a sum of Signal and Noise.

$$\begin{array}{ccccc} \mathbf{X} & = \sum_{i\in I} & \sqrt{\lambda_i}U_iV_i^T & + \sum_{i\notin I} & \sqrt{\lambda_i}U_iV_i^T \\ \Uparrow & \circlearrowright & \Downarrow & & \Downarrow \\ F & = & \hat{S} & + & \hat{E} \end{array}$$

To forecast of Signal, we can use the estimate $\hat{S}$ and the signal subspace generated by $\{U_i\}_{i\in I}$.

Recurrent forecasting

Let Signal satisfy LRF $s_j = \alpha_1 s_{j-1} + \ldots + \alpha_q s_{j-q}$.

- Construct LRF
  $\tilde{s}_j = -(a_1 \tilde{s}_{j-1} + \ldots + a_{L-1} \tilde{s}_{j-L+1})/a_L$

  $$(a_L, \ldots, a_1)^T = \mathbf{U}_{r+1,L} \mathbf{U}_{r+1,L}^T e_L$$

  $\mathbf{U}_{r+1,L} = (U_{r+1} \vdots \ldots \vdots U_L)$, $e_L = (0, \ldots, 0, 1)^T \in \mathbb{R}^L$

  $U_{r+1}, \ldots, U_L$ are orthogonal to the signal subspace
- Apply this LRF to $\hat{S}$ in recurrent way

Optimal LRF

- True LRF of Signal $s_j = \alpha_1 s_{j-1} + \ldots + \alpha_q s_{j-q}$.
- Class of all possible LRF based on the signal subspace

$$\tilde{s}_j = -(a_1 \tilde{s}_{j-1} + \ldots + a_{L-1} \tilde{s}_{j-L+1})/a_L$$

$$(a_L, \ldots, a_1)^T = \mathbf{U}_{r+1,L} \mathbf{U}_{r+1,L}^T h \text{ where } h \in \mathbb{R}^L$$

- Optimal SSA LRF used in recurrent forecasting

$$(a_L, \ldots, a_1)^T = \mathbf{U}_{r+1,L} \mathbf{U}_{r+1,L}^T e_L$$

$$e_L = (0, \ldots, 0, 1)^T \in \mathbb{R}^L$$

Studying LRF

- True LRF of Signal $s_j = \alpha_1 s_{j-1} + \ldots + \alpha_q s_{j-q}$ has the characteristic polynomial

$$p(t) = t^q - \alpha_q - \alpha_{q-1} t - \ldots - \alpha_1 t^{q-1}.$$

- General LRF

$$\tilde{s}_j = -(a_1 \tilde{s}_{j-1} + \ldots + a_{L-1} \tilde{s}_{j-L+1})/a_L$$

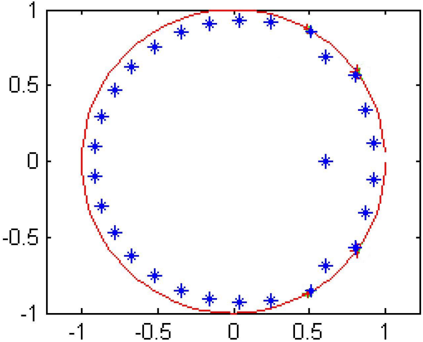
has the characteristic polynomial

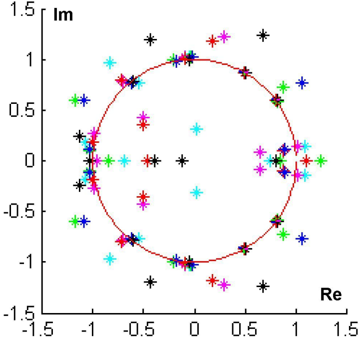
$$\tilde{p}(t) = a_L t^q - a_{L-1} - a_{L-2} t - \ldots - a_1 t^{L-1}$$

with "q signal roots" + "$L-q$ extraneous roots".

Roots of characteristic polynomial

$$s_j = \sin(2\pi j/6) + 0.5\sin(2\pi j/10),\ q=4,\ N=100,\ L=20$$





Left: roots for optimal SSA LRF

Right: roots for LRF with several random h

$$(a_L, \ldots, a_1)^T = \mathbf{U}_{r+1,L}\mathbf{U}_{r+1,L}^T h$$

Optimal property of SSA LRF

Theorem

All extraneous roots of SSA LRF lie inside the unit circle.

Overview of related results can be found in

Usevich, K. (2010). On signal and extraneous roots in Singular Spectrum Analysis. Statistics and Its Interface 3, 281–295.

Vector forecasting

$$\mathbf{X}_{(s)} = \sum_{i=1}^{r} \sqrt{\lambda_i} U_i V_i^T = \begin{array}{|c|c|c|c|c|} x_{11} & x_{12} & x_{13} & \cdots & x_{1K} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2K} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3K} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{L1} & x_{L2} & x_{L3} & \cdots & x_{LK} \end{array}$$

- $K = N - L + 1$
- rank$\mathbf{X}_{(s)} = r$
- Columns of $\mathbf{X}_{(s)}$ generate a signal subspace
- $U_1, \ldots, U_r$ is a basis of this signal subspace

Vector forecasting

x_{11}	x_{12}	x_{13}	$\cdots$	x_{1K}	z_1
x_{21}	x_{22}	x_{23}	$\cdots$	x_{2K}	z_2
x_{31}	x_{32}	x_{33}	$\cdots$	x_{3K}	z_3
$\vdots$	$\vdots$	$\vdots$		$\vdots$	
x_{L1}	x_{L2}	x_{L3}	$\cdots$	x_{LK}	z_L

Add a new column such that it belongs to the signal subspace

- $(z_1, z_2 \ldots, z_{L-1})^T$ is the orthogonal projection of $(x_{2K}, x_{3K}, \ldots, x_{LK})^T$
- $z_L = -(a_1 z_{L-1} + \ldots + a_{L-1} z_1)/a_L$

Vector forecasting

x_{11}	x_{12}	x_{13}	$\cdots$	x_{1K}	z_1			
x_{21}	x_{22}	x_{23}	$\cdots$	x_{2K}	z_2			
x_{31}	x_{33}	x_{33}	$\cdots$	x_{3K}	z_3			
$\vdots$	$\vdots$	$\vdots$		$\vdots$				
x_{L1}	x_{L2}	x_{L3}	$\cdots$	x_{LK}	z_L			

$\underbrace{\qquad\qquad}_{L}$

- To make h-step ahead forecast, we need to add $h + L$ columns
- Perform hankelization and then remove last L elements to obtain forecast

Performance of forecast

Square Deviation for h-step ahead forecast

$$\mathrm{SD} = \left(\frac{1}{h}\sum_{n=N+1}^{N+h}(\hat{s}_n - s_n)^2\right)^{1/2}$$

- s_n is a true signal
- $\hat{s}_n$ is a forecasted time series
- N is a length of time series
- 3000 repetitions were used to compute a distribution of SD

Performance of forecast

Square Deviation for h-step ahead forecast

$$\mathrm{SD} = \Big(\frac{1}{h}\sum_{n=N+1}^{N+h}(\hat{s}_n - s_n)^2\Big)^{1/2}$$

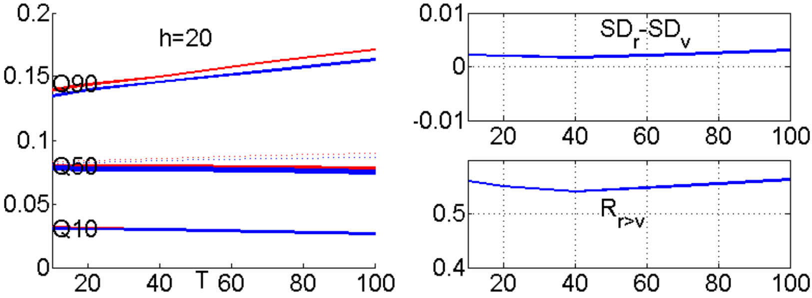
We are going to compare of distributions of SD for recurrent and vector forecasting by

- quantiles
- the average difference of $\mathrm{SD}_r - \mathrm{SD}_v$
- $R_{r>v}$, ratio of cases when $\mathrm{SD}_r > \mathrm{SD}_v$

Model: sin + white noise

$$f_n = \sin\left(\frac{2\pi n}{T}\right) + \varepsilon_n, \;\; n = 1, \ldots, N$$

$$N = 100, \; \sigma = 0.25, \qquad L = 50, \; r = 2$$

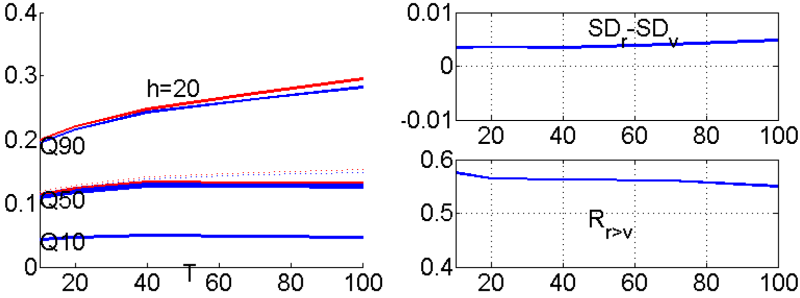


SD for recurrent (red) and vector (blue) forecasting

Model: sin + red noise

$$f_n = \sin\left(\frac{2\pi n}{T}\right) + \varepsilon_n, \;\; n = 1, \ldots, N$$

$$N = 100, \; \sigma = 0.25, \; \varepsilon_n = 0.5\varepsilon_{n-1} + \epsilon_n, \; L = 50, \; r = 2$$



SD for recurrent (red) and vector (blue) forecasting

Conclusion for sin + white/red noise

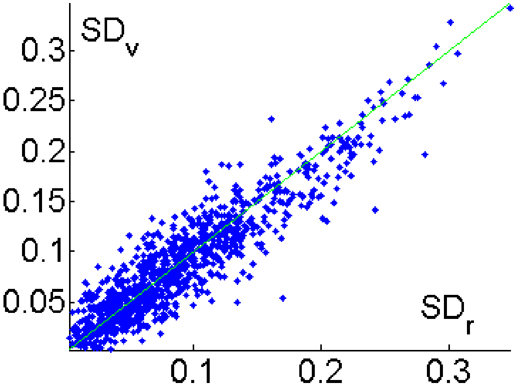
- Recurrent and vector forecasting have roughly the same performance
- SD grows as 'redness' of noise increases, upto 2 times for moderate redness
- Redness yields larger error for low frequency harmonics than for high frequency harmonics

This conclusion remains true for a modulated sine

$$f_n = e^{\pm 0.01 n} \sin\left(\frac{2\pi n}{T}\right) + \varepsilon_n, \;\; n = 1, \ldots, N$$

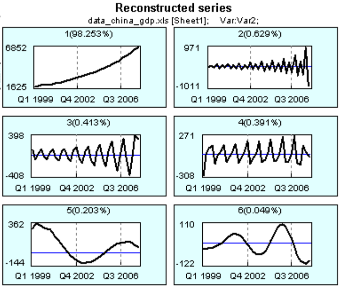
What procedure have to be chosen?

(SD_r,SD_v) for different realizations



Both forecasting procedures yield either small error or large error

China GDP time series



ET 1 is a trend
ET 2 is $\approx$ a harmonic with $T = 2$
ET 3&4 are harmonics with $T = 4$
ET 5 is a minor correction to trend

Setup for forecasts comparison

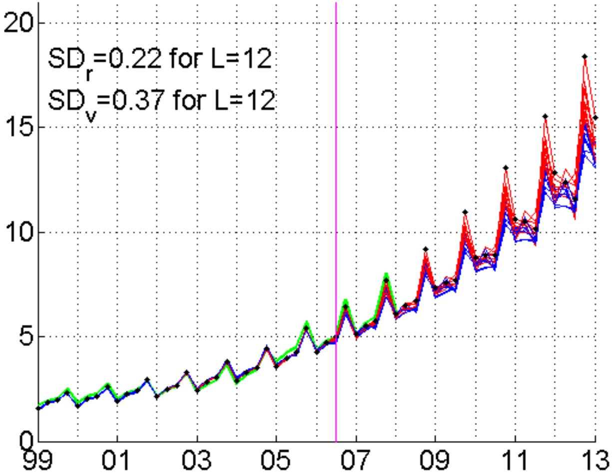
Construct a series of figures with recurrent and vector forecasts starting b steps ago, $b = 0, 1, \ldots, 6$. Forecasts are solely based on decomposition of truncated series.

Each figure shows $(20 + b)$-step ahead forecasts for $r = 4$ and $L = 12, 13, \ldots, 19$.

$$\mathrm{SD} = \Big(\frac{1}{b} \sum_{n=N-b+1}^{N} (\hat{s}_n - f_n)^2\Big)^{1/2}$$

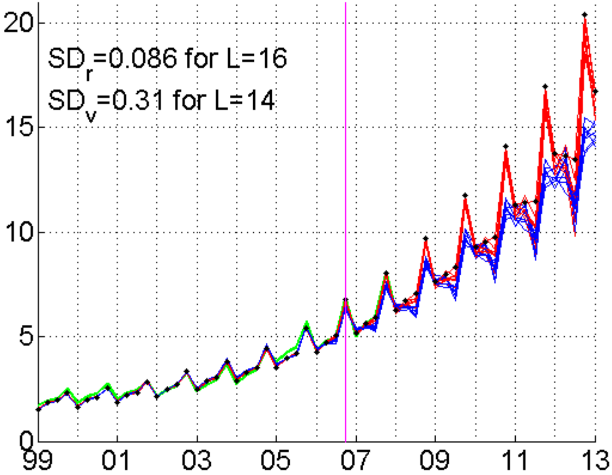
Original time series is given in green.

Forecasts starting 6 steps ago, $r = 4$



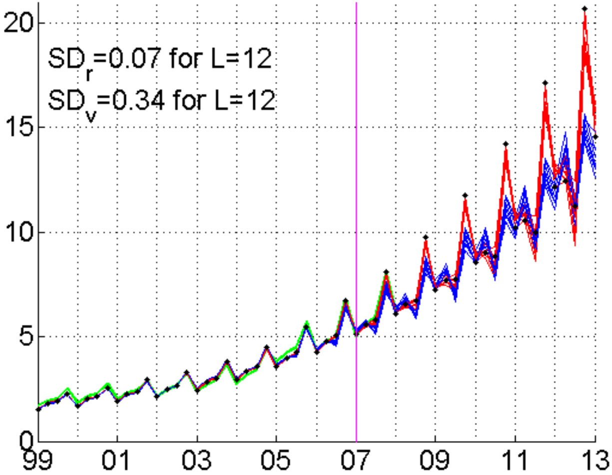
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 5 steps ago, $r = 4$



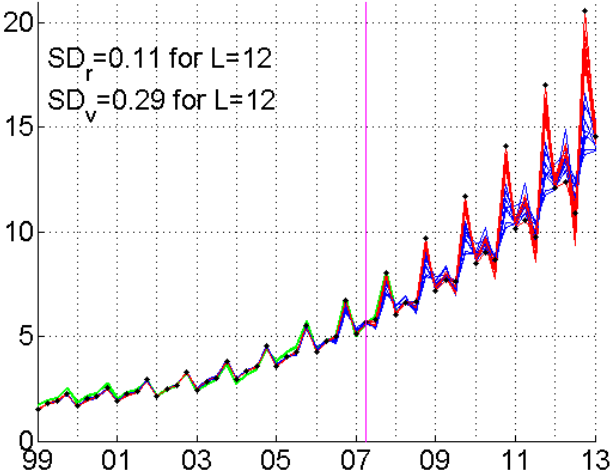
Recurrent (red) and vector (blue) forecasts, $L = 12, ..., 19$

Forecasts starting 4 steps ago, $r = 4$



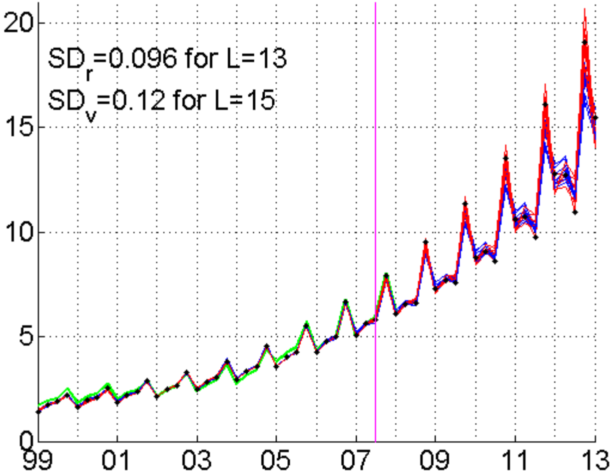
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 3 steps ago, $r = 4$



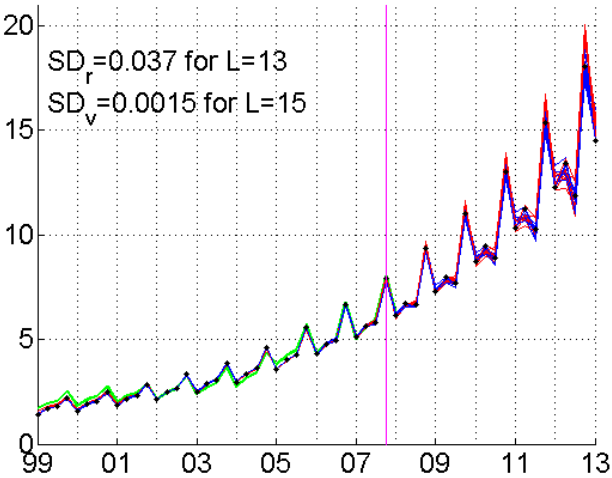
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 2 steps ago, $r = 4$



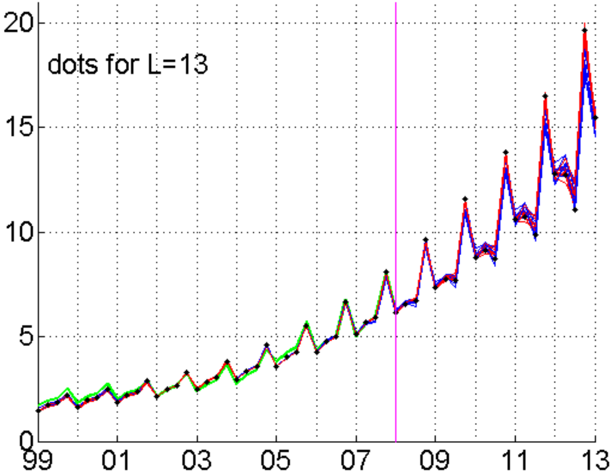
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 1 steps ago, $r = 4$



Recurrent (red) and vector (blue) forecasts, $L = 12, ..., 19$

Forecasts starting 0 steps ago, $r = 4$



Recurrent (red) and vector (blue) forecasts, $L = 12, ..., 19$

Setup for forecasts comparison

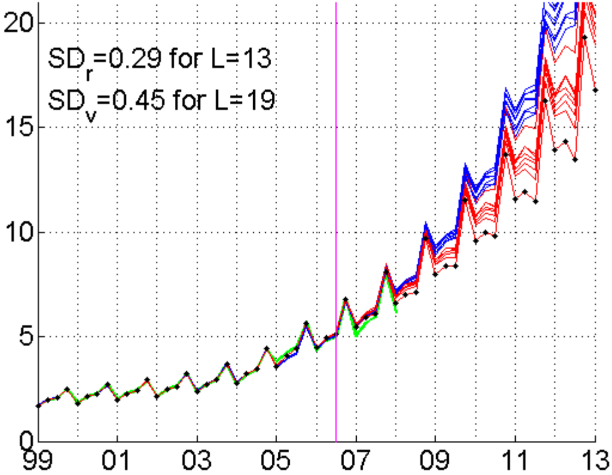
Construct a series of figures with recurrent and vector forecasts starting b steps ago, $b = 0, 1, \ldots, 6$. Forecasts are solely based on decomposition of truncated series.

Each figure shows $(20 + b)$-step ahead forecasts for $r = 5$ and $L = 12, 13, \ldots, 19$.

$$\mathrm{SD} = \left(\frac{1}{b} \sum_{n=N-b+1}^{N} (\hat{s}_n - f_n)^2\right)^{1/2}$$

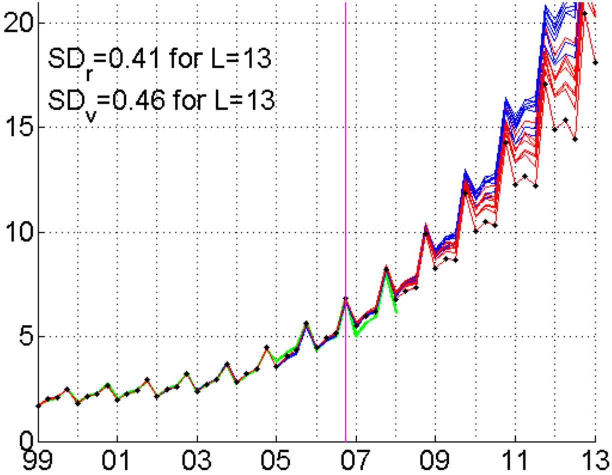
Original time series is given in green.

Forecasts starting 6 steps ago, $r = 5$



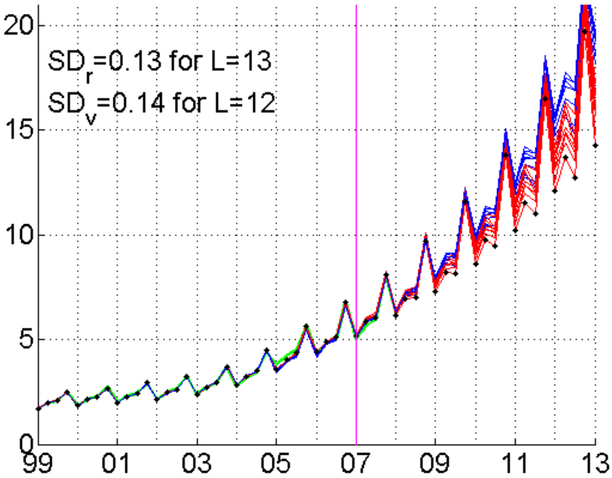
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 5 steps ago, $r = 5$



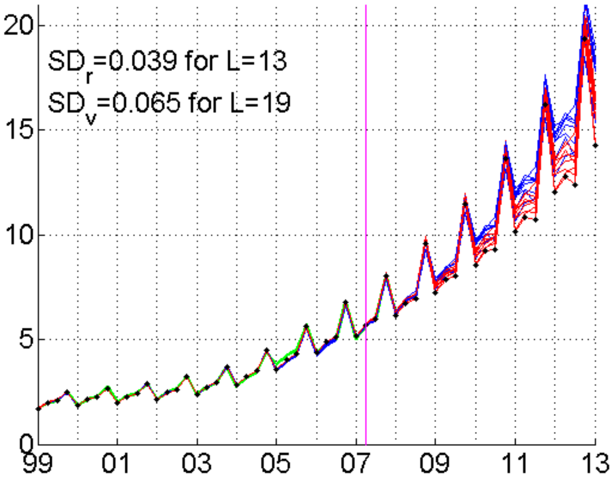
Recurrent (red) and vector (blue) forecasts, $L = 12, ..., 19$

Forecasts starting 4 steps ago, $r = 5$



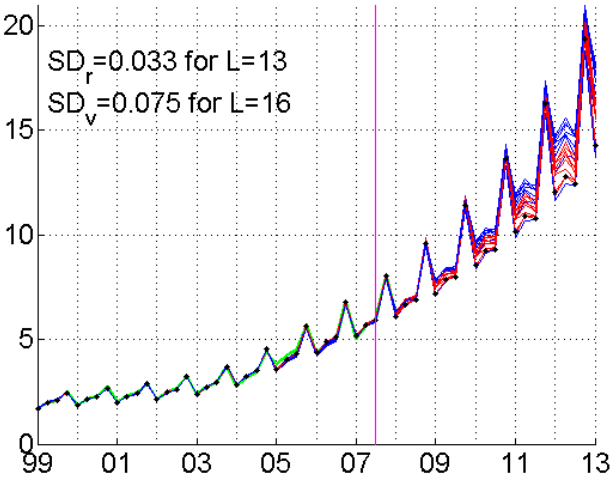
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 3 steps ago, $r = 5$



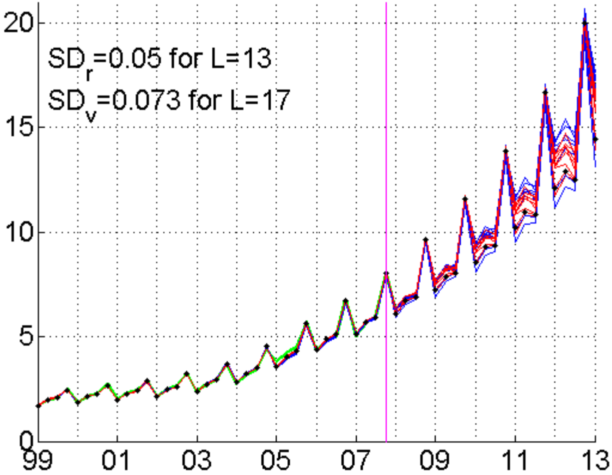
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 2 steps ago, $r = 5$



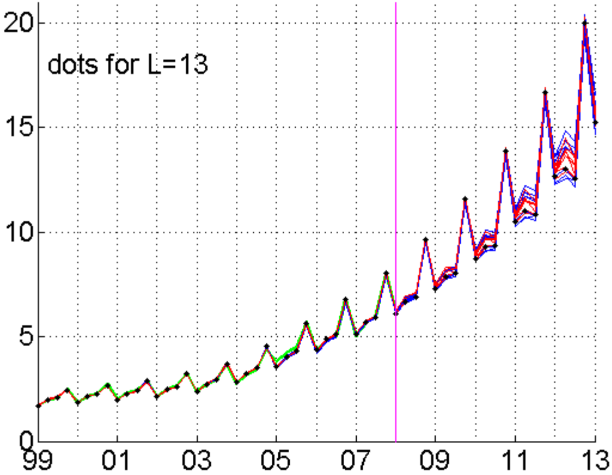
Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 1 steps ago, $r = 5$



Recurrent (red) and vector (blue) forecasts, $L = 12, ..., 19$

Forecasts starting 0 steps ago, $r = 5$

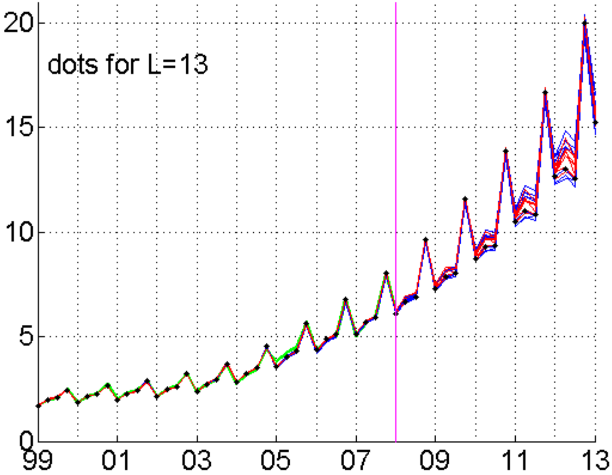


Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Conclusion

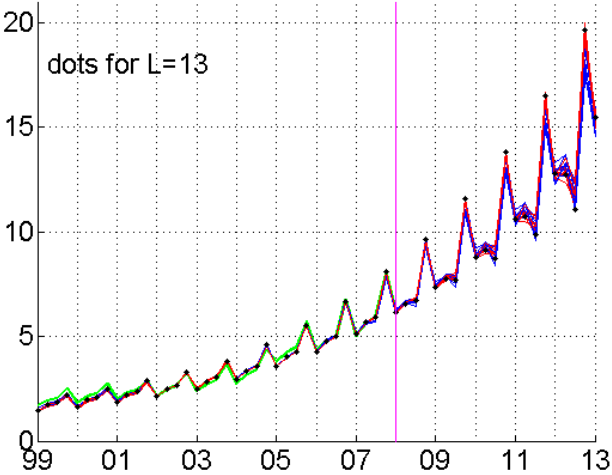
Good choice is $r = 4, 5$ and $L = 13$.

Forecasts starting 0 steps ago, $r = 5$



Recurrent (red) and vector (blue) forecasts, $L = 12, \ldots, 19$

Forecasts starting 0 steps ago, $r = 4$



Recurrent (red) and vector (blue) forecasts, $L = 12, \dots, 19$

Thank you for your attention.