A Hybrid Linear Dynamic System - Neural Network Model for Analysis and Forecasting of Time Series

Akram M. Chaudhry
University of Bahrain
Bahrain

Abstract:

The role of linear dynamic system models in analysis and forecasting of time series, whether bearing white noise or colored noise, is well known. These, well established models, are widely used in studying many real life, simple to complex, phenomena, in a parsimonious manner and yielding optimum results. On the other hand, use of neural network models is also picking up momentum due to their remarkable self learning and training capabilities.

Both of these types of models, naturally, are not holy at all. Like all man made systems, these can't be declared flawless. They have their own, pros and cons.

Keeping in mind evens and odds of both of these models a hybrid linear dynamic system- neural network model is presented for analysis and forecasting of time series and its practical implications discussed. The parameters of this new breed of models are optimally estimated using recursive equations. These equations also provide on line variance learning facility for unknown variances. To generate forecasts from near to far flung future a forecast generator is introduced.

This hybrid linear dynamic system-neural network model is expected to yield better results than the individual parent models.

Key words: Linear Dynamic System models, State Space models, Neural Network models, Hybrid model, White Noise, Colored Noise processes, Identification of a suitable model, ATS(Akram Test Statistic).

#1. Introduction

For analysis and forecasting of discrete time series \( \{Y_t\}_{t=1,2,...,n} \) linear dynamic system (LDS) models of numerous types are frequently being constructed and applied. These models in broader sense may be categorized in to the following two types, depending upon the nature of noise term, white or colored.

Type #1 LDS Models

These models are defined over the quadruple \( \Omega_i= (f, G, V, W) \); such as:

\[
\begin{align*}
y_t &= f \theta_t + v_t ; \quad v_t \sim N(0,V) \\
\theta_t &= G \theta_{t-1} + w_t ; \quad w_t \sim N(0,W)
\end{align*}
\]
are constructed for time series bearing white noise.
Where at time t:
\[ f = (1 \times n) \] is a vector of some known functions of independent variables or constants
\[ \theta = (n \times 1) \] is a state vector of unknown stochastic parameters
\[ G = \text{diag}\{G_i\}_{i=1,2,\ldots,r} \] is a \( (n \times n) \) state or transition matrix having \( n \) number of non-zero eigenvalues \( \{\lambda_i\}_{i=1,\ldots,n} \).
\[ v = \text{white noise, assumed to be independently, identically and normally distributed with mean zero and some constant variance V} \]
\[ w = (n \times 1) \] is the parameter noise vector, having zero mean vector and covariance matrix \( W \), the components of which are as defined by Harrison-Akram(1983) and Akram 1991, 92).

**Type #2 LDS Models**
These models, usually are, defined over the quadruple \( \Omega_2 = (f, L, V, W) \) such as:
\[
\begin{align*}
y_t &= f\theta_t \\
\theta_t &= L\theta_{t-1} + w_t; \quad w_t \sim N(0, W)
\end{align*}
\]
are constructed for time series bearing AR(p) type colored noise processes.
Where at time t, \( f \) and \( \theta \) are as defined earlier except that their dimensions for an AR(p) colored noise process are \( (1 \times (n+p)) \) and \( ((n+p) \times 1) \) rather than the above stated.

\[ L = \{L_{j,k}\} \] is a \( (n \times p) \) state or transition square matrix having \( n \) number of full rank.
\[ a_i \quad \text{for } j=k=i; \quad i=1,2,\ldots,p \]
Where \( L_{ij} = \{\lambda_i\} \quad \text{for } j=k=i+p; \quad i=1,2,\ldots,n \)
\[ 0 \quad \text{for } j=i, \quad k=i+1; \quad i=1,\ldots, (n+p-1) \]
and zero otherwise.

For some known prior of parameter \( \theta \) at time t-1
\[ (\theta_{t-1} | D_{t-1}) \sim N[m_{t-1} \; ; \; C_{t-1}] \]
the posterior of \( \theta \) at time t is
\[ (\theta_t | D_t) \sim N[m_t \; ; \; C_t] \]
is determined using the recursive estimation procedure of Harrison-Akram (1983) and Akram (1992).
#2. Neural Network (NN) Models

Various neural static and dynamic network models with one or more hidden layers has been proposed for linear and non linear systems. From these NN models, having weighted interconnections among neurons, the best model is selected using some forecast accuracy criteria, such as, Mean Square Percent Error (MSPE), Mean Square Error (MSE) and Whiteness of Residuals (WR). Further, many copies of the selected NN model are made and training is carried out using some suitable algorithms accommodating sigmoid function, such as, \( \text{sigmoid}(x) = \frac{1}{1+e^{-x}} \) transformed in to linear form. This function assists the neurons in training and ultimately tuning of the outcomes (forecasts) to some desired level of accuracy and avoid over parameterization in neural networks.

#3. Hybrid LDS-Neural Network Model

This model, a blend of both the neural networks and linear dynamic system models, is developed keeping in mind the merits of both the systems. Considering neural weights \( \omega \) the above LDS model may be written as:

\[
\Omega_H = \{ (\Omega_1 \text{ or } \Omega_2), \omega \} = \{ f, (\Gamma = G \text{ or } L), V, W, \omega \}
\]

To operate the model, first LDS type #1 Model: \( \Omega_1 = (f, G, V, W) \) is applied to time series \( \{y_t\}_{t=1}^{T} \) by setting \( f, G, W, m_0 \) and \( C_0 \) in accordance with the procedure explained by Harrison-Akram (1983) and then the parameters of the model are estimated using the following recursive linear dynamic system recursive equations. These equations at time \( t \) are defined as follows.

\[
R_t = \Gamma C_{t-1} \Gamma' + W
\]

\[
A_t = R_t f \left[ V + f R_t f' \right]^{-1}
\]

\[
C_t = [I - A_t f] R_t
\]

\[
e_t = y_t - f \Gamma m_{t-1}
\]

\[
X_t = \beta_v X_{t-1} + (1 - f A_t) d_t
\]

\[
d_t = \min \{ e_t^2, \xi (V_t + f R_t f') \}
\]

\[
N_t = \beta_v N_{t-1} + 1
\]
\[ V_t = X_t / N_t \]

\[ m_t = \Gamma m_{t-1} + A_t [y_t - f \Gamma m_{t-1}] \]

where at time t:
R is a system matrix,
I is an identity matrix
\( W = \text{diag}\{ W_i \} \) for \( i = 1, \ldots, n \).
A is an updating or gain vector and
All these vectors and matrices are assumed to be compatible in dimensions with their associated vectors and matrices of the system.
\( e_t = y_t - f \Gamma m_{t-1} \) are one step ahead forecast errors.
\( 0 < \beta_v < 1 \) is a smoothing constant related to variance learning.
\( \xi \) a preset constant is usually set equal to 4 or 6

Based on estimate \( m_t \) \( k \)-steps \( (k=1,2,\ldots) \) ahead forecast \( y_{t+k}^\wedge \) are obtained using the \( k \)-steps ahead forecast function.

\[ F_t^{(k)} = f \Gamma^k m_t \]

The suitability of this model is then determined by evaluating one step ahead forecast residuals or errors obtained from the one step ahead forecasts generated by the defined forecast function. For this purpose, whiteness of residuals is established using ATS: Akram Test Statistic (2003).

If the test ensures whiteness of residuals then these initial forecasts are used as input in to the NN Model and then network tuned forecasts are obtained by re-parameterizing the above forecast function with a set of neuron weights \( \omega_j \) and sigmoid function, such as, Logistic(\( x \)) = \((1 + e^{-x})^{-1} \). For more discussion see Hornick et al (1989, 94), Sontag (1993) and Sjoberg (1993).

The weights \( \omega_j \) can be estimated using Prediction Error (PE) method or dynamic system recursive equations similar to the stated above. For training of neural network candidate model with a given LDS architecture the following feed-forward procedure is adopted.
Where at time $t$:

$I_t = \text{Non feedback inputs of the predictor}$

$\theta_{t-1} = \text{Input of the state parameter vector}$

$O_t = \text{Output of the predictor}$

$e_t = y_t^* - f_t$ are the forecasts errors or deviations of forecasts generated by LDS and Neural Models.

For more discussion see Isabelle – Leon (1996).

For generating $k$-steps ahead forecasts, the following forecast function is considered.

Defining $H = \text{diagonal} \{ \omega, \Gamma \}$

$F_t^* (k) = f H^k m_t \quad \text{for } k=1,2,\ldots$}

#4. Practical Implications

For implementation of hybrid model various components of the model are formulated using priors and set up procedure of Akram (1992) and following Blak-Box Modeling technique of Isabelle, et al (1996). This procedure may be viewed in figure #1

This hybrid LDS-Neural model is expected to generate better forecasts than the LDS model due to on line learning and self training capabilities. To ensure optimality of forecasts generated by this model may be evaluated by employing conventional forecast accuracy measuring criteria, such as, MSE: Mean Square Error, MPSE: Mean Percent Square Error or more advanced criteria such as AIC of Akaike (1973) and ATS of Akram (2001). For implementation of ATS see Akram (2001, 07). This process of achieving targeted forecasts may be viewed as a figure #3.
The hybrid model is presented in canonical form. If desired, it may be re-parameterized in to diagonal form by using recursive transformation of Akram (1988). For more discussion see Akram (1991, 92). For self learning and training purpose single layer feed forward network with neuron weights $\omega_j$ is considered. For such purpose, multilayer feed forward networks

To ensure optimality of forecasts generated by this model may be evaluated by employing conventional forecast accuracy measuring criteria, such as, MSE: Mean Square Error, MPSE: Mean Percent Square Error or more advanced criteria such as AIC of Akaike (1973) and ATS of Akram (2001). For implementation of ATS see Akram (2001, 07).
Figure #3

References


