CONSTRAINED FORMULATIONS AND ALGORITHMS FOR STOCK-PRICE PREDICTIONS USING RECURRENT FIR NEURAL NETWORKS

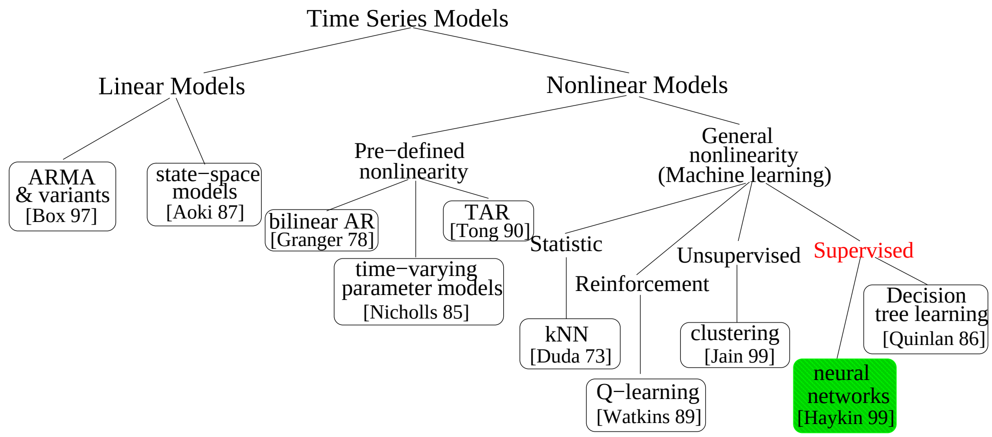
Benjamin W. Wah and Minglun Qian

Department of Electrical and Computer Engineering
and the Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
E-mail:{wah,m-qian}@manip.crhc.uiuc.edu

Outline

- **Existing models for nonlinear time series analysis**
- **Preprocessing for noisy stock-price time series**
- **Constrained formulation**
  - Constraints on individual patterns
  - Constraints on validation sets
  - Constraints on lag period and learning algorithm
- **Violation-guided backpropagation algorithm**
- **Experimental results**
- **Conclusions and future work**

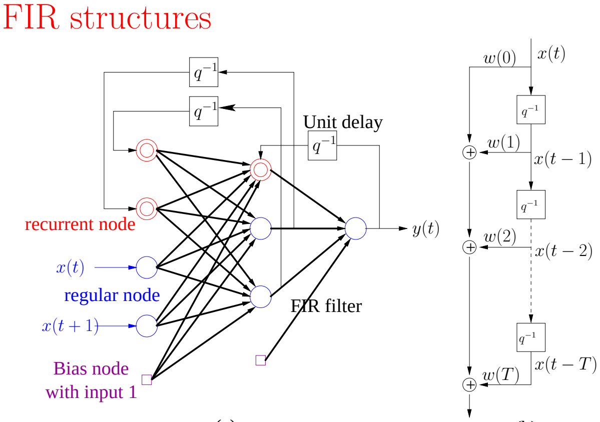
Existing Models for Nonlinear Time Series



- Issues in existing nonlinear supervised learning techniques
  - Single nonlinear objective on training set
  - Cannot enforce individual pattern behavior
- Constraint on individual pattern behavior is desirable

Model Used: Artificial Neural Networks

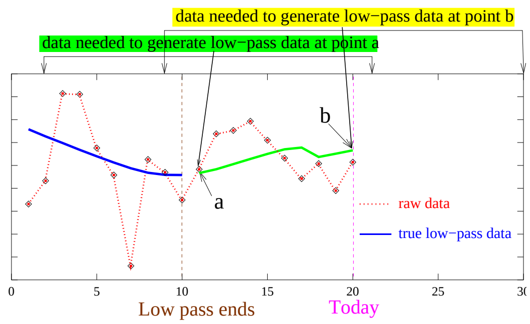
- Architectures
  - Memory-based (e.g. time-delayed, FIR), or recurrent-based
  - Issue: cannot provide both accurate short-term memory and indefinite long-term memory
  - Proposed recurrent FIR neural network (RFIR) with connections modeled by FIR structures



(a) (b)

High Frequency Random Noise in Stock Prices

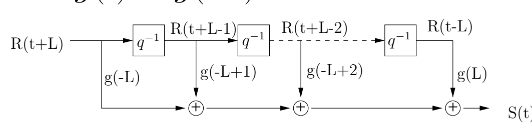
- Random noise presented in stock time series [Zheng99,Hellstrom97]
  - Eliminated by low-pass filter
- Issues
  - Lag: filtering process utilizes future data to generate low-pass data and causes low-pass data to lag behind original data
  - High frequency data: random noise and not predictable



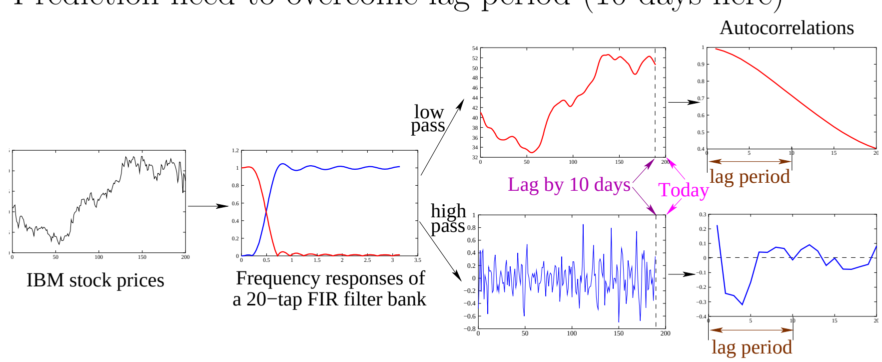
- Predict low-pass data in the lag period before predicting into the future

Illustration of Filtering Process

- Symmetric FIR filter: $g(l) = g(-l)$

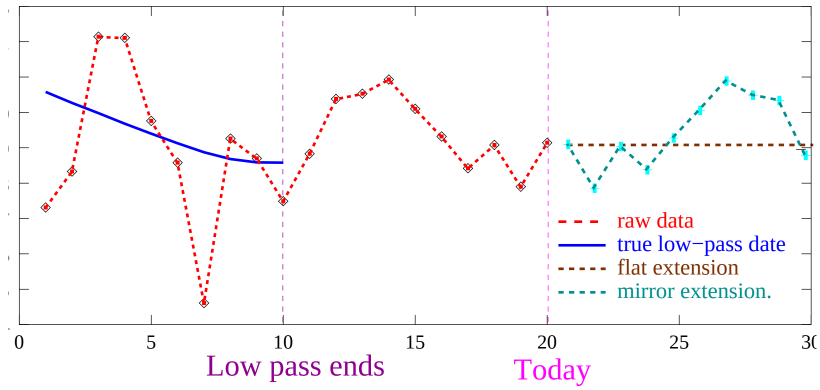


- Low-pass and high-pass data
  - Prediction need to overcome lag period (10 days here)



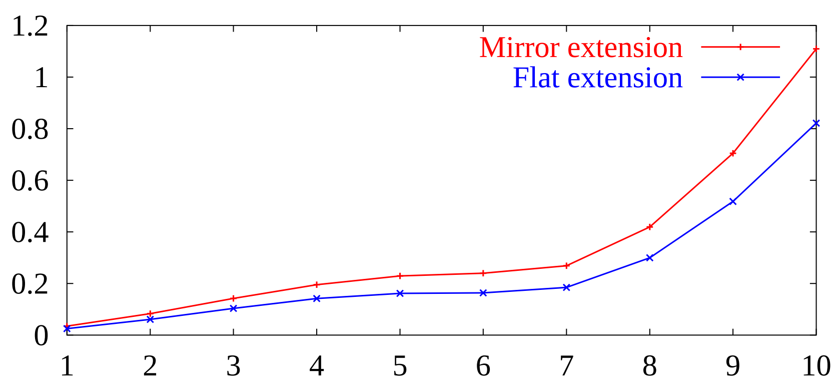
Previous Work for Handling Lags

- Extending raw data based on pre-defined assumptions [Masters 95]
  - Flat extension
  - Mirror extension



Issues in Existing Methods for Lag Problem

- Issues
  - Large mean of absolute errors (MAE) between predictions and targets at the end of lag period
- Need to predict last three data in the lag period



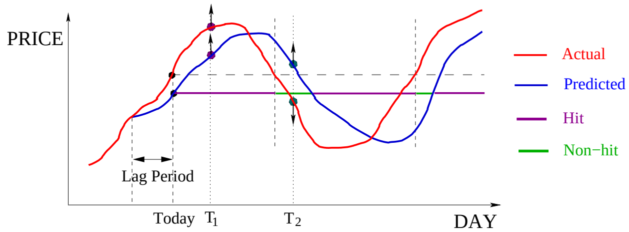
Performance Metrics

- Normalized Mean Square Error ($nMSE$)

$$nMSE = \frac{1}{\sigma^2 n} \sum_{t=t_1}^{t_1+n-1} (o(t) - d(t))^2, \quad (1)$$

  - σ^2: the variance of the true time series during time $[t_1, t_1 + n - 1]$
  - $o(t)$: predicted output at time t
  - $d(t)$: desired output at time t

- Hit



- Hit rate: probability of hit for a prediction

Constraints on Individual Patterns

- Each pattern treated as a new constraint:

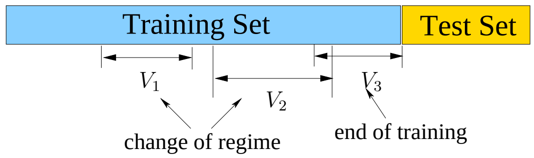
$$h_t^p(w) = (o_t(w) - d_t)^2 \leq \tau$$

  - τ: small positive number

- Advantages over traditional unconstrained formulation
  - Violated patterns guide search out of local minima

Constraints on Multiple Cross-Validation Sets

- Multiple validation sets within training set allowed



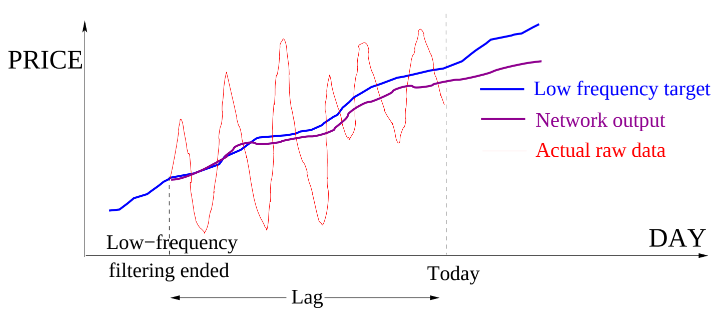
- Validation errors treated as constraints for each horizon i
  - Mean absolute error (MAE) over multiple validation sets:
  $$h_i^v(w) \leq \tau_i^v$$
  - Average of non-hit rate (1 − hit rate):
  $$h_i^r(w) \leq \tau_i^r$$
- Advantages over traditional cross-validation
  - Training patterns fully used
  - Optimizing learning errors and validation errors simultaneously

Constraints in Lag Period

- Outputs in the lag period is constrained to be centered by raw data

$$h^{lag} = \sum_{t=t_0-m+1}^{t_0} \hat{S}(t) - R(t) \leq \tau^{lag}, \qquad (3)$$

where $\hat{S}(t)$: network output at t, t_0: current day, m: number of lags.



- Advantages: Prevent predictions in late lag period from drifting away from desired values.

Constrained Formulations for ANN

- Constrained formulation

$$\begin{array}{ll} \min_w & E(w) = \frac{1}{n}\sum_{t=1}^{n} max\{(o_t(w)-d_t)^2-\tau, 0\} \\ \text{s.t.} & h_t(w) = (o_t(w)-d_t)^2 \le \tau, \\ & h_i^v(w) = \tau_i^I, \\ & h_i^r(w) = \tau_i^S, \\ & h^{lag}(w) = \tau^{lag}. \end{array} \quad (4)$$

- Issues
  - Nonlinear constrained global optimization problem
  - Some constraints not in closed forms and hard to compute gradients
- Eq. (4) solved by violation-guided back-propagation (VGBP) based on Theory of Lagrange multipliers for discrete constrained optimization [Wah & Wu]

Lagrange Multipliers for Discrete Optimization

- Transform Eq. (4) into augmented Lagrangian function:

$$\begin{aligned} L(w,\lambda) = \ & E(w) + \sum_{t=1}^{n}\left(\lambda_t max\{0, h_t-\tau\} + \tfrac{1}{2}max^2\{0, h_t-\tau\}\right) + \\ & \sum_{i}\sum_{j=v,r}\left(\lambda_i^j max\{0, h_i^j-\tau_i^j\} + \tfrac{1}{2}max^2\{0, h_i^j-\tau_i^j\}\right) + \\ & \lambda^{lag} max\{0, h^{lag}-\tau^{lag}\} + \tfrac{1}{2}max^2\{0, h^{lag}-\tau^{lag}\} \end{aligned} \quad (5)$$

- Theory of Lagrange Multipliers for discrete optimization [Wah & Wu]
  - Solution to (4) is equivalent to saddle point of (5)
- Saddle point
  - Local min. of $L(w,\lambda)$ in w subspace and local max. in λ subspace

Violation-Guided Backpropagation

- Gradient descents and stochastic acceptances in w subspace by VGBP
  - Using BP to generate approximate gradient for $L(w, \lambda)$ (not $E(w)$)
  - Accepting trial points with Metropolis probability

$$A_T(\mathrm{w}', \mathrm{w})|_\lambda = exp\left\{\frac{(L(\mathrm{w})-L(\mathrm{w}'))^+}{T}\right\} \tag{6}$$

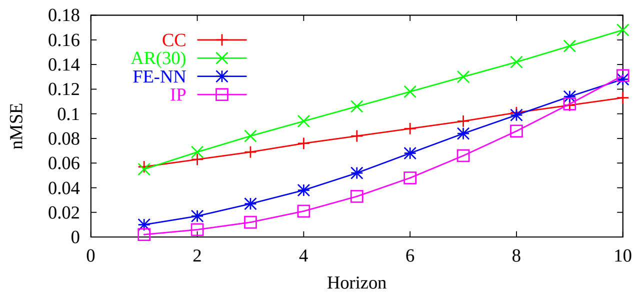
  where $x^+ = min\{0, x\}$ and T is a fixed parameter (temperature).
- Gradient assents in λ subspace by deterministic increases of λ
  - Big violation $\Rightarrow$ increased $\lambda \Rightarrow$ more contribution to gradient
- Relax-and-Tighten technique to speed up convergence [Wah & Qian]
  - Set initial τ's loose enough
  - Gradually tighten τ's as loose constraints are satisfied.

Experiments Setup

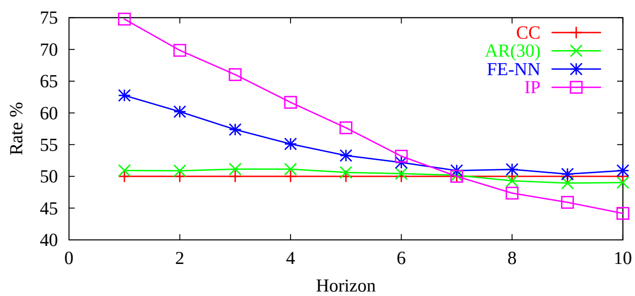
- Predictors compared
  - CC: carbon copy the most recently available data
  - AR: Autoregression
  - FE-NN: Proposed neural network predictor
  - IP: Ideal predictor by using 7 true data in lag and trained by VGBP (approximate upper bound for predictions)
- Stocks
  - Citigroup (Symbol **C**), IBM (**IBM**), Exxon-Mobil (**XOM**)
  - Duration: 04/1997 to 03/2002

Predictions for Citigroup

- *nMSE*



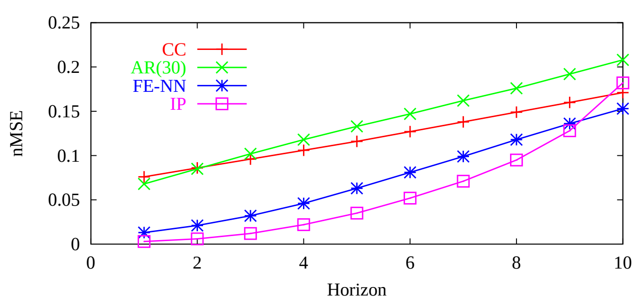
- Hit rate



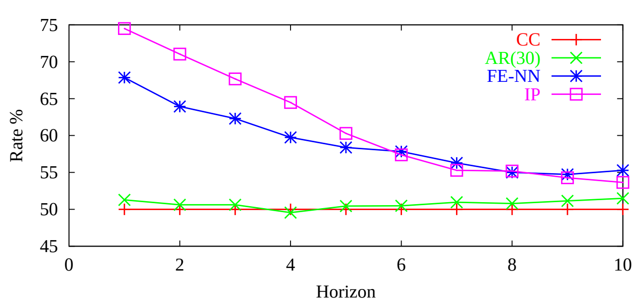
Stock Price Predictions

Predictions for IBM

- *nMSE*

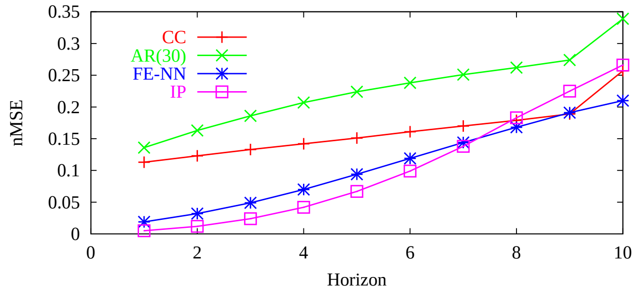


- Hit rate

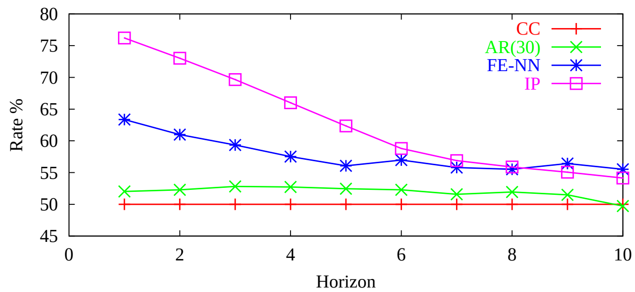


Predictions for Exxon-Mobil

- $nMSE$



- Hit rate



Comments on Hit Rates

- Significantly better than random walk
  - Random walk having a probability of $p = 0.5$ that a guess is correct

$$\text{Prob}(\text{Hits} = \boldsymbol{k}|\boldsymbol{n} \text{ predictions}) = \frac{\boldsymbol{n!}}{\boldsymbol{k!(n-k)!}}\boldsymbol{0.5^n}$$

  - $\text{Prob}(\text{Hits} < k|n)$ follows binomial distribution
  - Some probabilities
    - $\text{Prob}(\text{Hits} > 660|1100) = 1.15 \times 10^{-11}$ (hit rate > 0.6)
    - $\text{Prob}(\text{Hits} > 605|1100) = 4.05 \times 10^{-4}$ (hit rate > 0.55)

  ⇒ FE-NN predictor is significantly better than random walk

- Results presented in most literatures have next-day hit rates below 55% [Gutjahr 97, Hellstrom 2000]

Conclusions

- Systematic study of lag effect due to low-pass filtering
- Proposed constraints in lag period to improve prediction quality
- Proposed constrained formulation for noisy stock-price time series
- Much better prediction performance than traditional autoregression