INTELLIGENT MINING FOR TIME SERIES PREDICTIONS (AND ITS APPLICATIONS IN STOCK MARKET PREDICTIONS)

Benjamin W. Wah

Department of Electrical and Computer Engineering
and the Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801, USA
http://manip.crhc.uiuc.edu

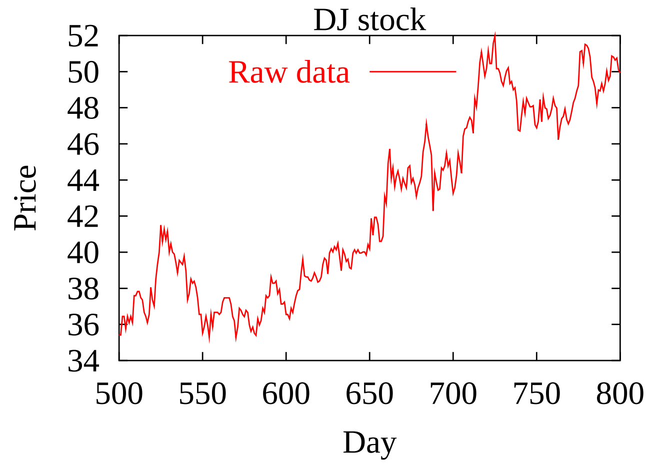
December 21, 2002

Outline

- Market-trend prediction problem
  - Time series predictions
  - Metrics
- Signal processing of time series
  - Lags in predictable low-frequency components
- Data mining techniques
  - Intelligent mining and major design issues
  - Prediction agents
- Constrained optimizations using neural networks
  - Lagrange multipliers for discrete constrained optimization
- Some sample results

Time Series Predictions

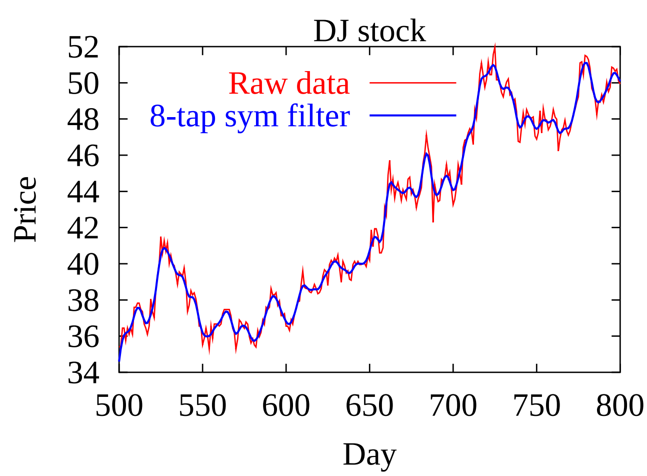
- Prediction of future values based on a sequence of past (and hopefully correlated) values
  - Stock market predictions
  - Product failures
  - Occurrence of sunspots
  - Census data classification
  - Earthquake predictions
  - plus many others



Intelligent Mining for Time Series Predictions Introductions

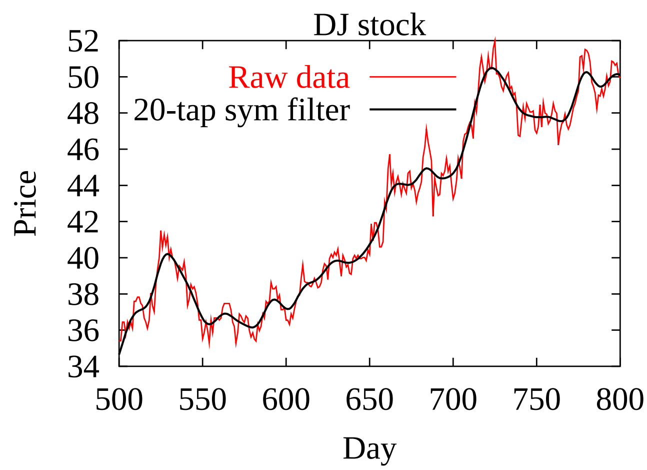
Time Series Predictions

- Prediction of future values based on a sequence of past (and hopefully correlated) values
  - Stock market predictions
  - Product failures
  - Occurrence of sunspots
  - Census data classification
  - Earthquake predictions
  - plus many others



Time Series Predictions

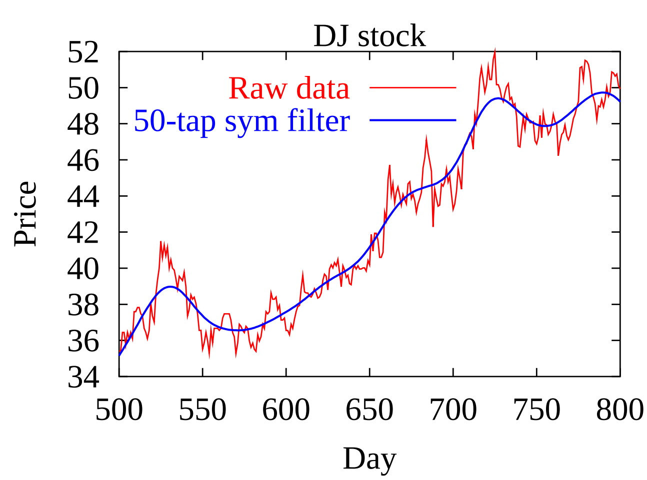
- Prediction of future values based on a sequence of past (and hopefully correlated) values
  - Stock market predictions
  - Product failures
  - Occurrence of sunspots
  - Census data classification
  - Earthquake predictions
  - plus many others



Intelligent Mining for Time Series Predictions Introductions

Time Series Predictions

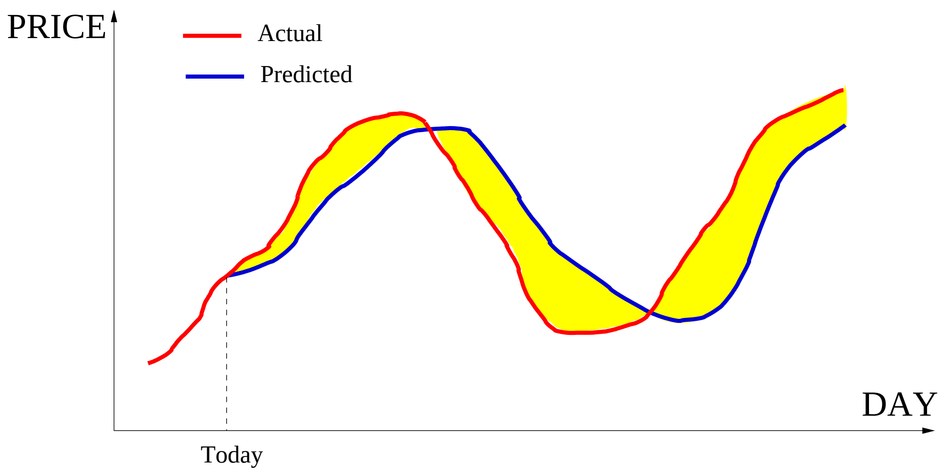
- Prediction of future values based on a sequence of past (and hopefully correlated) values
  - Stock market predictions
  - Product failures
  - Occurrence of sunspots
  - Census data classification
  - Earthquake predictions
  - plus many others



Benjamin W. Wah 2

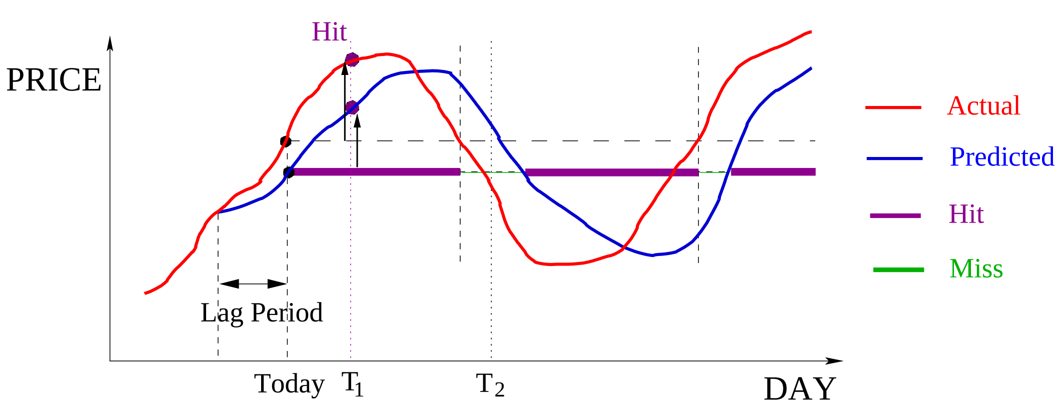
Metrics

a) Sum of squared errors between original and predicted curves



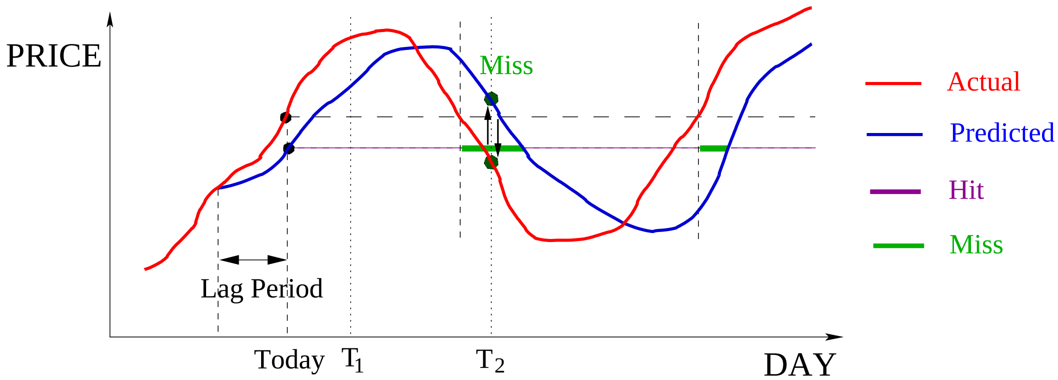
Metrics (cont'd)

b) Hit rate: fraction of consistent trend predictions
(Hit: consistency defined by relative trends)



Metrics (cont'd)

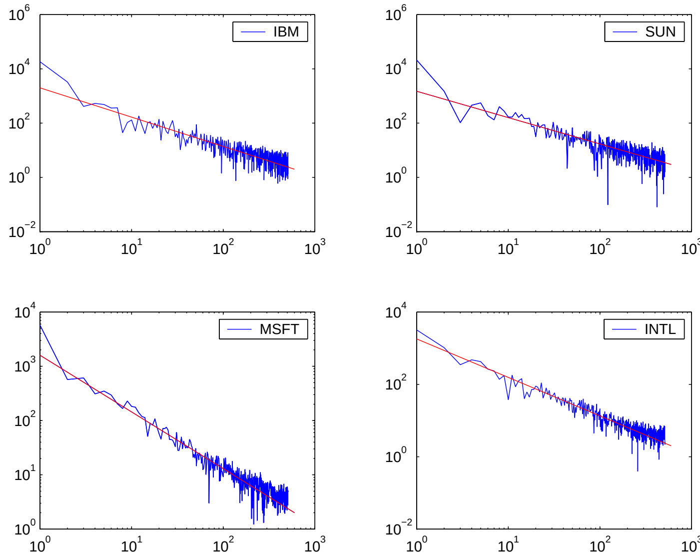
b) Hit rate: fraction of consistent trend predictions
(Hit: consistency defined by relative trends)



Outline

- Market trend prediction problem
  - Time series predictions
  - Metrics
- Signal processing of time series
  - Lags in predictable low-frequency components
- Data mining techniques
  - Intelligent mining and major design issues
  - Prediction agents
- Constrained optimizations using neural networks
  - Lagrange multipliers for discrete constrained optimization
- Some sample results

FFT Transformations of 1024 Daily Closing Prices

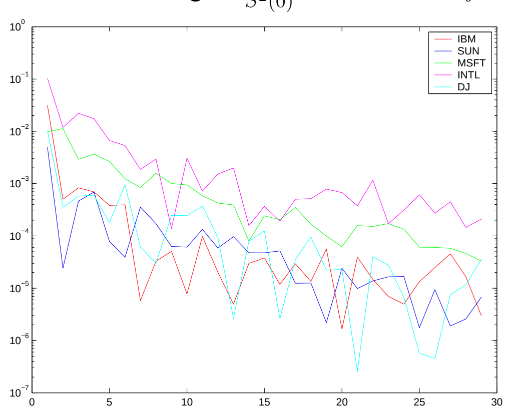


Random walks, stock-price movements, exchange rates follow the $\frac{1}{f}$ line

Dow Jones Theory for Stock Price Movements

- Detect
  - Primary trends: changes that are larger than 20%, typically lasting more than a year
  - Secondary trends: $\frac{1}{3}$ to $\frac{2}{3}$ relative change over primary trends, typically lasting a few months
- Ignore minor trends

Relative Energies $\frac{S^2(f)}{S^2(0)}$ of Lowest 29 f



Filtering of Time Series

- Random noise in time series is not predictable [Zheng99,Hellstrom97]
  - Decompose signals into additive short-term noise and long-term trends, since most energy is in low frequencies

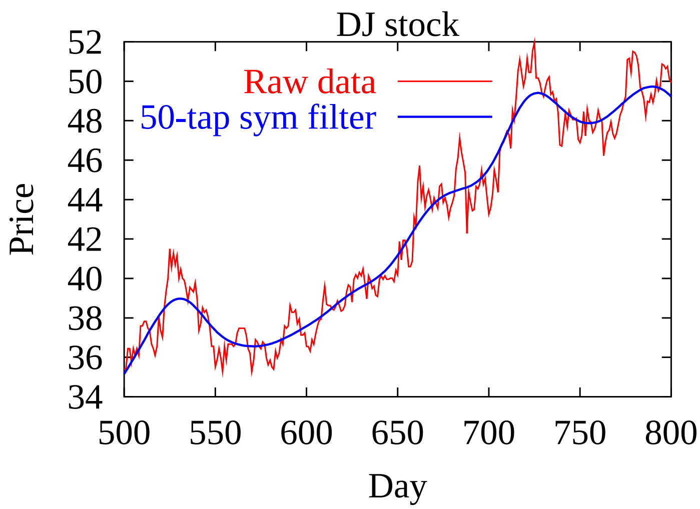
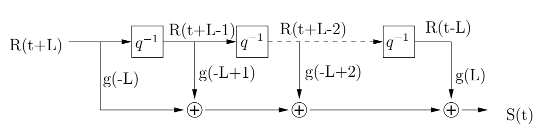
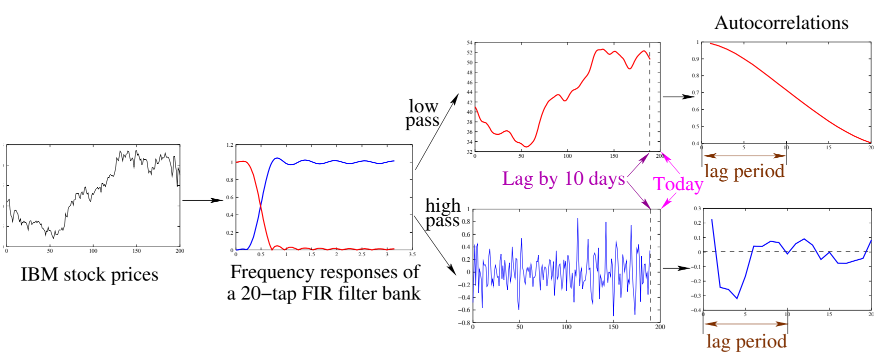


Illustration of Filtering Process

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



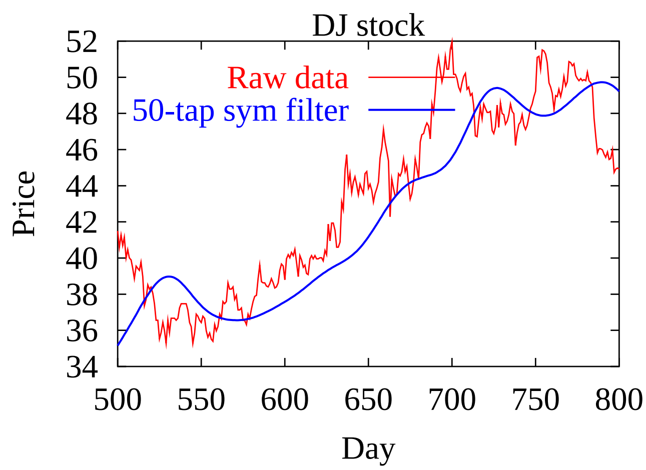
- Low-pass and high-pass data
  - Prediction need to overcome lag period (10 days here)



Lags due to Low-Pass Filtering

- Filtering uses future data to generate low-pass data that lags behind original data
  - High frequency data: random noise and not predictable

A lag of 25 days for a 50-tap filters



Filter Banks

- Multi-band filter banks
  - Equal width for each band and maximally decimated
- Multi-resolution wavelet transforms
  - Exponentially larger passbands from low to high frequencies
  - Perfect reconstruction at time t is only related to information at time t of different scales, with no error propagation
  - Shift variant: Decomposed outputs depend on the origin for decimations
- Redundant (Á Trous) wavelet transforms
  - Similar to multi-resolution wavelet transforms, except on different constraint on wavelet function and no decimation (more storage requirement)
  - Shift invariant: statistical estimators are not sensitive to the choice of origin

Redundant (Á. Trous) Wavelet Transforms

Algorithm

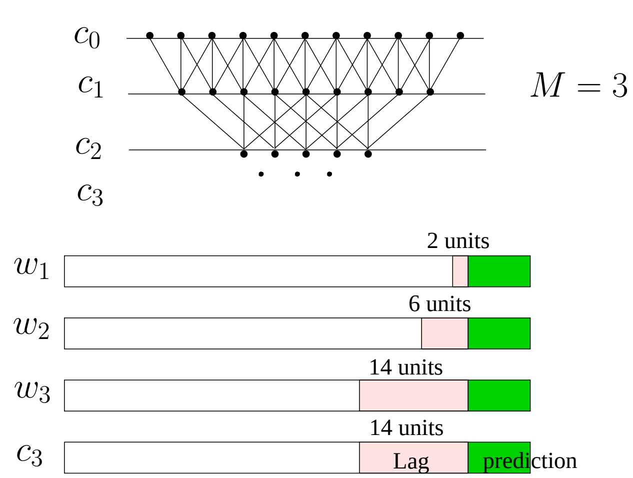
set $c_0(t) = x(t)$;
set $M =$ total number of channels;
select low-pass filter $h(\cdot)$;
for $j \leftarrow 1$ **to** M **do**
  $c_j(t) = \sum_l h(l) c_{j-1}(t - 2^{j-1} l)$;
  $w_j(t) = c_{j-1}(t) - c_j(t)$;
end_for

Properties

- No decimation: redundant transform
- Reconstruction of $x(t)$ with no lag: $c_0(t) = c_M(t) + \sum_{j=1}^{M} w_j(t)$

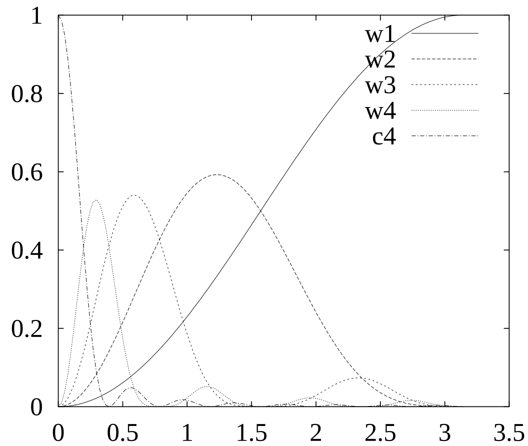
Redundant WT Using Symmetric LP Filters

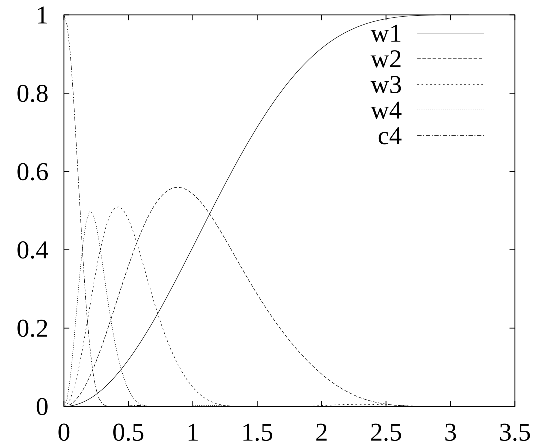
Example: $B(2) = \{h(-1) = 0.25, h(0) = 0.5, h(1) = 0.25\}$



Lags in all bands

Examples of Frequency Response Using Symmetric LP Filters

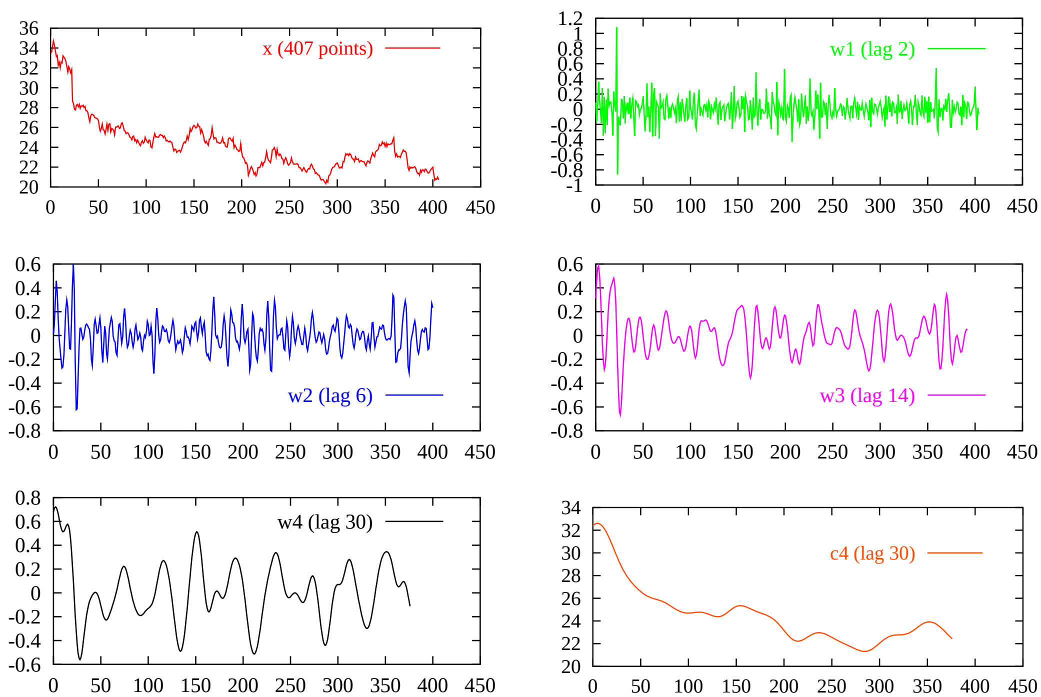




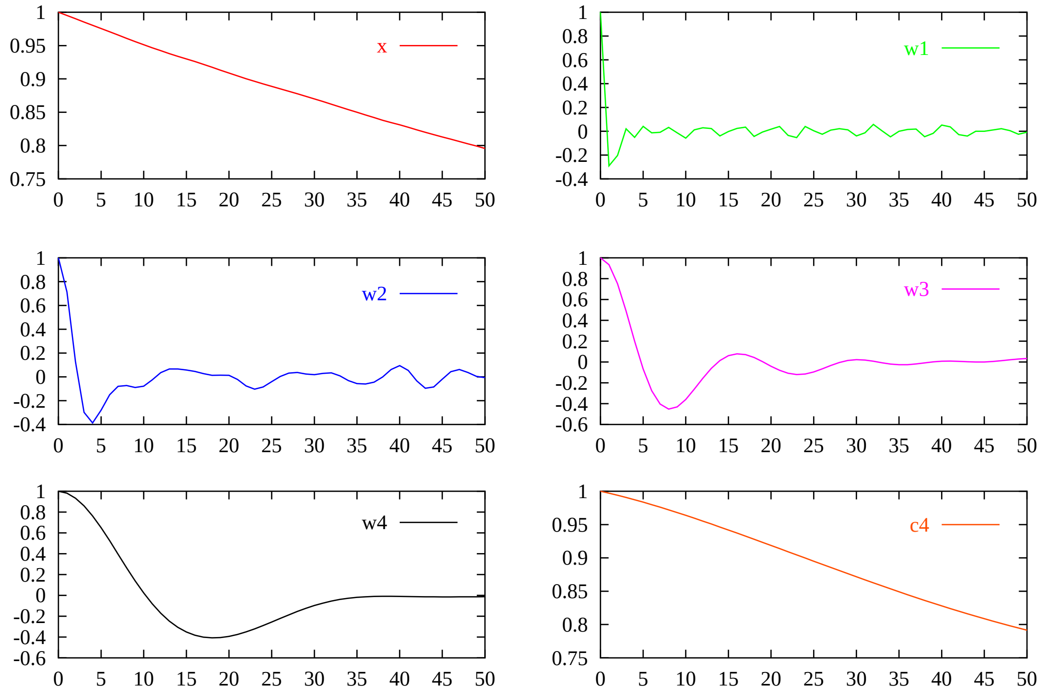
$B2 = \{h(-1) = h(1) = 0.25,$
$h(0) = 0.5\}$

$B3 = \{h(-2) = h(2) = 0.0625,$
$h(-1) = h(1) = 0.25,$
$h(0) = 0.375\}$

Example of Applying WT with Symmetric B3 to IBM Stock Trace



Corresponding Auto-correlation Functions



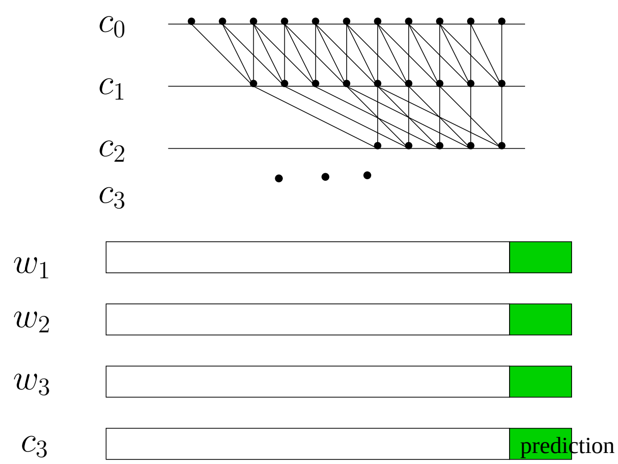
Relationship Between Lags and ACF

High-frequency components have lags longer than sequence of correlated points

	WT with Symmetric LP Filters		
Signal	Days with $ACF > 0.5$		Lag
	IBM	MSFT	
W_1	0	0	2
W_2	1	1	6
W_3	2	2	14
W_4	6	5	30
C_4	50+	50+	30

Redundant WT Using Asymmetric LP Filters

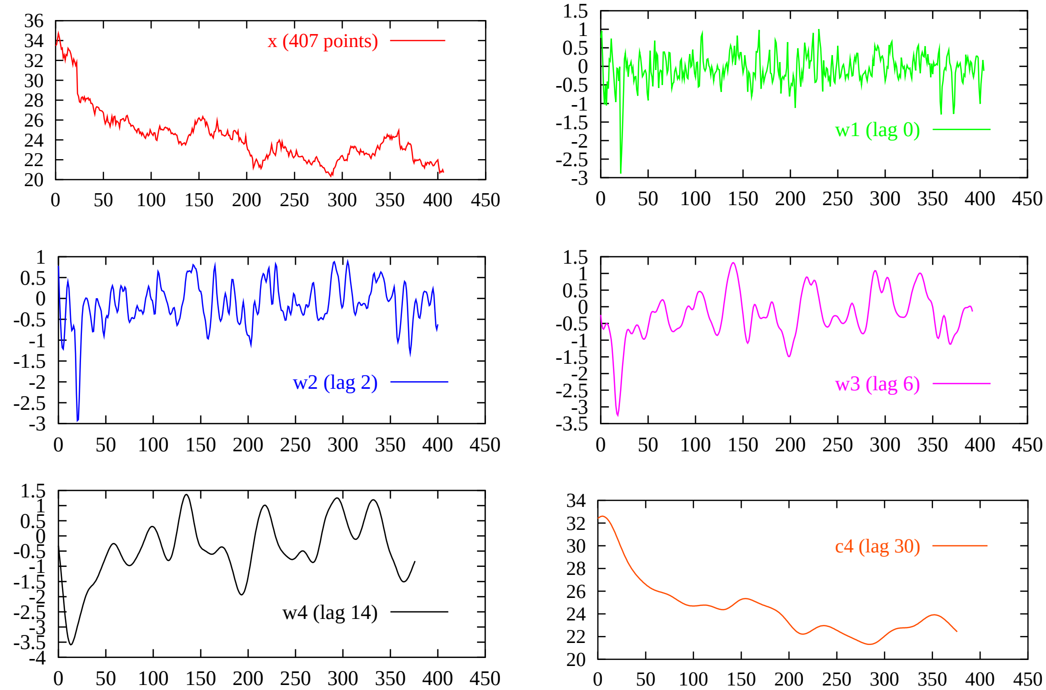
Example: $B(2) = \{h(0) = 0.25, h(1) = 0.5, h(2) = 0.25\}$



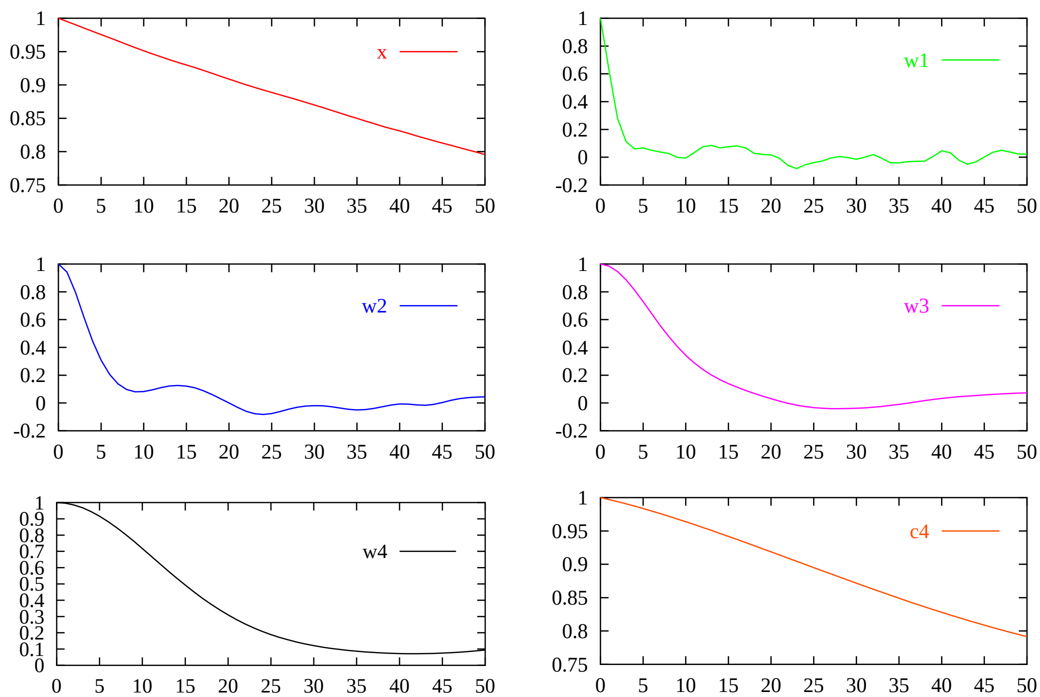
Comments

- $c_j^s(t)$ obtained using symmetric LP filters is a shifted version of $c_j^a(t)$ obtained by corresponding asymmetric LP filters
  For example:
  $$c_1^a(t) = c_1^s(t-1)$$
  $$c_2^a(t) = c_2^s(t-3)$$
  $$\cdots$$
- $w_j^s(t)$ is related to $w_j^a(t)$
  For example:
  $$\begin{aligned} w_2^a(t) &= c_1^a(t) - c_2^a(t) \\ &= c_1^s(t-1) - c_2^s(t-3) \\ &= (c_1^s(t-1) - c_1^s(t-3)) + (c_1^s(t-3) - c_2^s(t-3)) \\ &= (c_1^s(t-1) - c_1^s(t-3)) + w_2^s(t-3) \end{aligned}$$
- Equivalent shifts of corresponding bands filtered by symmetric filters
  $\Longrightarrow$ using these functions alone leads to similar (but smaller) lags

Example of Applying WT with Asymmetric B3 to IBM Stock Trace



Corresponding Auto-correlation Functions



Key Observations

- High-frequency components have lags longer than sequence of correlated points

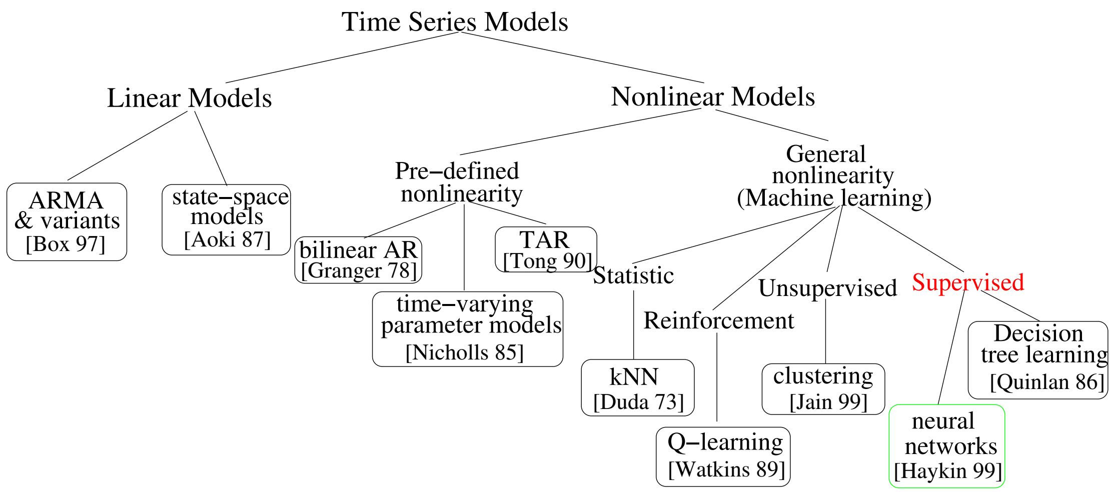
Signal	WT with Symmetric LP Filters			WT with Asymmetric LP Filters		
	Days with $ACF > 0.5$		Lag	Days with $ACF > 0.5$		Lag
	IBM	MSFT		IBM	MSFT	
W_1	0	0	2	1	1	0
W_2	1	1	6	3	2	2
W_3	2	2	14	7	6	6
W_4	6	5	30	14	15	14
C_4	50+	50+	30	50+	50+	30

- Additional information within lag, such as price of fluctuations and volume of transactions, may be used to augment learning and prediction mechanisms
- Transformed objective (e.g. return function $\frac{S(t)-S(t-5)}{S(t-5)}$) and better filters may help improve (short-term or long-term) ACF with respect to lag

Outline

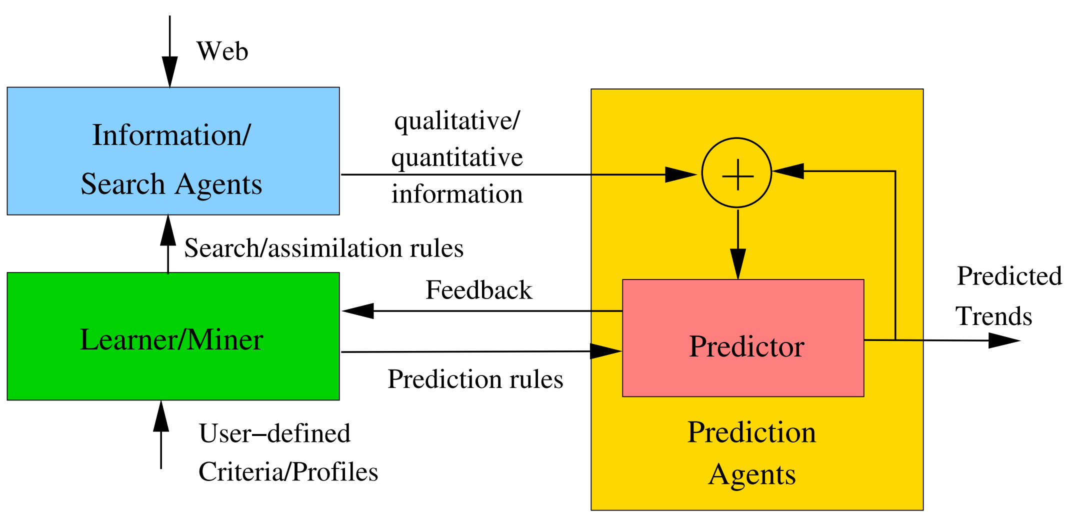
- Market trend prediction problem
  - Time series predictions
  - Metrics
- Signal processing of time series
  - Lags in predictable low-frequency components
- Data mining techniques
  - Intelligent mining and major design issues
  - Prediction agents
- Constrained optimizations using neural networks
  - Lagrange multipliers for discrete constrained optimization
- Some sample results

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

Ideal Model of Intelligent Mining for Trend Prediction



Major Design Issues

- Information/search agents to get information
  - Use of wrong, too many, or too little search criteria
    - Possibly inconsistent information from many sources
  - Semantic analysis of (meta-) information
  - Assimilation of information into inputs to predictor agents
- Learner/miner to modify information selection criteria
  - Apportioning of biases to feedback
  - Developing rules for Search Agents to collect information
  - Developing rules for Information Agents to assimilate information
- Predictor agents to predict trends
  - Incorporation of qualitative information
  - Multi-objective optimization not in closed form

Prediction Agents: Numerical Approaches

Memory-based approaches: data mining

- Using historical information to build a model of time-series behavior in order to predict future behavior
- KNN (K-Nearest Neighbor) classification techniques to locate points in multi-dimensional space
  - Too much noise in matching original time series
  - Difficulty in overcoming lags in low-frequency data

Computation-based approaches: neural networks and time-series analysis

- Formulation: Incorporation of quantitative and qualitative information
- Training algorithm
  - Window size, sampling lags, network topology, training parameters, training set, etc.

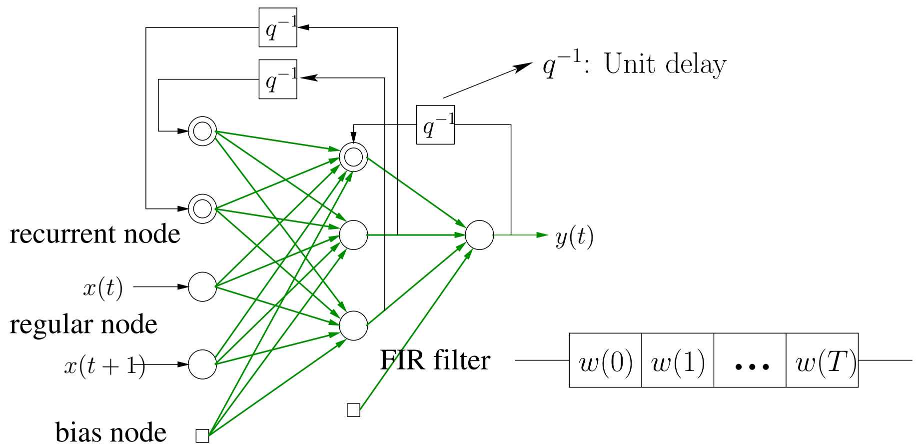
Outline

- Market trend prediction problem
  - Time series predictions
  - Metrics
- Signal processing of time series
  - Lags in predictable low-frequency components
- Data mining techniques
  - Intelligent mining and major design issues
  - Prediction agents
- Constrained optimizations using neural networks
  - Lagrange multipliers for discrete constrained optimization
- Some sample results

ANN Models for Time Series Predictions

- Existing architectures
  - Recurrent neural networks (RNN)
  - Memory-based neural networks (TDNN and FIR-NN)
  - Dynamic recurrent neural networks (DRNN): FIR + feedback without delay
  - No consensus on which architecture is better [Horne][Hallas]
  - Training algorithm is more important than architecture [Koskela]
- Proposed architecture: Recurrent FIR neural network (RFIR)
  - *RFIR*: FIR + recurrent feedback with time delay

(A) Proposed Recurrent FIR Architecture



Unit delay $\Rightarrow$ easier to derive gradients as compared with DRNN

Performance Metrics

- Normalized mean square error (nMSE):

$$\varepsilon = \frac{1}{\sigma^2 N} \sum_{t=t_0}^{t_1} (o(t) - d(t))^2,$$

  - σ^2 is the variance of the true time series in $[t_0, t_1]$
  - $o(t)$ is actual output at t; $d(t)$ is desired output
  - N is number of patterns in the measurement
- Open-loop single-step measurement: external input is true observed data
- Close-loop iterative measurement: external input is predicted output

Traditional Formulations for ANN Training

- Unconstrained formulation

$$\min_{w} \ E(w) = \frac{1}{n} \sum_{t=1}^{n} (o_t(w) - d_t)^2$$

- Training algorithms
  - BP/BP variants and gradient-based methods
  - Genetic algorithms
  - Simulated annealing
- Issues
  - No guidance when search reaches a non-zero local minimum of $E(w)$
  - Nonuniform errors across patterns – not good for training

(B) Proposed Constrained Formulations

- Each learning pattern is treated as an additional constraint:

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

  - τ decreases towards 0 as looser constraints are satisfied
  - Non-zero constraints provide guidance when search reaches a sub-optimum of the objective function
- New constraints added
  - Make the problem more difficult to solve
  - Do not lead to over-training of the neural network

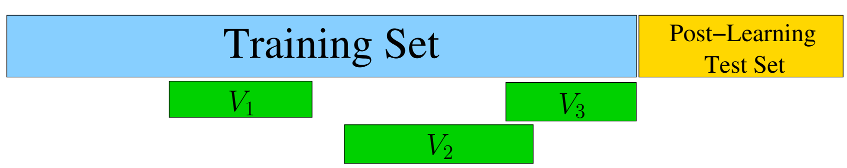
Traditional Cross-Validation

- Divide historical data into two *disjoint* sets
  - Training set
  - Cross-validation set

- Issues
  - Hard to choose appropriate validation set: how long?
  - Data used for cross-validation cannot be used for training
  - Only one validation set is used at any time: not good when time series is multi-stationary
  - Single-objective optimization minimizes errors in validation set: what about errors in learning?

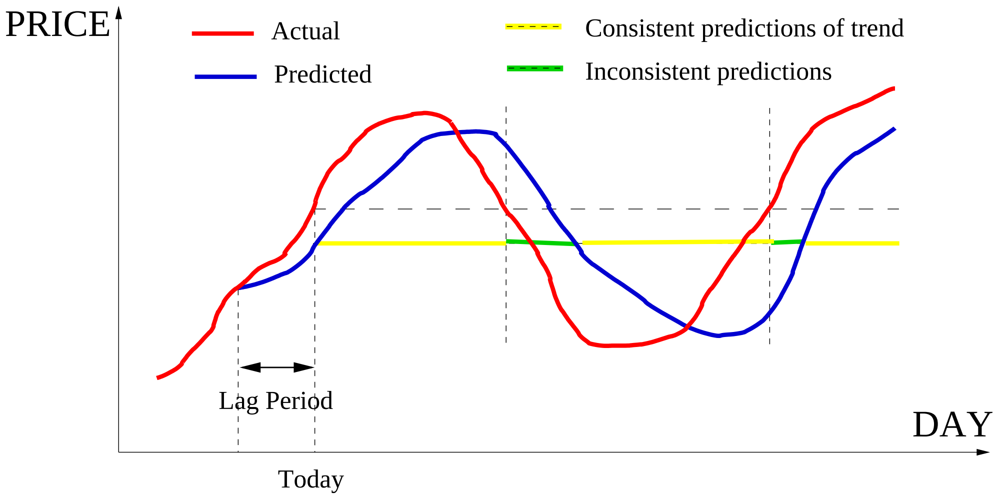
(C) Proposed Cross-Validation Method

- Multiple validation sets within training set



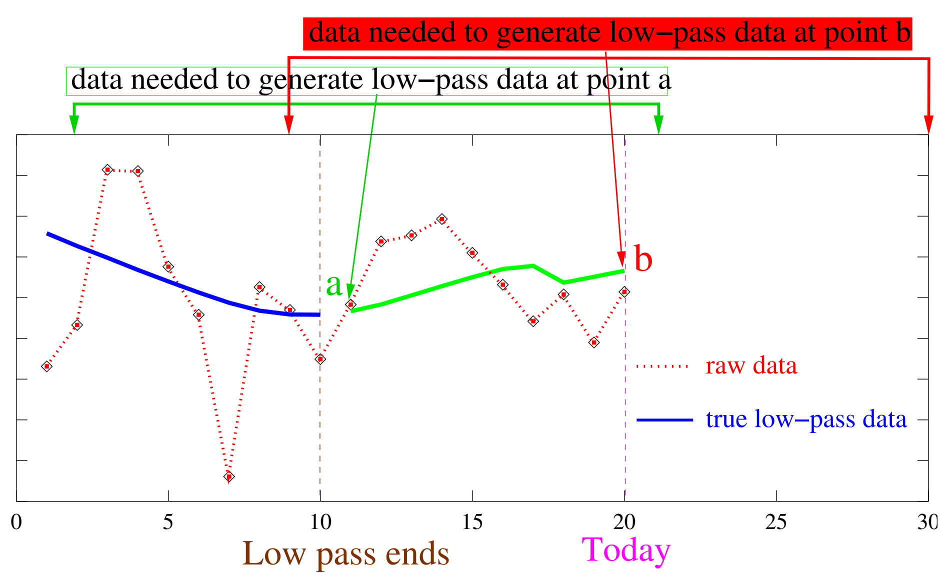
- Iterative and single-step validation errors are added as new constraints
  - Training patterns are fully used
  - Multiple regimes in a multi-stationary time-series are covered
  - Flexibility in choosing validation sets

(D) Penalties on Incorrect Trend Predictions



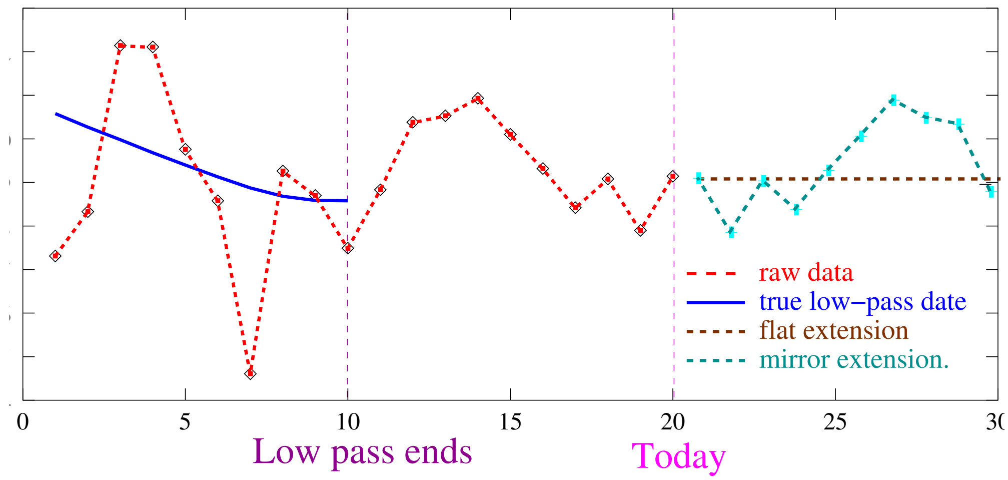
Patterns with inconsistent trend predictions are further penalized

(E) Predictions of Low-Pass Data in Lag Period



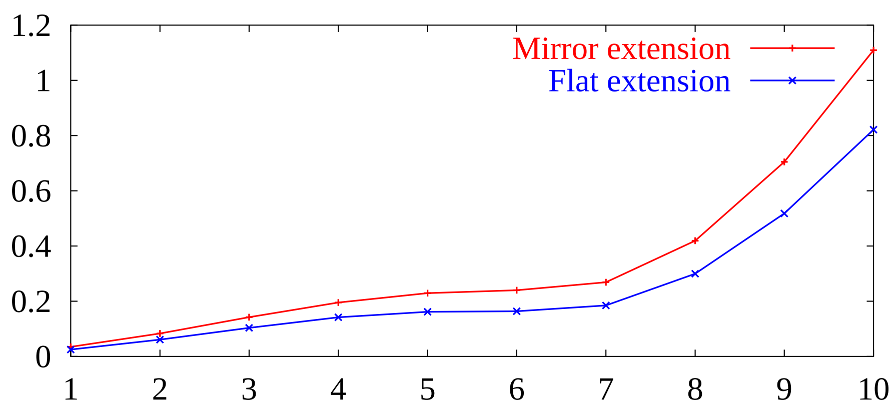
Previous Work on Handling Lags

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



Issues in Existing Methods for Lag Problem

- 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



(F) Constrained Formulation with Cross-Validation

- Constrained formulation without all closed-form functions

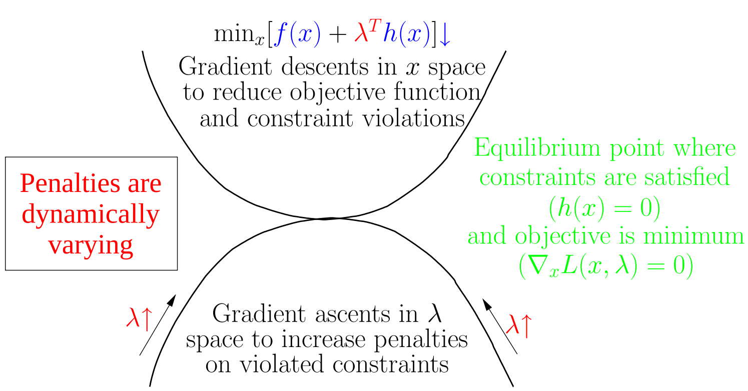
$$
\begin{array}{lll}
\min_w & E(w) = \frac{1}{n}\sum_{t=1}^{n} max\{(o_t(w) - d_t)^2 - \tau, 0\} & \\
\text{s.t.} & h_t(w) = (o_t(w) - d_t)^2 \le \tau, & \\
 & h_i^I(w) = \varepsilon_i^I \le \tau_i^I, & \text{(iterative validation)} \\
 & h_i^S(w) = \varepsilon_i^S \le \tau_i^S, & \text{(single-step validation)} \\
 & \sum Error_{lag} \le \tau_{lag} & \text{(sum of errors in lag period)}
\end{array}
$$

- Transformed into non-differentiable augmented Lagrangian function:

$$
\begin{aligned}
L(w,\lambda) = \; & E(w) + \sum_{t=1}^{n} \left(\lambda_t \max\{0, h_t - \tau\} + \tfrac{1}{2}\max^2\{0, h_t - \tau\}\right) \\
& + \sum_{k=1}^{v} \sum_{i=I,S} \left(\lambda_k^i \max\{0, \varepsilon_k^i - \tau_k^i\} + \tfrac{1}{2}\max^2\{0, \varepsilon_k^i - \tau_k^i\}\right) \\
& + \max\left(0, \sum Error_{lag} - \tau_{lag}\right)
\end{aligned}
$$

(G) Search for Saddle Points

- Constrained formulation solvable by Theory of Lagrange Multipliers for Nonlinear Discrete Constrained Optimization [Wah & Wu 1999]
- Discrete-neighborhood saddle point $\Longleftrightarrow$ constrained local minimum
  - Local minimum of $L(w,\lambda)$ in w subspace
  - Local maximum of $L(w,\lambda)$ in λ subspace



Violation-Guided Back Propagation (VGBP)

- Gradient descents in w subspace and stochastic acceptance of ascents
  - Using BP to generate approximate gradient direction in $L(w, \lambda)$
  - Accepting trial points with Metropolis probability using fixed temperature T

$$A_T(\mathbf{w}', \mathbf{w})|_\lambda = exp\left\{\frac{\min(0, L(\mathbf{w}) - L(\mathbf{w}'))}{T}\right\}$$

- Gradient assents in λ subspace by deterministic increases of λ
  - Large violation $\Rightarrow$ increased λ $\Rightarrow$ more penalty

Relax-and-Tighten Strategy

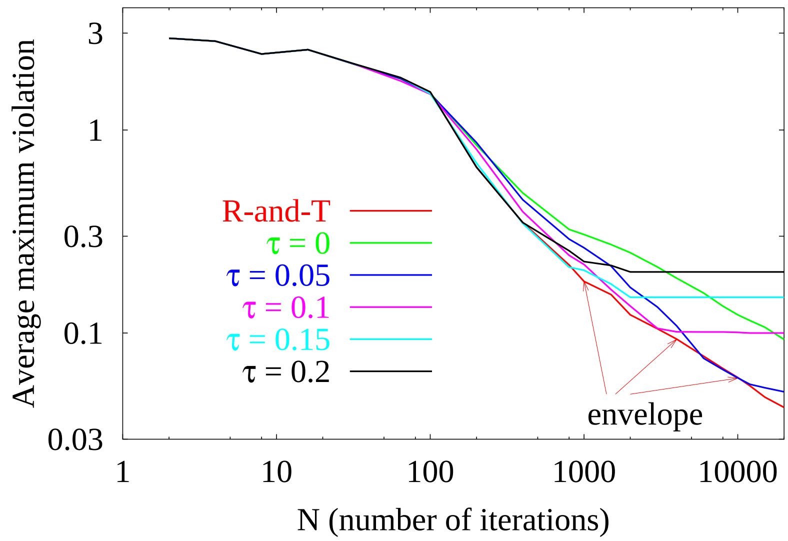
- Observations
  - Looser constraints

    $\Rightarrow$ Faster convergence and larger maximum violation at convergence
  - Tighter constraints

    $\Rightarrow$ Slower convergence and smaller maximum violation at convergence
- Relax-and-Tighten strategy
  - Loose constraints in the beginning and tighten gradually

    $\Rightarrow$ Faster convergence, and smaller maximum violation at convergence

Relax-and-Tighten Strategy



Outline

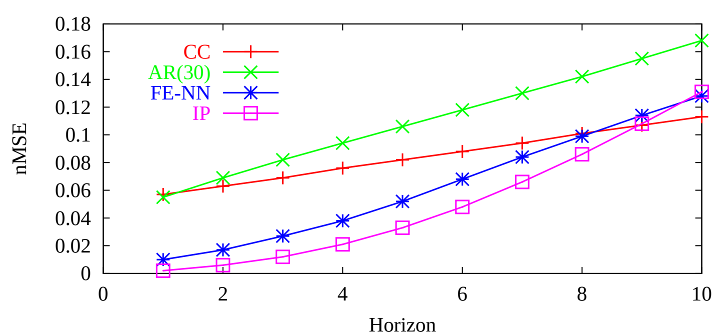
- Market trend prediction problem
  - Time series predictions
  - Metrics
- Signal processing of time series
  - Lags in predictable low-frequency components
- Data mining techniques
  - Intelligent mining and major design issues
  - Prediction agents
- Constrained optimizations using neural networks
  - Lagrange multipliers for discrete constrained optimization
- Some sample results

Experiments Setup

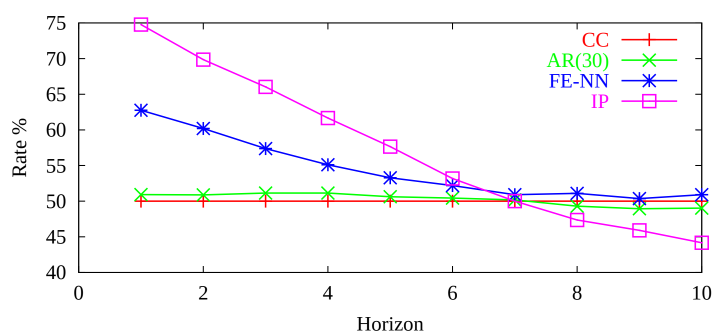
- Predictors compared
  - CC: carbon copy the most recently available data
  - AR: Auto-regression
  - FE-NN: Proposed neural network predictor
  - IP: Ideal predictor by using 7 true data in lag and trained by VGBP (approximate upper bound for predictions)
  - Results presented in most literatures have next-day hit rates below 55% [Gutjahr 97, Hellstrom 2000]
- Stocks
  - Citigroup (Symbol **C**), IBM (**IBM**), Exxon-Mobil (**XOM**)
  - Duration: 04/1997 to 03/2002

Predictions for Citigroup

- $nMSE$

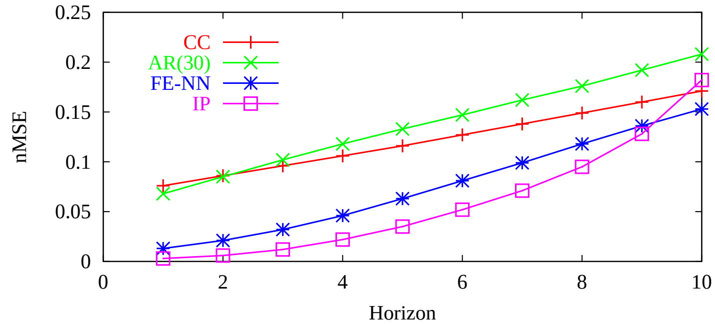


- Hit rate

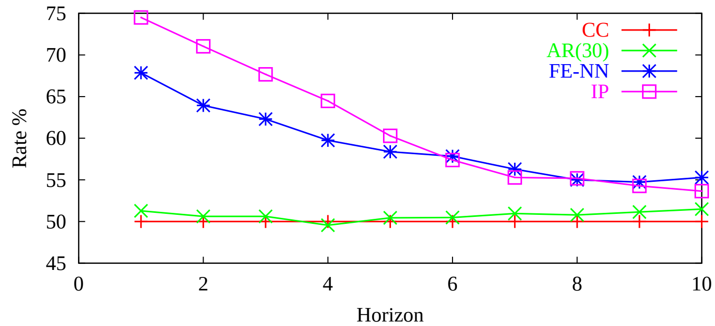


Predictions for IBM

- $nMSE$

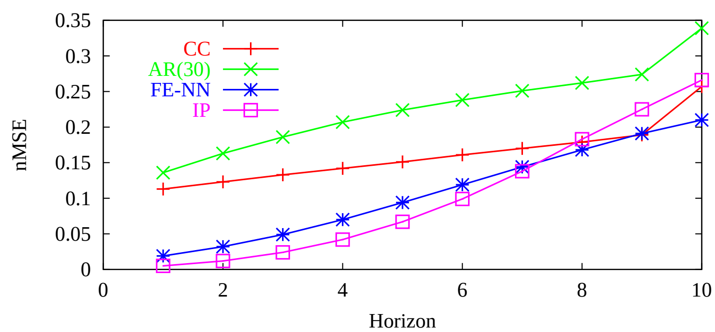


- Hit rate

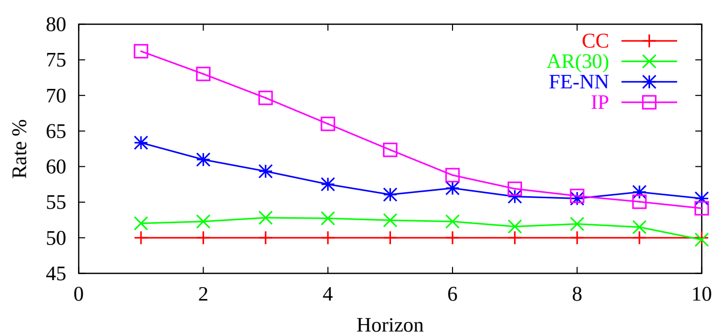


Predictions for Exxon-Mobil

- $nMSE$



- Hit rate



Conclusions

Signal processing is useful for

- Generating frequency components with shorter lags and better correlations
  - Low-frequency components have stronger long-term correlations but long lags
  - High-frequency components are not useful due to long lags and low correlations

Data mining is useful for

- Identifying information that can form new constraints or biases in learning
- Discovering promising input transformations in different frequency bands

Nonlinear constrained optimization is useful for

- Nonlinear predictions
- Multi-stage planning