Stock Price Predictions

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**

- **Linear Models**
  - ARMA & variants [Box 97]
  - state-space models [Aoki 87]

- **Nonlinear Models**
  - Pre-defined nonlinearity
    - TAR [Tong 90]
  - General nonlinearity (Machine learning)
    - Machine learning
      - Q-learning [Watkins 89]
      - kNN [Duda 73]
      - Reinforcement clustering [Jain 99]
    - Decision tree learning [Quinlan 86]

- **Time Series Models**
  - time-varying parameter models [Nicholls 85]

- **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
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

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)**

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

\(\sigma^2\): 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**:

\[
h^p_t(w) = (o_t(w) - d_t)^2 \leq \tau
\]

- \(\tau\): 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
  ![Diagram]

- 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
  \[
  h_{\text{lag}} = \sum_{t=t_0-m+1}^{t_0} \hat{S}(t) - R(t) \leq \tau_{\text{lag}},
  \]
  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

\[
\min_w E(w) = \frac{1}{n} \sum_{i=1}^{n} \max \{ (o_i(w) - d_i)^2 - \tau, 0 \}
\]

s.t. \( h_i(w) = (o_i(w) - d_i)^2 \leq \tau \),
\( h^e_i(w) = \tau^e_i \),
\( h^o_i(w) = \tau^o_i \),
\( h^{lag}(w) = \tau^{lag} \).

(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:

\[
L(w, \lambda) = E(w) + \sum_{i=1}^{n} \left( \lambda_i \max \{0, h_i - \tau\} + \frac{1}{2} \max \{0, h_i - \tau\}^2 \right) + \\
\sum_{i, j=x,t} \left( \lambda_{i,j} \max \{0, h_{i,j}^t - \tau^t_i\} + \frac{1}{2} \max \{0, h_{i,j}^t - \tau^t_i\}^2 \right) + \\
\lambda^{log}_{max} \max \{0, h^{log} - \tau^{log}\} + \frac{1}{2} \max \{0, h^{log} - \tau^{log}\}^2 \]

(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 \( \lambda \) 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

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

where $x^+ = \min\{0, x\}$ and $T$ is a fixed parameter (temperature).

- Gradient ascents in $\lambda$ subspace by deterministic increases of $\lambda$
  - Big violation $\Rightarrow$ increased $\lambda$ $\Rightarrow$ more contribution to gradient

- Relax-and-Tighten technique to speed up convergence [Wah & Qian]
  - Set initial $\tau$’s loose enough
  - Gradually tighten $\tau$’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
Stock Price Predictions

Predictions for Citigroup

- **nMSE**

- **Hit rate**

Predictions for IBM

- **nMSE**

- **Hit rate**
Stock Price Predictions

**Predictions for Exxon-Mobil**

- **nMSE**

![Graph showing nMSE vs Horizon for different models: CC, AR(30), FE-NN, IP.](image)

- **Hit rate**

![Graph showing Hit rate vs Horizon for different models: CC, AR(30), FE-NN, IP.](image)

Benjamin W. Wah and Minglun Qian

Stock Price Predictions

**Comments on Hit Rates**

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

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

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

$\Rightarrow$ 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]

Benjamin W. Wah and Minglun Qian
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