Factor Learning Portfolio Optimization

Presented by: Sinong Geng
Princeton University

October 18, 2022
Authors

Sinong Geng
Princeton University

Houssam Nassif
Amazon

Zhaobin Kuang
Google

A. Max Reppen
Boston University

Ronnie Sircar
Princeton University

Supported by Amazon AWS Credits.
Contents

Motivation

Methodology

Results
Motivation
Motivation

→ Portfolio Optimization: Learn a policy for wealth allocation in order to
  • maximize return,
  • minimize risk.

→ Stochastic Factors like economic indexes and proprietary trading signals:
  • Not controllable.
  • Evolve over time stochastically.
  • Affect asset prices.
Motivation

→ Machine Learning v.s. Continuous-Time Finance

Machine Learning:

- Flexible Representation
- Poor sample complexity and tend to overfit

Continuous-Time Finance:

- Small sample complexity.
- Rely on domain knowledge and thus may end up with over-simplified models.
Motivation

→ Combine machine learning with continuous-time finance

- **Challenges**
  - Huge Noise
  - Complicated Factor Effects
  - Discrete Time vs. Continuous Time

- **Our Solution**
  - Neural Stochastic Factor Models
  - Model-Regularized Policy Learning

- **Existing Works**
  - Reinforcement Learning
  - Continuous-Time Finance

FaLPO
Methodology

Neural Stochastic Factor Model
Model-Regularized Policy Learning
Methodology

→ Problem Formulation

• Assets: \( S_t := [S^1_t, S^2_t, \cdots S^{d_S}_t]^{\top} \) and a risk-free money market account with, for simplicity, zero interest rate of return;

• Features: \( Y_t \)

• Factors: From \( Y_t \), we can derive \( d_X \) factors denoted as \( X_t \) which
  • affect the dynamics of asset prices;
  • evolve over time stochastically;
  • are not affected by investment decisions.

• Policy: \( \pi_t \) as the fractions of wealth invested in the \( d_S \) assets at time point \( t \).

• Wealth: \( Z^\pi_t \).

• Performance Objective/Value Function:

\[
\max_{\pi} V(\pi) \text{ with } V(\pi) := \mathbb{E}[U(Z^\pi_T)].
\]
Neural Stochastic Factor Models

\[ \frac{dS^i_t}{S^i_t} = \int_S f^i_S(X_t; \theta^*_S) dt + \sum_{j=1}^{d_W} g^i_j S(X_t; \theta^*_S) dW^i_t, \quad i \in \{1, 2, \ldots, d_S\}, \]

\[ dX_t = f_X(X_t; \theta^*_S) dt + g_X(X_t; \theta^*_S)^T dW_t. \]

\rightarrow \text{Representation Function}

\[ X_t = \phi(Y_t; \theta^*_\phi) \]
Model-Regularized Policy Learning

→ Policy Functional Form

- Using tools in stochastic optimal control, we can derive the functional form of an optimal continuous-time policy:
  \[ \tilde{\pi}^* = \Pi(t, S_t, Z_t, X_t; \theta^*_\pi). \]
- Use the functional form in policy parameterization.
  \[ \pi(t, S_t, Z_t, Y_t; \theta_\phi, \theta_\pi) := \Pi(t, S_t, Z_t, \phi(Y_t; \theta_\phi); \theta_\pi). \]

→ Model Calibration

\[ \max_{(\theta_\phi, \theta_\pi, \theta_S) \in \mathcal{A}} H(\theta_\phi, \theta_\pi, \theta_S), \]
\[ H(\theta_\phi, \theta_\pi, \theta_S) := (1 - \lambda)V(\theta_\phi, \theta_\pi) + \lambda L(\theta_\phi, \theta_S). \]
Model-Regularized Policy Learning

→ Algorithm

**Algorithm** FaLPO

1: **Input:** number of iterations $N$.
2: Initialize $\theta_\phi$ and $\theta_\pi$.
3: for $n \in [N]$ do
4: Parameterize the policy function with $\Pi$.
5: Estimate the policy gradient for $H$.
6: Update $\theta_\phi$, $\theta_\pi$, and $\theta_S$.
7: end for
8: **Return** $\pi(\cdot; \theta_\phi, \theta_\pi)$
Example: Kim-Omberg Model

→ Neural Stochastic Factor Model

\[
\frac{dS^i_t}{S^i_t} = X^i_t \, dt + \sum_{j=1}^{d_W} \sigma^{ij} \, dW^j_t,
\]

\[
dX_t = \mu(\omega - X_t) \, dt + \nu \, dW_t,
\]

and \(X_t = \phi(Y_t; \theta^*_\phi)\).

→ Model-Regularized Policy Learning

- **Policy Functional Form:** For power utility \(\Pi(t, S_t, Z_t, \phi(Y_t; \theta^*_\phi); \theta^*_\pi) = k_1(t; \theta^*_\pi)\phi(Y_t; \theta^*_\phi) + k_2(t; \theta^*_\pi);\) for exponential utility \(\Pi(t, S_t, Z_t, \phi(Y_t; \theta^*_\phi); \theta^*_\pi) = k_1(t; \theta^*_\pi)\phi(Y_t; \theta^*_\phi)/Z_t + k_2(t; \theta^*_\pi)/Z_t\).

- **Model Calibration:**

\[
L(\theta^*_\phi, \theta^*_S) := -\mathbb{E} \left[ \sum_{i=1}^{d_S} \left[ \log(S^i_{t+\Delta t}) - \log(S^i_t) - \phi^i(Y_t; \theta^*_\phi)\Delta t - \theta^*_S \right]^2 \right]
\]
Results
Theory

→ Setup

**Algorithm** Projected FaLPO

1: **Input**: Number of iterations $N$ and a ball $B$.
2: **Output**: $\theta_\phi$, $\theta_\pi$, and $\theta_S$
3: **for** $n \in [N]$ **do**
4: Parameterize the policy function by $\Pi$.
5: Estimate the gradients of $H$.
6: Update $\theta_S$ and $\theta_R$ with learning rate $\eta$ by gradients.
7: Project the achieved update to $B$.
8: **end for**
9: **Return** $\theta_\phi$, $\theta_\pi$, and $\theta_S$.

→ In $B$, we pose assumptions.
With the aforementioned projection-based FaLPO algorithm and assumptions, there exist positive constants $C_1, C_2, C_3,$ and $C_4$ such that

\[ E[V_{\Delta t}^* - V(\bar{\theta})] \leq \frac{e^{\Delta t}}{1 - \lambda} + \frac{H(\theta_{\Delta t}^*) - H(\theta^\dagger)}{1 - \lambda} + \frac{C_1 \log(N)}{N(1 - \lambda)} \]

\[ + \frac{C_1 \log(N)}{BN(1 - \lambda)} \left[(1 - \lambda)^2 C_2 + \lambda^2 C_3 + 2\lambda(1 - \lambda)C_4\right], \]

where $\lambda \in [0, 1]$. Also, $e_{\Delta t}$ is an error term not related to $N$ or $B$ but dependent on $\Delta t$ with $\lim_{\Delta t \to 0} e_{\Delta t} = 0$. 
## Experiments

→ **Synthetic:**

<table>
<thead>
<tr>
<th>Annual Volatility</th>
<th>0.1</th>
<th>0.2</th>
<th>0.3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FaLPO</strong></td>
<td>$-0.465 \pm 0.446$</td>
<td>$-1.35 \pm 0.155$</td>
<td>$-2.737 \pm 0.219$</td>
</tr>
<tr>
<td><strong>DDPG</strong></td>
<td>$-1.650 \pm 0.456$</td>
<td>$-3.30 \pm 1.294$</td>
<td>$-5.495 \pm 1.269$</td>
</tr>
<tr>
<td><strong>SLAC</strong></td>
<td>$-0.750 \pm 0.210$</td>
<td>$-5.50 \pm 0.011$</td>
<td>$-6.160 \pm 0.012$</td>
</tr>
<tr>
<td><strong>RichID</strong></td>
<td>$-3.350 \pm 0.111$</td>
<td>$-5.65 \pm 0.102$</td>
<td>$-6.325 \pm 0.048$</td>
</tr>
<tr>
<td><strong>CT-MB-RL</strong></td>
<td>$-2.850 \pm 0.014$</td>
<td>$-5.35 \pm 0.020$</td>
<td>$-6.160 \pm 0.026$</td>
</tr>
<tr>
<td><strong>MMMC</strong></td>
<td>$-4.723 \pm 7.619$</td>
<td>$-5.602 \pm 4.299$</td>
<td>$-6.124 \pm 3.217$</td>
</tr>
</tbody>
</table>

**Table:** Average terminal utility after tuning with standard deviation for synthetic data
Experiments

→ Real-world portfolio optimization:

<table>
<thead>
<tr>
<th>Methods</th>
<th>Energy</th>
<th>Material</th>
<th>Industrials</th>
<th>Mix</th>
</tr>
</thead>
<tbody>
<tr>
<td>FalPO</td>
<td>$-2.4 \pm 1.9$</td>
<td>$-3.2 \pm 1.0$</td>
<td>$-6.3 \pm 2.3$</td>
<td>$-3.5 \pm 1.5$</td>
</tr>
<tr>
<td>DDPG</td>
<td>$-6.6 \pm 1.2$</td>
<td>$-7.3 \pm 1.5$</td>
<td>$-7.3 \pm 2.1$</td>
<td>$-2.5 \times 10^4 \pm 3.3 \times 10^8$</td>
</tr>
<tr>
<td>SLAC</td>
<td>$-6.8 \pm 0.2$</td>
<td>$-7.0 \pm 1.5$</td>
<td>$-342.4 \pm 886.8$</td>
<td>$-3.0 \times 10^8 \pm 4.3 \times 10^{12}$</td>
</tr>
<tr>
<td>RichID</td>
<td>$-6.5 \pm 0.1$</td>
<td>$-6.9 \pm 1.4$</td>
<td>$-6.9 \pm 0.4$</td>
<td>$-8.1 \pm 3.9$</td>
</tr>
<tr>
<td>CT-MB-RL</td>
<td>$-4.2 \pm 6.2$</td>
<td>$-5.4 \pm 4.3$</td>
<td>$-11655 \pm 32947.5$</td>
<td>$-5.7 \pm 3.1$</td>
</tr>
<tr>
<td>MMC</td>
<td>$-8.5 \pm 7.6$</td>
<td>$-6.5 \pm 1.7$</td>
<td>$-11.0 \pm 5.4$</td>
<td>$-7.5 \pm 4.4$</td>
</tr>
</tbody>
</table>

Table: Average terminal utility for real-world data. Mix denotes a mix of stocks in the previous three sectors.
Thank you!
Appendix

→ Competing Methods:

<table>
<thead>
<tr>
<th>Methods</th>
<th>Explicit Factor Representation</th>
<th>Continuous-Time Model</th>
<th>Discrete-Time Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMMC</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>DDPG</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>SLAC</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>RichID</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>CT-MB-RL</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>FaLPO</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
</tr>
</tbody>
</table>

Table: Competing methods and their characteristics.
Appendix

→ More results:

Figure: Sensitivity analysis for $\lambda$