



AI × 金融 前沿问题

北京航空航天大学 计算机学院



报告内容

- 深度学习在选股和择时中的应用已有广泛研究，面临创新瓶颈，有必要寻求新的应用领域和问题。
- RL在金融领域的前沿应用
 - 因子生成 (Factor Construction)
 - 最优执行 (Optimal Execution)
 - 做市 (Market Making)
 - 期权对冲 (Option Hedging)
- 重点侧重RL视角下具体问题和建模方法
- 假设大家对金融知识有基本的了解



目录

content

01

因子生成

02

期权对冲

03

做市

04

最优执行



因子生成

使用强化学习挖掘股票因子

Generating Synergistic Formulaic Alpha Collections via Reinforcement Learning, KDD, 2023

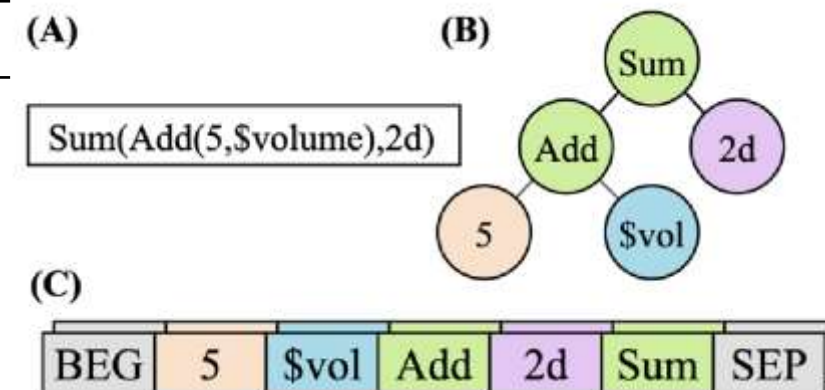
- **Motivation:** 量化交易领域仍大量使用Factor Model，由人工挖掘因子；传统挖因子方法没有考虑因子的协作性能。
- **Problem:** 从历史数据出发，生成Formulaic Alpha
- **Prior Work:** 使用遗传算法生成
- **Key Issue:** 1.如何搜索因子表达式；2.如何考虑因子的协作性能。



Generating Synergistic Formulaic Alpha Collections via Reinforcement Learning, KDD, 2023

- Part 1: alpha generation
- 因子表示：逆波兰表达式

Category	Token Examples
Operators	<i>CS-Log, CS-Add, TS-Mean, ...</i>
Features	<i>\$open, \$volume, ...</i>
Constants	$\pm 30, \pm 10, \pm 5, \pm 2, \pm 1, \pm 0.5, \pm 0.01$
Time Deltas	<i>10d, 20d, 30d, 40d, 50d</i>
Sequence Indicator	<i>BEG(begin), SEP(end of expr)</i>

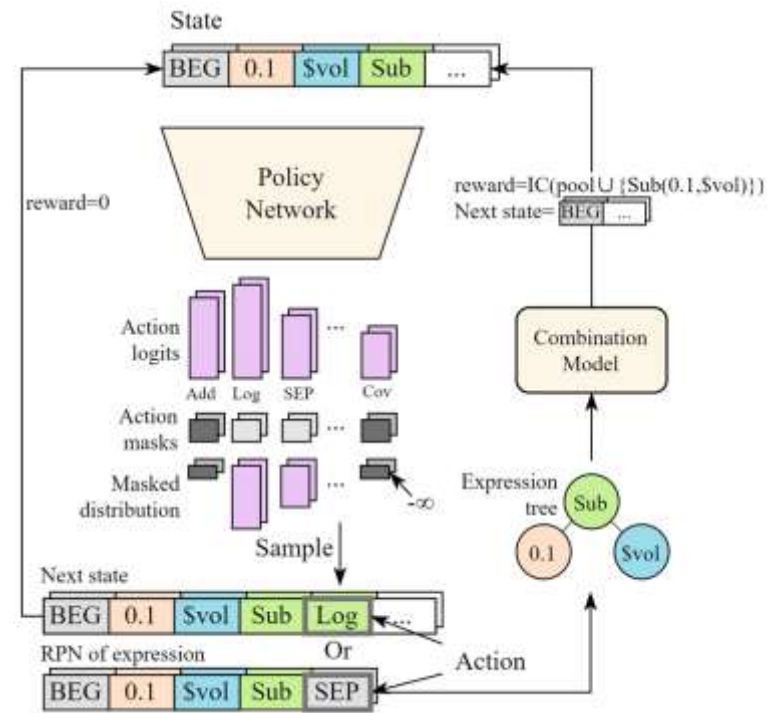


- CS: Cross Section, TS: Time Series
- 建模为非平稳马尔可夫决策过程



Generating Synergistic Formulaic Alpha Collections via Reinforcement Learning, KDD, 2023

- **State Space:** 已经生成的token序列，用LSTM提取特征
- **Action Space:** 可选Token，采用一些规则避免非法token
- **Reward:** r_{end} =Part 2中多个因子协同的效果， $r_t = 0$
- 为了鼓励更长的因子， $\gamma = 1$
- 使用PPO优化



Generating Synergistic Formulaic Alpha Collections via Reinforcement Learning, KDD, 2023

- **Part 2: Alpha Combination Model**

- 用于计算多个因子 f_i 组合的效果

- **Reward:** $r = \bar{\sigma}_y(\sum_{i=1}^k w_i f_i)$

- $\bar{\sigma}_y(f) = \mathbb{E}_t[\text{corr}(f(X), y)]$

- $w = \text{argmin}_w \text{MSE}(\sum_{i=1}^k w_i f_i(X), y)$

- \mathcal{F}, y 在计算前经过归一化

- 更新因子集: $\mathcal{F} = \mathcal{F} \setminus \{f_p\}, p = \text{argmin}_i |w_i|$



Generating Synergistic Formulaic Alpha Collections via Reinforcement Learning, KDD, 2023

- **Experiments**

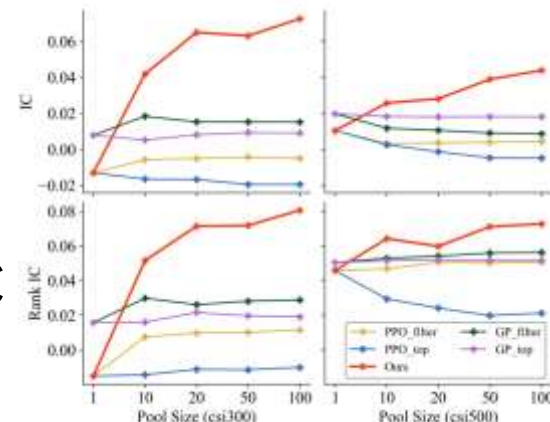
- **Res 1: 因子值和收益的IC与Rank IC**

- baselines:
 - MLP XGBoost LightGBM
 - PPO(single alpha) 遗传算法——top-k filter-IC

- **Res 2: 不同因子数量的情况**

- **Res 3: 一个10个因子的例子, f_i, w_i, IC**

- **Res 4: 回测Investment Simulation**

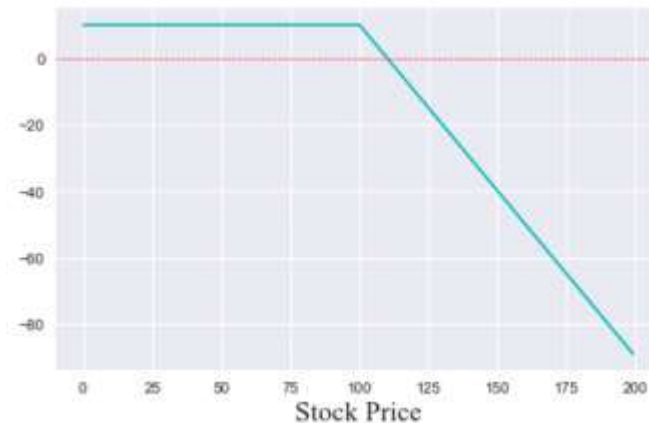


期权对冲

金融来源、概念、方法

期权对冲

- 期权 (option)
- 卖出欧式看涨期权：对方支付权利金，享有在特定行权时间向我方按条约价格买入资产的权力（也可以不买）
- 行权时刻资产价格高于条约价格：亏损对应差价减权利金
- 行权时刻资产价格低于条约价格：对方放弃行权，赚权利金



条约价格为100时的收入



期权对冲

- **BSM模型**

- 当市场满足：

- 价格服从几何布朗运动
- 波动率恒定
- 无摩擦（无交易成本、滑点）
- 无套利机会（市场完全有效）

- 存在一个动态投资组合 Π ，由基础资产(underlying asset)和无风险资产组成，并满足：

- $V[\text{期权} - \text{复制组合}\Pi] = 0$

- Example:

- 当资产上涨时，买入更多资产抵消期权的亏损
- 当资产下跌时，买入更多无风险资产减少 Π 的亏损



期权对冲

- 卖出期权并持有复制组合的情况下，风险（方差）为0，但会固定亏损一个值，该值即为该期权的定价（权利金）。
 - 当市场符合BSM模型假设时，期权是多余的金融衍生品
- 在实际的市场中，假设中的许多条件不满足，需要新的方法进行期权对冲，若权利金固定，在风险更低的情况下，我们希望 $E[\Pi - \text{期权}]$ 尽量大，使得收入尽量高。
- 可以看出该问题与单资产交易问题很接近，但有以下区别：
 - 期权有行权时间，策略需要考虑这一点（内在价值和实际价值）
 - 期权自身一般有价格，可以纳入考虑甚至交易期权，关于期权价格的分析理论可以作为参考
 - 优化目标和评价标准不同



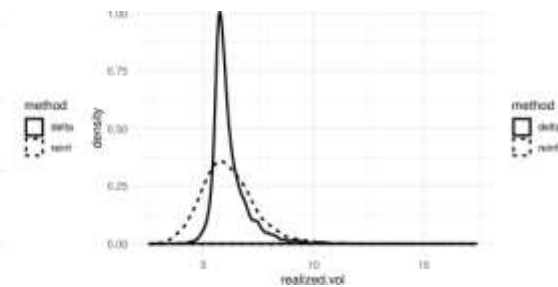
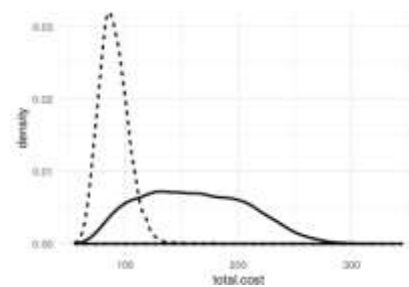
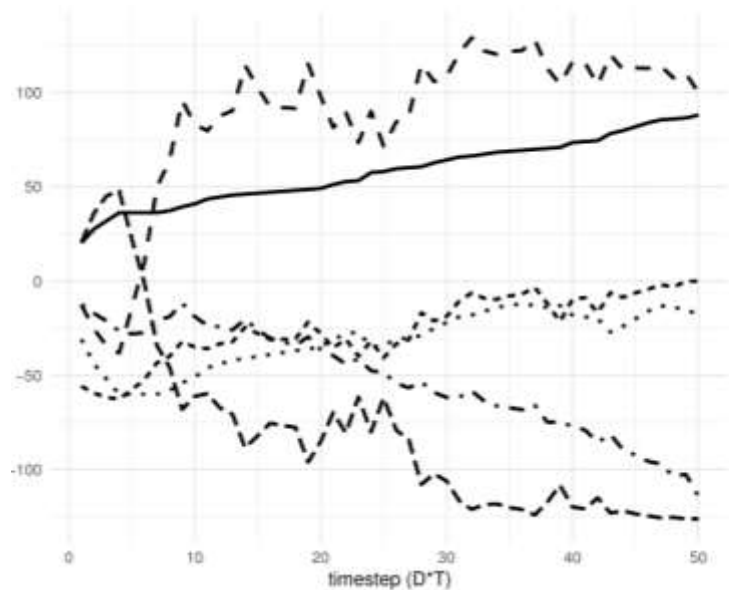
Dynamic Replication And Hedging: A Reinforcement Learning Approach, The Journal of Financial Data Science, 2019

- **Action:** 交易基础资产
- **Space:** $\mathcal{S} = (P, \tau, n)$, 即价格、到期时间、目前资产持仓
- **RL算法:** SARSA, ϵ -greedy
- 模型可以非常简单 (线性回归、随机森林、支持向量机)
- **目标函数:** $\max(\mathbb{E}[w_T] - \frac{\kappa}{2} \mathbb{V}[w_T])$
 - 假设 $\text{cov}(\delta w_{t_1}, \delta w_{t_2}) = 0, \delta w_{t_1} \neq \delta w_{t_2}$
 - $\min_{\pi} \sum_{t=0}^T (\mathbb{E}[-\delta w_t] + \frac{\kappa}{2} \mathbb{V}[\delta w_t])$
- **市场摩擦:** $cost(n) = multiplier \times Spread \times (|n| + 0.01n^2)$



Dynamic Replication And Hedging: A Reinforcement Learning Approach, The Journal of Financial Data Science, 2019

- **Experiments:**
- **Res 1:** total PnL随时间变化图 (RL、baseline)
- **Res 2:** 实验中total cost、波动性、total PnL的分布图



做市

做市商连续挂买卖订单为市场提供流动性

IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- **做市:**
- 做市商不断在限价订单簿上同时挂买入和卖出订单，以此为市场提供流动性，获取收益和来自交易所的补偿
- **Motivation:**
- 挂单需要参考市场中间价，但市场价格不稳定
- 现有方法往往只在单价格级别挂单，现实中在多个档位挂单可以获得更好的队列位置，但这会使建模变复杂
- 在此情况下，RL动作空间大难以学习



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- 稳定参考价:

- $\tilde{p}_{ref,t} = m_t = \frac{ask_t + bid_t}{2}$ 取整

- 当 m_t 变化时, 只有当对应位置为空时才更新 p_{ref} , 避免了发生交易但LOB不变时参考价改变

- State:

- $s_t = (s_t^m, s_t^s, s_t^p)$,

- s^m 市场变量, s^s 信号变量, $s^p = (s^q, s^v)$ 私有变量

- $s^q = (q^{-K}, \dots, q^{-1}, q^1, \dots, q^K)$, $s^v = (v^{-K}, \dots, v^{-1}, v^1, \dots, v^K)$

- $q^i = \sum_{j=1}^{l_i} \frac{pos_{i,j}}{l_i} \frac{v_{i,j}}{l_i}$ 即加权位置和, v^i 为数量



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- **Action:**

- $a_t = (m_t^*, \delta_t^*, \phi_t^{bid}, \phi_t^{ask})$

- ϕ_t^{bid} 为从 $m_t^* - \frac{\delta_t^*}{2}$ 开始的买入分布, ϕ_t^{ask} 同理

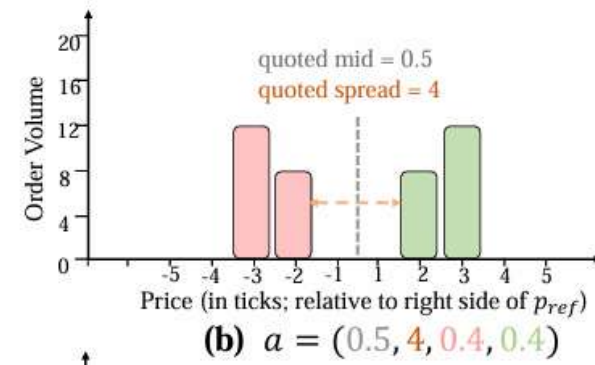
- **Reward:**

- **PnL:** $PnL_t = (\sum_{i \in A_t} p_i^{ask} v_i^{ask} - \sum_{i \in B_t} p_i^{bid} v_i^{bid}) + (p_{t+1} - p_t)z_{t+1}$

- **IP:** $IP_t = -\eta |z_t| [|z_t| > Const]$ 持有资产的惩罚

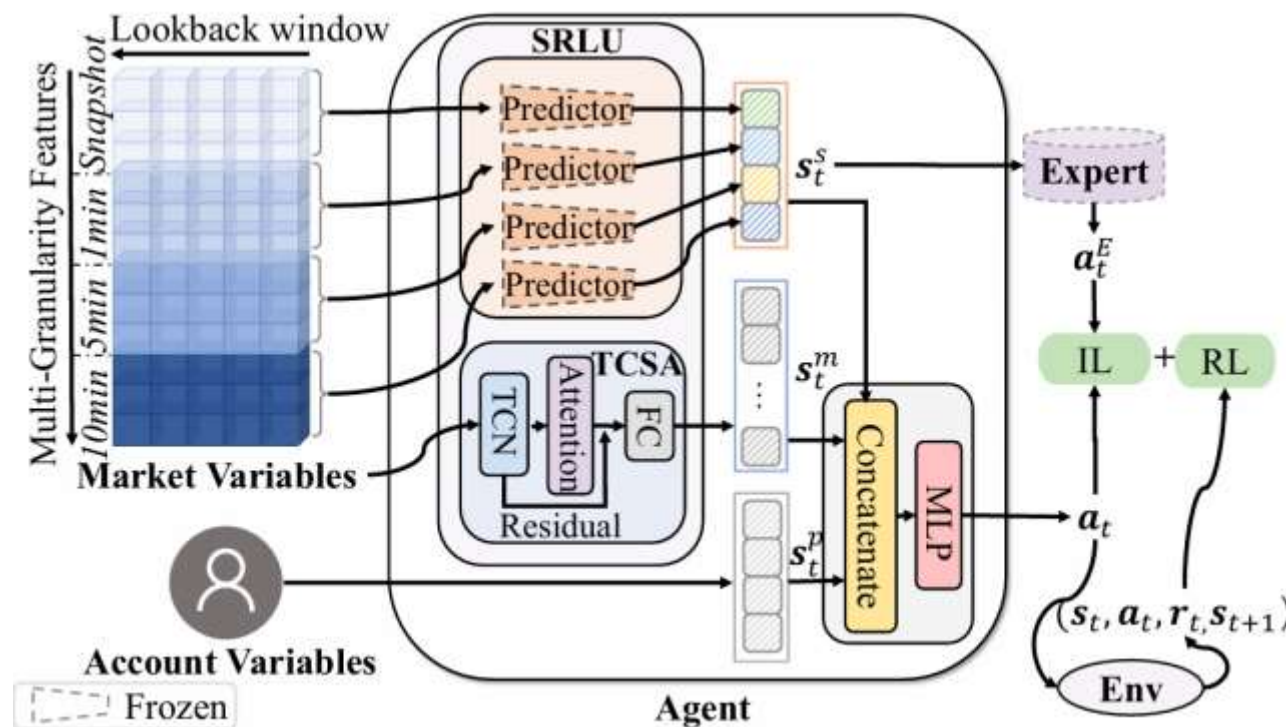
- **C:** $C_t = \beta (\sum_{i \in A_t} p_i^{ask} v_i^{ask} + \sum_{i \in B_t} p_i^{bid} v_i^{bid})$ 尽可能多交易以获取交易所补偿

- $R = PnL + IP + C$



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- s^s 由LightGBM预测变化趋势，模型采用监督预训练
- TCSA: TCN+Attention，获取市场状态



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- 模仿学习
- 构造了一个Expert, 当 $-d \leq z_t \leq d$ 时
 - $$\begin{cases} ask_t = m_t + a + b \cdot z_t + c \cdot \hat{y}_t, & \text{if } z_t \leq d \\ bid_t = m_t - a + b \cdot z_t + c \cdot \hat{y}_t, & \text{if } z_t \geq -d \end{cases}$$
 - a, b, c, d 预先设定, $\hat{y}_t \in \{-1, 0, 1\}$ 为预测的变化趋势
- RL算法: TD3
- $$\pi = \operatorname{argmax}_{\pi} \mathbb{E}_{(s,a) \sim D} [q(s, \pi(s))] - \lambda \mathbb{E}_{(s,\hat{a}) \sim D_E} [(\pi(s) - \hat{a})^2]$$



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- **Experiments:**
- **Benchmarks:**
 - FOIC, LIIC, LTIIC
 - RL_{DS} : 单价格水平
 - DRL_{OS} : 多价格水平, 考虑每一单位资产在每个价格的交易
- **Metrics:**
 - PnL
 - MAP: 平均绝对仓位 $MAP = \frac{\sum_t |z_t|}{\sum_t [|z_t| > 0]}$
 - RPT: 每交易收益 $RPT = \left(\frac{\sum_{i \in A} p_i^{ask} n_i^{ask}}{\sum_{i \in A} n_i^{ask}} - \frac{\sum_{i \in B} p_i^{bid} n_i^{bid}}{\sum_{i \in B} n_i^{bid}} \right) / \overline{\delta^m}$
 - PnL/MAP



IMM: An Imitative Reinforcement Learning Approach with Predictive Representation Learning for Automatic Market Making, IJCAI, 2024

- **Dataset:** 上海期货交易所RB,FU,CU,AG
- **Res 1:** 与benchmark比较
- **Res 2:** 消融实验, 删去LOB状态 s^q 、 s^s 、TCSA、RL、IL
- **Res 3:** 不同Reward的实验
- **Res 4:** 构造了反转比率 adv_ratio 验证 s^s 在避免Adverse Selection方面的效果, 通过测试LOB队列中订单数量验证了 s^q 在避免撤单和保持队列位置方面的效果



最优执行

将大订单分拆成多个小订单或LOB

最优执行

- 在执行很大的订单时可能会出现巨大的滑点，一般将其拆分为多个小订单依次执行。
- 订单
 - 市价单：直接与对手方以市场价成交，产生市场冲击
市场冲击与价格风险的权衡
 - 限价单：以指定价格在订单簿上下单，等待条件满足触发交易
未执行风险与价格风险的权衡

Bid		Ask	
Qty	Price	Price	Qty
60	8.9	9.0	300
20	8.8	9.1	180

LOB



最优执行

- 问题建模
- 对于市价单，需要估计前面的订单对后面订单的冲击
 - 例: Almgren–Chriss模型：
 - $$\mathbb{E}[\Delta] = \sum_{t=1}^T v_t (\sum_{t'=1}^{t-1} G(t-t')v_{t'}) + \sum_{t=1}^T \frac{v_t S_t}{2}, G(t) = G_0 e^{-\omega t}$$
 - Temporary Impact: 只对当前订单生效
 - Permanent Impact: 影响之后的价格
- 对于限价单，需要模拟LOB的变化
 - 影响订单簿，从而影响交易者的行为，较为复杂
 - 常见做法是当作小的交易对LOB没有影响



最优执行

- **衡量指标**

- PnL: 整体收益
- Implementation Shortfall: 与全部立即执行相比的提升
- Sharpe Ratio

- **基准策略**

- TWAP: 在时间上均匀执行
- VWAP: 在交易量上均匀执行, 需要对交易量进行估计
- SnL: 挂一个固定的卖单, 最后用市价单全部卖出
- Almgren-Chriss: 随机控制模型



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- **Motivation:**

- **1:** 以往的RL方法一般只能处理几分钟的交易，而现实中可能会使用若干小时甚至一天来完成一个大单的交易。
- **2:** 以往同时确定限价单价格和大小的方法动作空间过大。
- **3:** 将最优执行建模为两层MDP，适合使用Hierarchical RL方法解决。
- **4:** RL需要长时、准确的市场表示，借鉴最近无监督辅助任务的研究。



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- **Part 1: 2-level MDP**

- **High Level:**

- **State:** $s_i = (m_i, p_i)$, $m_i: [i\Delta t - L\tau, i\Delta t]$ 市场表示, $p_i = (\frac{rem_i}{Q}, 1 - \frac{i\Delta t}{T})$
- **Action:** $a_i \in [0, \min(c, \frac{rem_i}{Q})]$, c 最大交易比例
- **Reward:** $r_i = \sum_{j \in \mathcal{O}_i} \frac{q_j}{Q} (\frac{p_j - \tilde{p}}{\tilde{p}})$, 即按订单大小加权的相对VWAP的优势

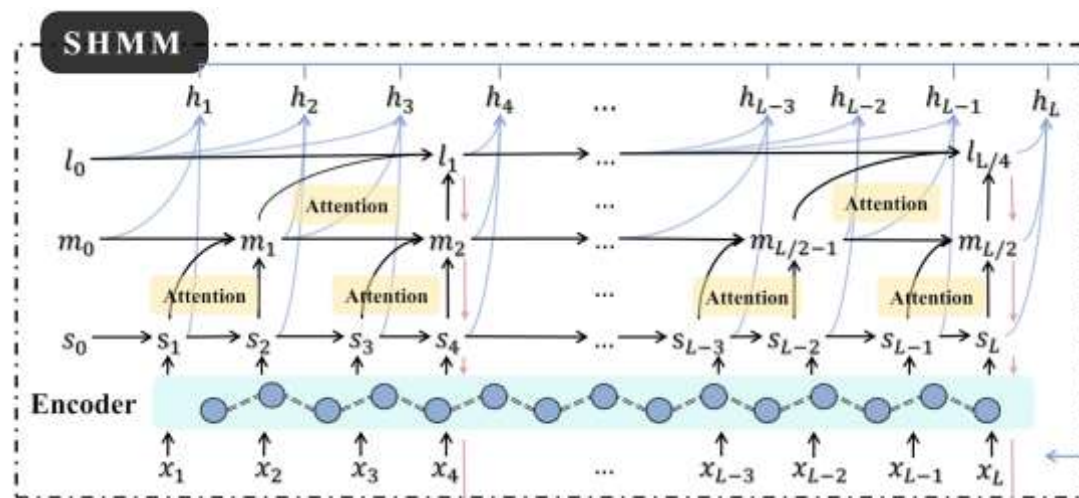
- **Low Level:**

- **State:** $s_t = \left(m_t, \left(\frac{rem_t}{a_i}, \frac{t-i\Delta t}{\Delta t}, \frac{a_i}{b_i} \right) \right)$, b_i : 上一段时间市场上平均订单大小
- **Action:** $a_t = [-n_p, n_p]$, 即LOB价格范围
- **Reward:** $r_t = \frac{b_i}{a_i} \left(\frac{p_t - \tilde{p}_i}{\tilde{p}_i} \right)$, p_t 实际交易价格 $p_t \geq a_t$



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- Part 2: SHMM(stacking hidden Markov model)
- $i_j(s_j, m_k, l_j)$ 为不同时间粒度的表示
- 使用多层多粒度 [sequential VAE](#) 编码市场历史数据, 先验概率用GRU+Attention建模, 优化ELBO
- 最后将 (s_t, m_j, l_i) 合在一起作为结果



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- **Part 3: policy训练**
- **模型**: 时序Attention+Linear
- **High Level**: TD3
 - 模仿学习: 构造一个expert
 - $\pi = \operatorname{argmax}_{\pi} \mathbb{E}_{(s,a) \sim D} [(s, \pi(s))] - \lambda \mathbb{E}_{(s,\hat{a}) \sim D_E} [(\pi(s) - \hat{a})^2]$
- **Low Level**: Dueling Q-Network
- **训练策略**: 先预训练Low Level, 再训练基于TWAP的High Level, 最后一起训练



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- **Experiments**
- **Metrics:**
 - **Price Advantage(PA):** 相对于VWAP的excess return
 - **Win Ratio:** 超过VWAP的天数占比
 - **Gain-Loss Ratio:** $\frac{\mathbb{E}(PA|PA>0)}{\mathbb{E}(PA|PA<0)}$
 - **Averaged Final Inventory:** 平均剩余 (任务未完成)
- **Benchmarks:**
 - **Traditional:** TWAP, VWAP, Almgren-Chriss
 - **RL-based:** DDQN, PPO, OPD, HALOP



MacMic: Executing Iceberg Orders via Hierarchical Reinforcement Learning, IJCAI, 2024

- **Res 1:** 和benchmark在CSI100、NASDAQ100的对比
- **Res 2:** 消融实验，删去High Level、Low Level、SHMM、IL
- **Res 3:** 在例子上的可视化效果



其他题目 (推荐资料)

- TRADES, QUOTES AND PRICES-Financial Markets Under the Microscope
 - LOB Dynamics(Arrival, Clustering, Spread), Price Impact
- ALGORITHMIC AND HIGH-FREQUENCY TRADING
 - Pairs Trading, Stochastic Optimal Control
- 知乎: [敲代码的quant](#)、[Niu Hui](#)、[张海抱](#)





谢谢大家