多因子选股模型中数据时变规律研究

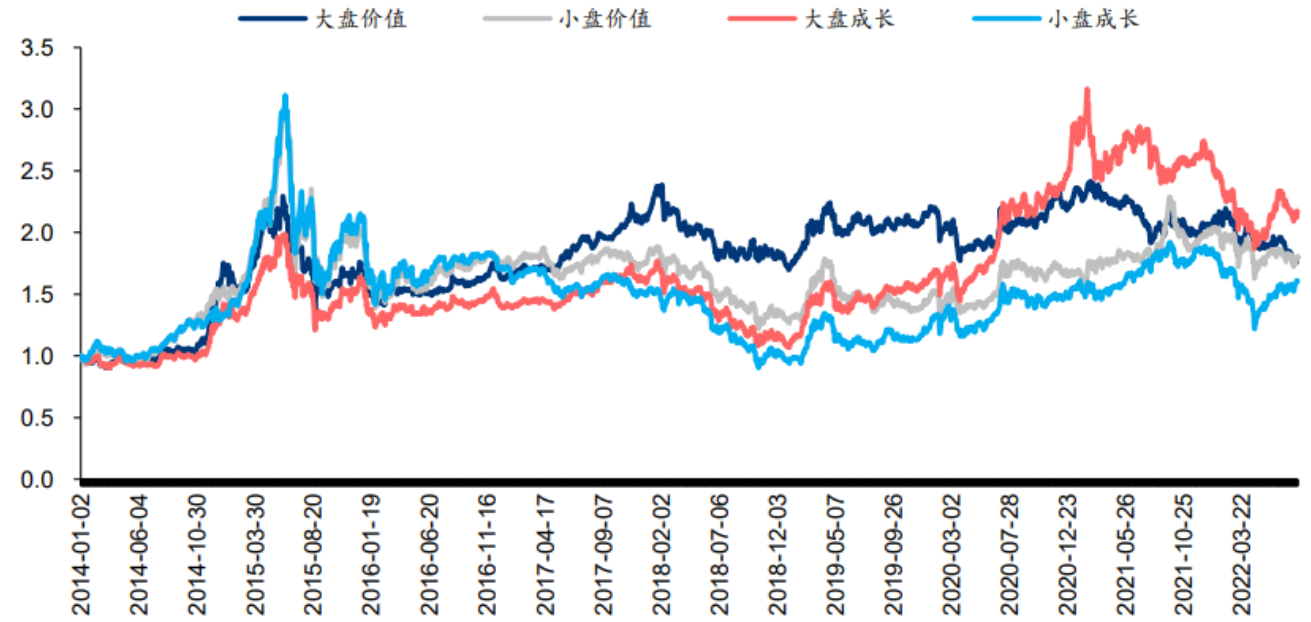
华泰研究 · 金融工程暑期实习生 商陈诚

2022年8月20日

股票市场中的数据时变现象

□ **股票市场中的数据时变现象**也指**市场风格**，而**成长**和**价值**是市场风格常见的分类方式。回溯过往8年股市风格表现情况，在不同时间段市场有不同表现，如2015—2016年期间，小盘股整体表现强于大盘股，而2016年末开始出现风格分化，大盘股变得更强势。

图表：2014 年-2022 年不同市场风格指数情况



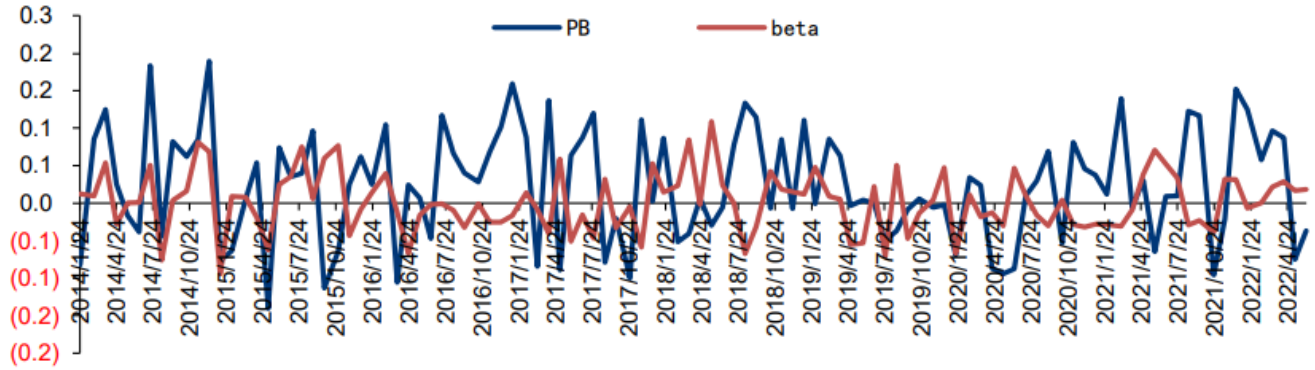
资料来源：Wind，华泰研究

数据时变现象对于多因子模型的影响

- 传统线性多因子模型研究框架下，**数据时变现象会导致样本内外因子截面收益率发生剧烈波动，从而导致多因子模型表现下滑，策略产生回撤。**

- 近年来机器学习被广泛运用到因子挖掘和合成中，但是**根据PAC学习理论，要保证机器学习模型在样本外的泛化性，数据分布必须满足IID的假设条件，**显然股票市场由于数据时变现象的存在，很难满足这个假设。

图表：2014 年-2022 年 PB 和 beta 因子在 A 股截面上收益率的波动情况



资料来源：Wind，华泰研究

目前对于数据时变现象的研究

- 学术界称时变现象为**Domain Shift，** 相关的研究为**Domain Generalization。**根据2022年微软亚研院发布最新综述Generalizing to Unseen Domains: A Survey on Domain Generalization. *IEEE Transactions on Knowledge and Data Engineering 2022,* 目前主要有数据操作、表征学习和学习策略三个方向。

- 本次汇报主要是研究并改进了微软亚研院在学习策略方向下最新的论文DDG-DA: Data Distribution Generation for Predictable Concept Drift Adaptation. *AAAI2022*。论文创造性地提出学习元模型来对样本进行时间序列上赋权，从而提高样本外的泛化能力。

图表：Domain Generalization 的主要研究方向和典型方法

资料来源：华泰研究

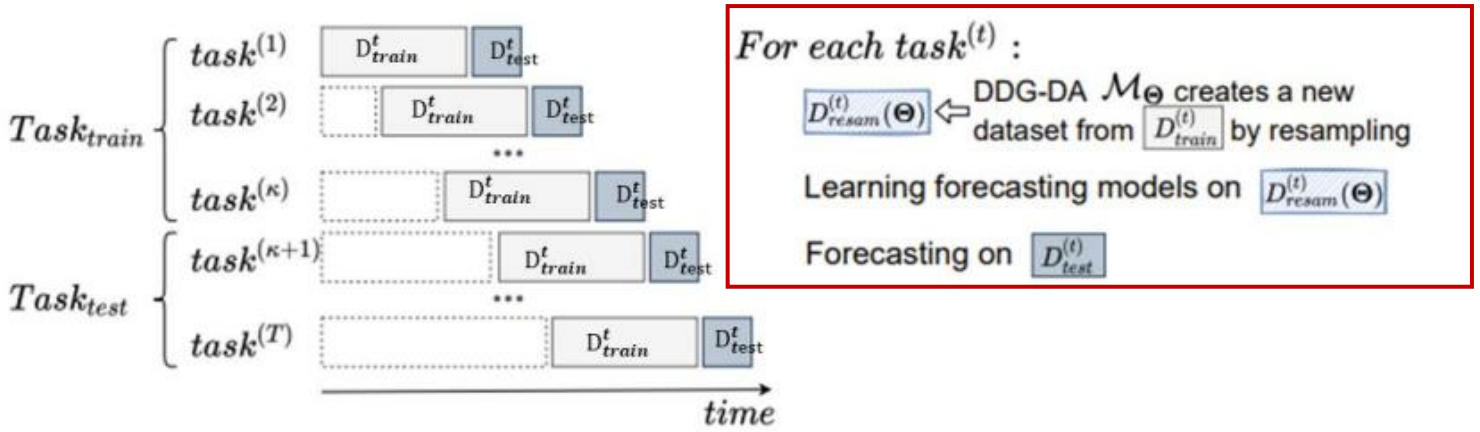
<华泰研究>

DDG-DA的核心思想：学习一个给不同时间点样本赋权的元模型 M_θ

- **典型的量化建模场景：**在t时刻我们要在训练集$D^t_{train}(x,y)\sim P^t_{train}(y|x)$构建量化模型，然后对测试集$D^t_{test}(x,y)\sim P^t_{test}(y|x)$进行预测。**由于数据的时变现象，$P^t_{train}(y|x)\neq P^t_{test}(y|x)$。**

- **DDG-DA的核心思想：学习一个元模型M_θ能对训练集$D^t_{train}(x,y)\sim P^t_{train}(y|x)$不同时间点的样本赋权，使得与当前市场风格接近的数据权重高，与当前市场风格不接近的数据权重低。**重新赋权生成新的训练集$D^t_{resample}(x,y;\theta)\sim P^t_{resample}(y|x;\theta)$与测试集分布$P^t_{resample}(y|x;\theta)$相似，从而满足IID假设，实现模型泛化。

图表：DDG-DA 的核心思想和工作流程



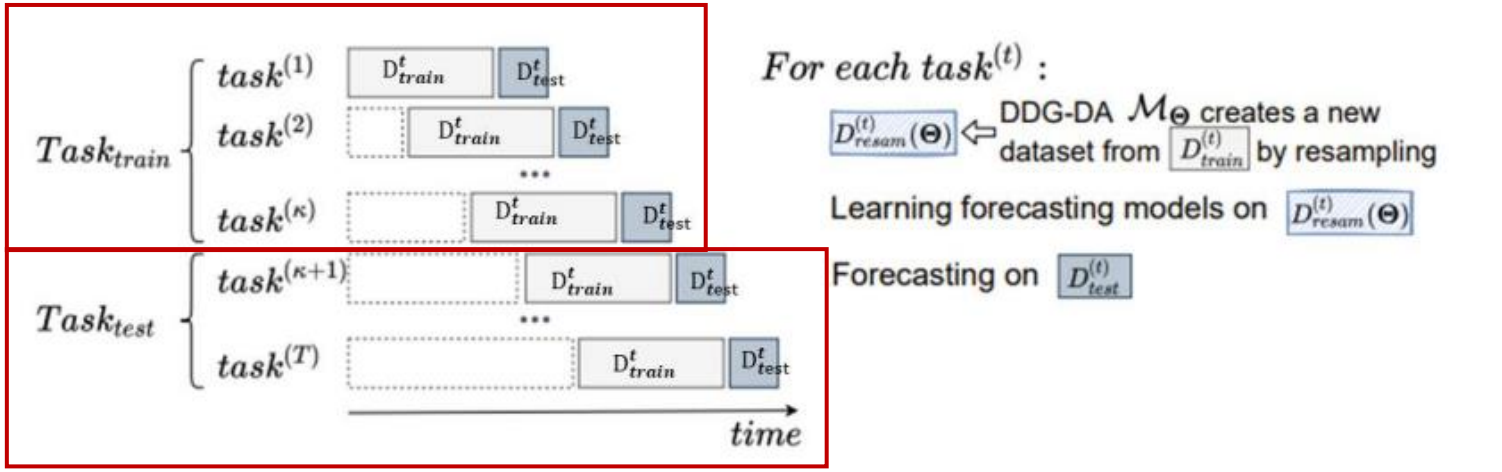
资料来源：华泰研究

DDG-DA的工作流程：训练过程和预测过程

- **DDG-DA的训练过程：**构建训练任务$\mathbf{Task}_{train}=\{task^1,task^2,\dots,task^{\kappa}\}$，元模型$M_\theta$将会对训练任务中的每一个$task^t=(D^t_{train},D^t_{test})_{t=1,\dots,\kappa}$中的训练数据$D^t_{train}$的每一个样本赋予一个新的权重，从而可以获得对应的重采样样本$D^t_{resample}(\theta)$，**训练过程中DDG-DA的优化算法将会尽可能减少$\mathbf{Task}_{train}$里重采样样本$D^t_{resample}(\theta)$与D^t_{test}间的分布差异。**

- **DDG-DA的测试过程：**元模型M_θ在测试任务$\mathbf{Task}_{test}=\{task^{\kappa+1},\dots,task^{T}\}$的每一个任务上生成新的数据集$D^t_{resample}(\theta)_{t=\kappa+1,\dots,T}$，然后用量化模型(例如LGBM)将在其上训练，最终提升量化模型(例如LGBM)在$D^t_{test}{}_{t=\kappa+1,\dots,T}$上的泛化能力。

图表：DDG-DA 的核心思想和工作流程



资料来源：华泰研究

DDG-DA的如何训练元模型M_θ

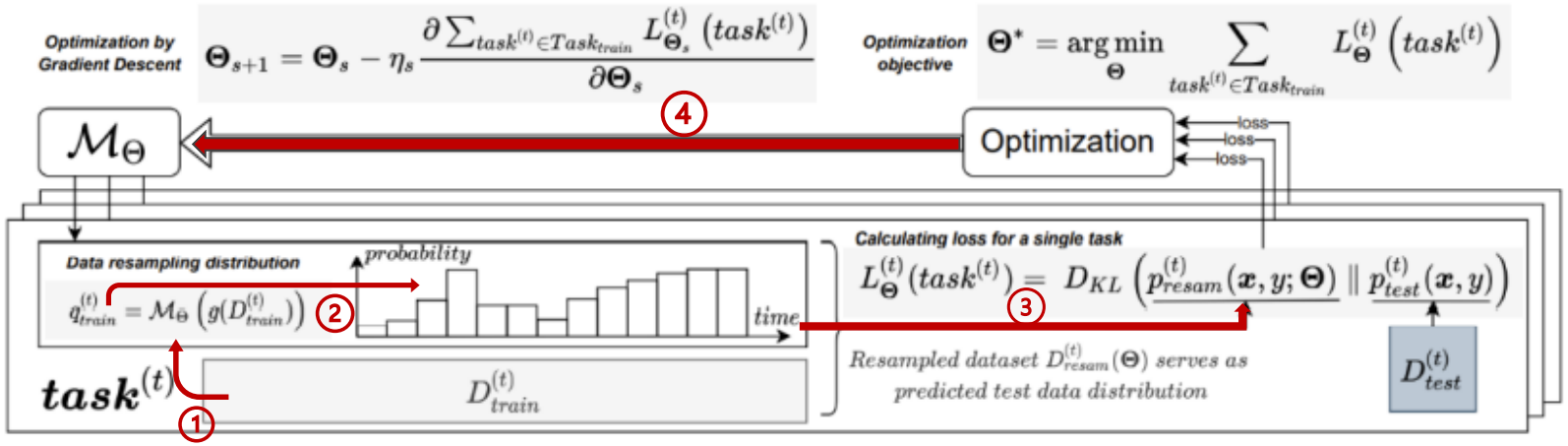
- **输入输出：**DDG-DA元模型M_θ的输入是历史数据D^t_{train}，通过特征提取函数g，模型M_θ将提取其中的数据分布信息，输出不同时间点的样本权重Q^t_{train}。
- **损失函数：**模型M_θ输出样本权重Q^t_{train}后，可以得到$D^t_{resample}(\theta)\sim P^t_{resample}(y|x;\theta)$，用$KL$散度来衡量$D^t_{resample}(\theta)\sim P^t_{resample}(y|x;\theta)$和$D^t_{test}(x,y)\sim P^t_{test}(y|x)$的差别：

$$L^t_\theta(task^t) = D_{KL}(P^t_{resample}(y|x;\theta)||P^t_{test}(y|x))$$

- **优化算法：**通过梯度下降来使得损失函数在所有的$Task_{train}$上最小：

$$\Theta^* = argmin \sum_{task^t \in Task_{train}} L^t_\theta(task^t)$$

图表24：图表：DDG-DA的训练任务$Task_{train}$中单个$task^t$训练过程的详细剖析



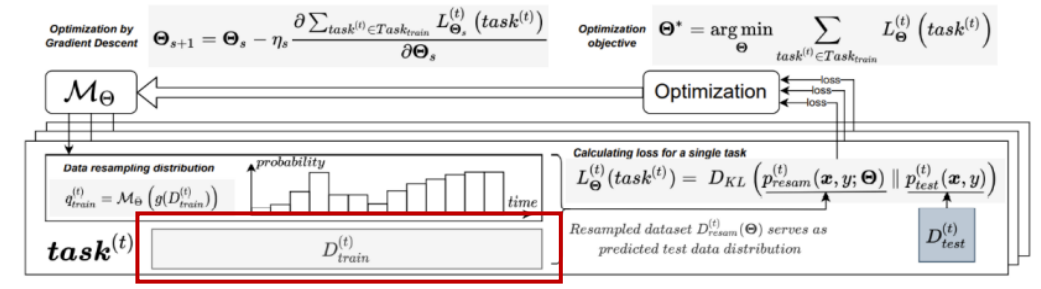
资料来源：华泰研究

DDG-DA 模型训练细节

□ 原始数据集D_{train}：

以$Task_{train}\{task^1, task^2, \dots, task^K\}$中的$task^1$为例，$D_{train}$我们取全部A股（剔除ST、PT股票，剔除每个截面期下一交易日停牌的股票）2007年1月5日-2013年2月8日周频的73个因子值和对应的10日收益率。DDG-DA模型最终目的就是对2007年1月5日-2013年2月8日这350个时间截面赋权。

图表24：图表：DDG-DA 的训练任务$Task_{train}$中单个$task^t$训练过程的详细剖析



资料来源：华泰研究

图表：原始数据集D_{train}

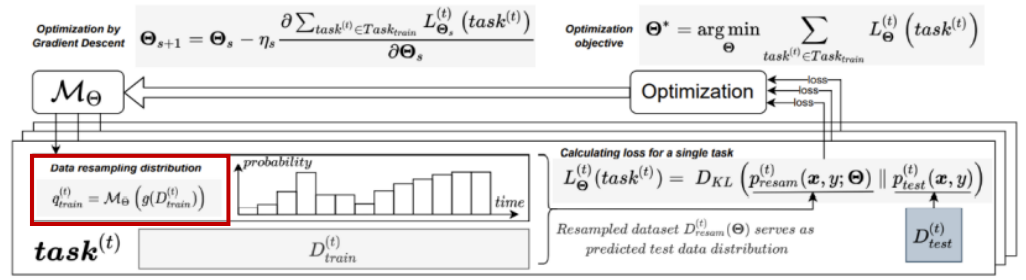
datetime	instrument	feature					label
		factor_1	factor_2	factor_3	…	factor73	label10
2007/1/5	000001.SZ	0.512890423	0.179639678	0.487496431	…	0.667159586	0.591464025
	000002.SZ	0.11440001	0.763033794	0.782481007	…	0.59775708	0.246776392
	000005.SZ	0.639344564	0.559061217	0.893257491	…	0.23956949	0.802060949
	…	…	…	…	…	…	…
	688981.SH	0.852502441	0.71479866	0.559555052	…	0.496231419	0.287937946
2007/1/12	000001.SZ	0.21444967	0.248956223	0.736837848	…	0.734531233	0.449019224
	000002.SZ	0.839420022	0.559738715	0.051935888	…	0.481059208	0.67546392
	000005.SZ	0.232678054	0.710981327	0.328352519	…	0.396383686	0.810655113
	…	…	…	…	…	…	…
	688981.SH	0.048046822	0.183040681	0.641446706	…	0.416611825	0.591559133
…	…	…	…	…	…	…	…
2013/2/8	000001.SZ	0.257930286	0.394248825	0.226132687	…	0.803899917	0.360742559
	000002.SZ	0.584531533	0.581705234	0.394651027	…	0.485037055	0.384354195
	000005.SZ	0.99136146	0.436811365	0.042723669	…	0.442285233	0.708678023
	…	…	…	…	…	…	…
	688981.SH	0.116089703	0.133231297	0.668988335	…	0.885488087	0.496507807

资料来源：华泰研究

DDG-DA 模型训练细节

- **分布信息提取$g(D_{train})$**：选取某个截面，进行横截面回归，再在整体数据集上进行预测，从而可以得到该截面与其他周IC。以此类推，我们可以得到每个截面与其他周数据的IC。为了防止未来信息，需要将当月训练在当月上预测的IC剔除。

图表24：图表：DDG-DA 的训练任务$Task_{train}$中单个$task^t$训练过程的详细剖析



资料来源：华泰研究

- **直观上来看，IC越高代表该截面与其他周的数据分布越相似。**

图表：分布信息提取$g(D_{train})$

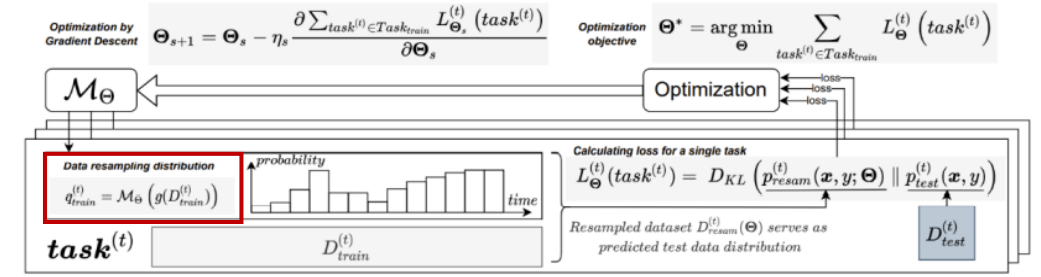
	2007/1/5	2007/1/12	2007/1/19	…	2013/2/1	2013/2/8
2012/1/7	NaN	0.072057734	0.03515197	…	0.068979688	0.08336474
2012/1/14	0.021659685	NaN	0.058094948	…	0.001093949	0.051755687
2012/1/21	0.082923735	0.000526999	NaN	…	0.030133787	0.016239311
…	…	…	…	…	…	…
2013/2/1	0.051081145	0.08197103	0.052806006	…	NaN	0.025667936
2013/2/8	0.03122201	0.019935849	0.088116966	…	0.005293644	NaN

资料来源：华泰研究

DDG-DA 模型训练细节

- **权重生成$\mathbf{Q}_{train} = \boldsymbol{M}_{\theta}(g(\boldsymbol{D}_{train}))$：**对提取的分布填充空值，提取最近60个截面与$D_{train}$所有周的$IC$数据。将其作为输入，通过$\boldsymbol{M}_{\theta}$（一层全连接层+tanh 激活函数）即可得到每周数据的权重Q_{train}。

图表24：图表：DDG-DA 的训练任务$Task_{train}$中单个$task^t$训练过程的详细剖析

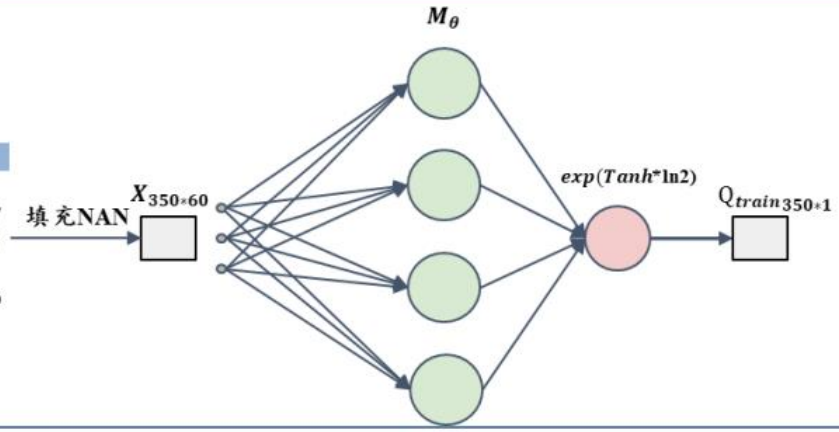


资料来源：华泰研究

- **DDG-DA核心假设是过去60周截面与历史数据的分布相似情况，和未来几周与历史数据的分布相似情况之间存在一定规律。**

图表：权重生成$\mathbf{Q}_{train} = \boldsymbol{M}_{\theta}(g(\boldsymbol{D}_{train}))$

	2007/1/5	2007/1/12	2007/1/19	…	2013/2/1	2013/2/8
2012/1/7	NaN	0.072057734	0.03515197	…	0.068979688	0.08336474
2012/1/14	0.021659685	NaN	0.058094948	…	0.001093949	0.051755687
2012/1/21	0.082923735	0.000526999	NaN	…	0.030133787	0.016239311
…	…	…	…	…	…	…
2013/2/1	0.051081145	0.08197103	0.052806006	…	NaN	0.025667936
2013/2/8	0.03122201	0.019935849	0.088116966	…	0.005293644	NaN

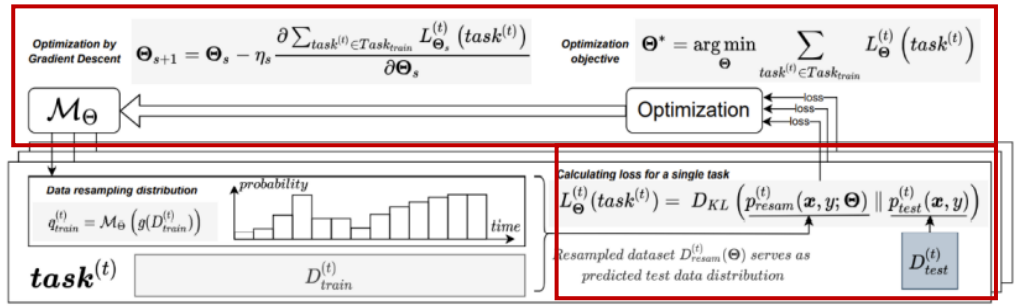


资料来源：华泰研究

DDG-DA 模型训练细节

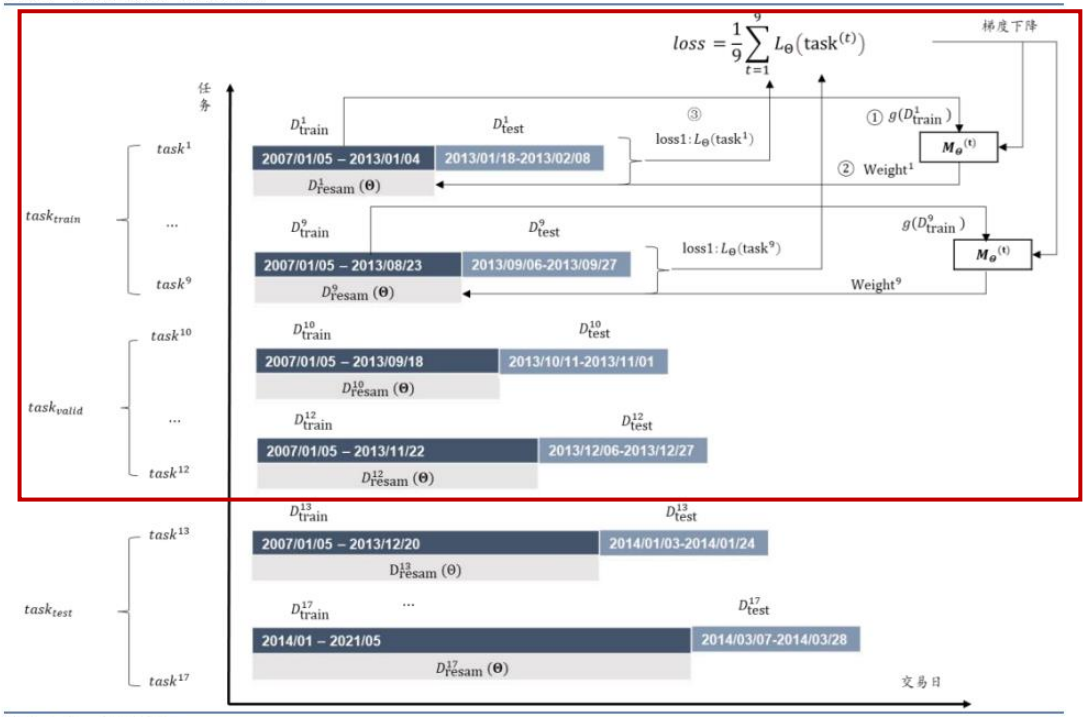
- **损失函数和优化算法：**如右下图所示，训练任务task^1，输入D^1_{train}，经过分布信息提取，权重生成，M_θ输出Q^1_{train}和$D^1_{resample}(\theta)$，进一步可以得到$task^1$的损失$L_\Theta(task^1)$。以此类推，经过9个$task$，得到整体训练任务的损失$\frac{1}{9}\sum_{t=1}^{9} L_\Theta(task^t)$，最后梯度下降，完成一个epoch。

图表24：图表：DDG-DA 的训练任务$Task_{train}$中单个$task^t$训练过程的详细剖析



资料来源：华泰研究

图表：损失函数和优化算法



资料来源：华泰研究

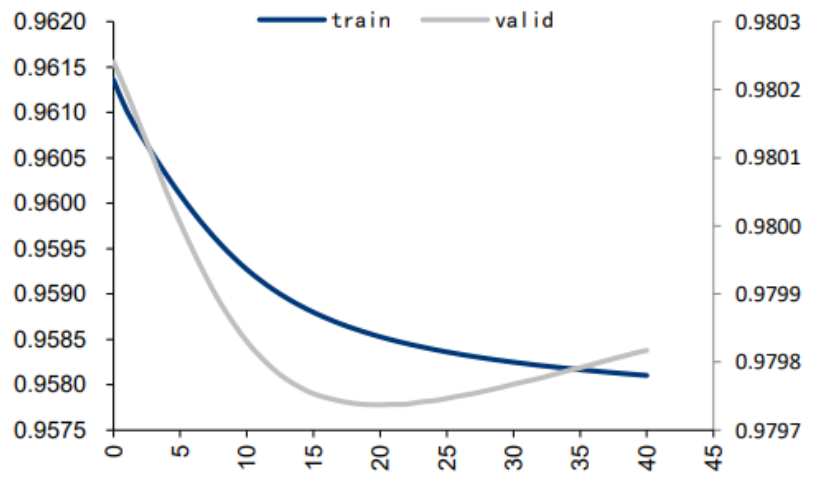
训练过程正常，说明学习到了数据中的时变规律

□ **测试细节：**

- **滚动训练：**从2014/01/03开始，每3个月滚动训练一次DDG-DA模型（取2007年开始至今的数据构建$Task_{train}$，用最后三个$task$的数据作为验证集，并用LGBM在$task_{test}$上建立量化模型）。$BenchMark$我们采用同样时间划分下，等权滚动训练的LGBM。
- **调仓周期：**策略周频率调仓，分层测试交易费为单边千分之二。

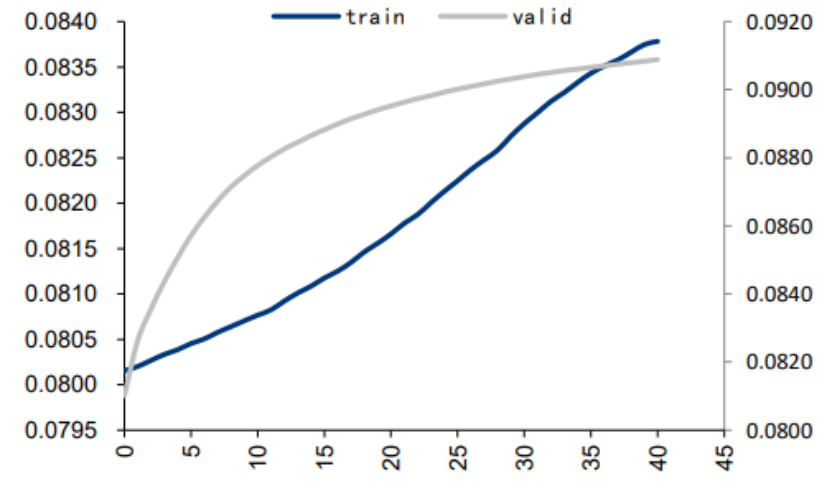
□ DDG-DA在滚动训练的过程中，我们发现只有50%季度可以收敛，如2021/8/21-2021/11/12。

图表：训练和验证 task 的损失变化曲线



注：训练 task：2007/1/5～2021/8/20；测试 task：2021/8/21～2021/11/19
资料来源：华泰研究

图表：训练和验证 task 的 IC 变化曲线

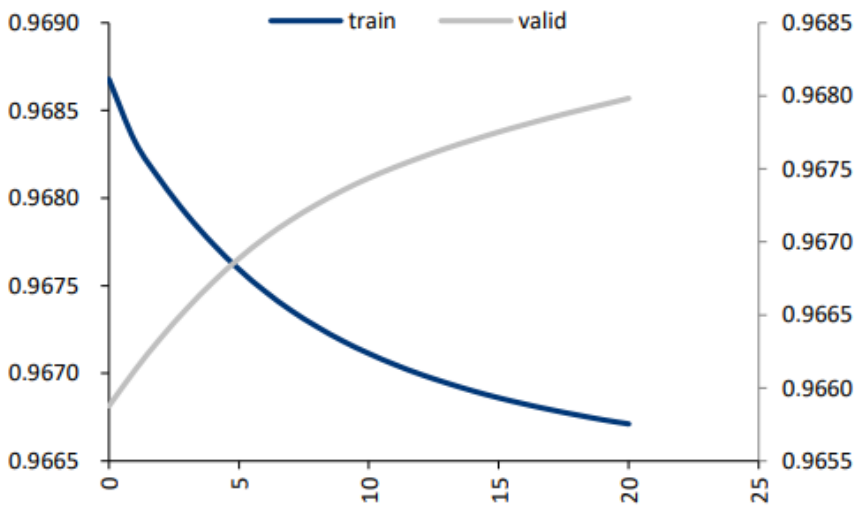


注：训练 task：2007/1/5～2021/8/20；测试集：2021/8/21～2021/11/19
资料来源：华泰研究

训练过程不正常，说明无法学习到数据中的时变规律

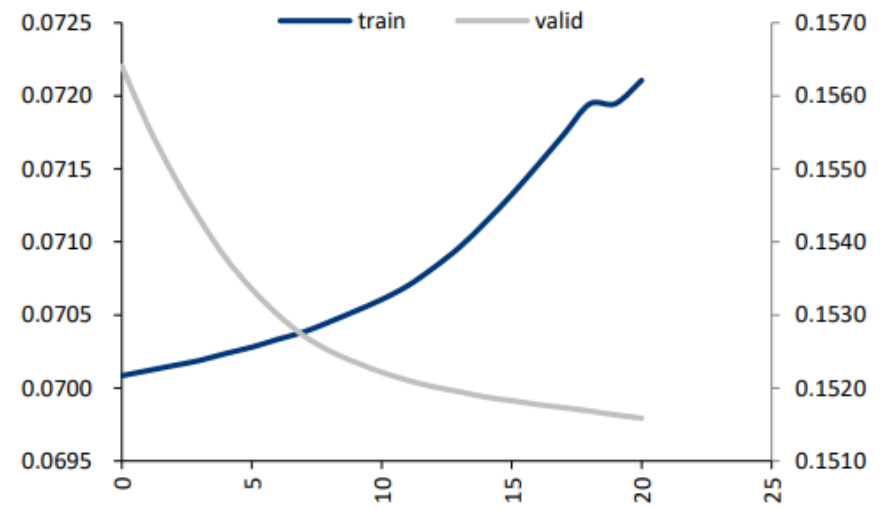
□ 而2021/11/26-2022/2/11 上DDG-DA元模型M_θ无法收敛，我们认为这是之前提到的核心假设**（过去60周截面与历史数据的分布相似情况，和未来几周与历史数据的分布相似情况之间存在的规律）**发生变化。因此我们在这段时间将会用等权来代替DDG-DA元模型M_θ给出的权重。

图表：训练和验证 task 的损失变化曲线



注：训练 task：2007/1/5～2021/11/19；测试 task：2021/11/26～2022/2/18
资料来源：华泰研究

图表：训练和验证 task 的 IC 变化曲线



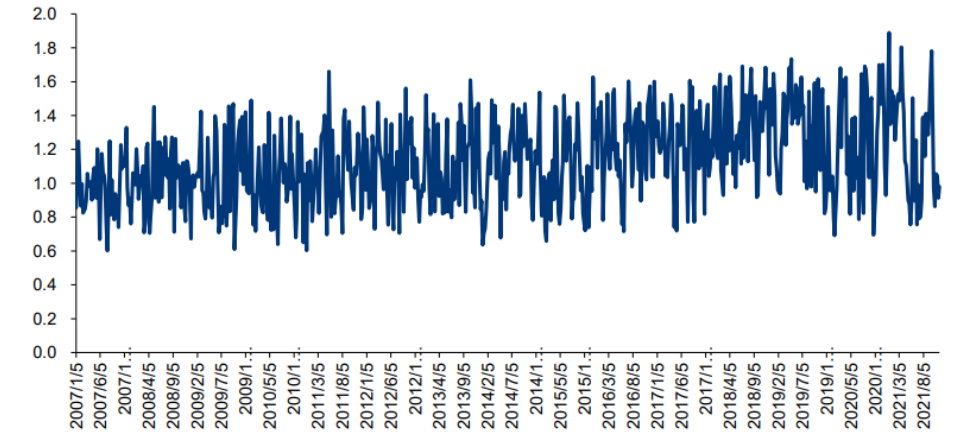
注：训练 task：2007/1/5～2021/11/19；测试 task：2021/11/26～2022/2/18
资料来源：华泰研究

数据的时间权重分析

- 2021/8/21-2021/11/12 DDG-DA元模型M_θ给出的权重如左图所示，可以看到尽管存在一定的波动，但最近一段时间的权重较高，符合我们的直觉。

- 而2021/11/26-2022/2/11 上DDG-DA元模型M_θ无法收敛，我们用等权代替。

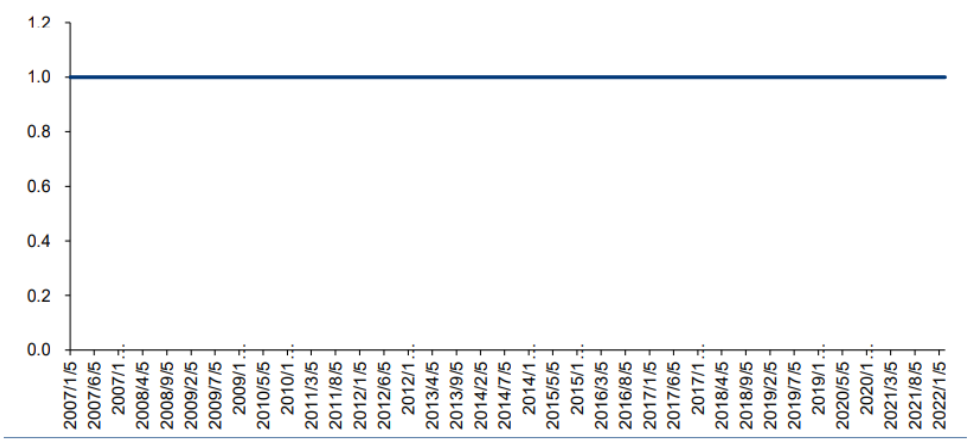
图表： DDG-DA 模型 2007/1/5～2021/11/12 样本权重分布情况



注：回测期：2007/1/5～2021/11/12

资料来源：华泰研究

图表： DDG-DA 模型 2007/1/5～2021/2/11 样本权重分布情况



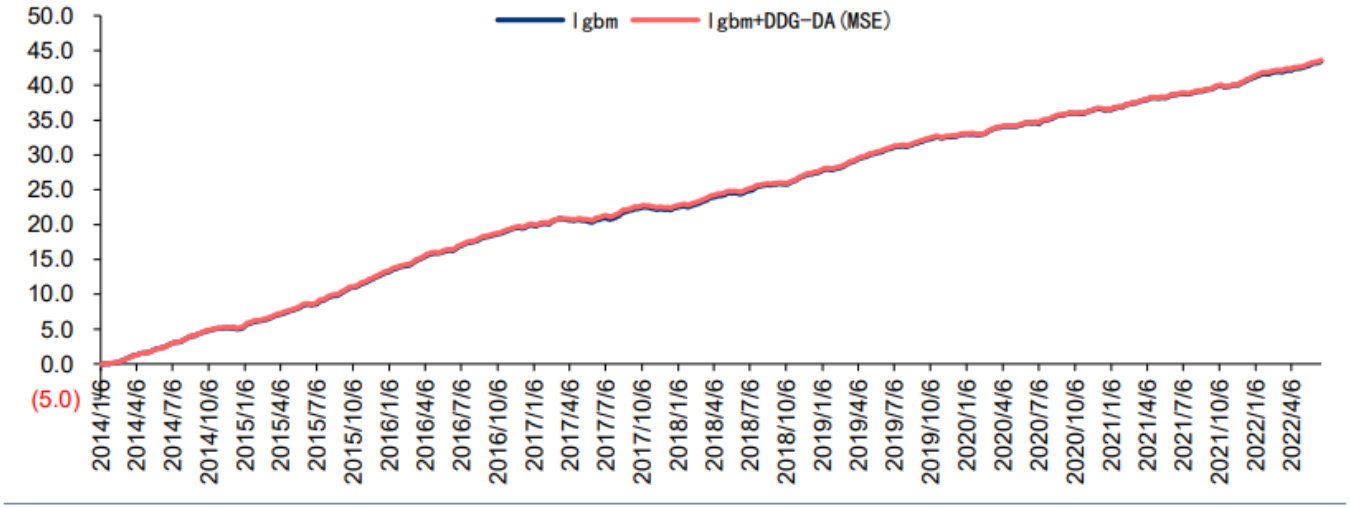
注：回测期：2007/1/5～2021/2/11

资料来源：华泰研究

DDG-DA 测试结果

□ DDG-DA在$RankIC$、IC_IR、$IC>0$占比上有一定优势。

图表：lgbm 和 lgbm+DDG-DA(MSE)的累计 RankIC



资料来源：华泰研究

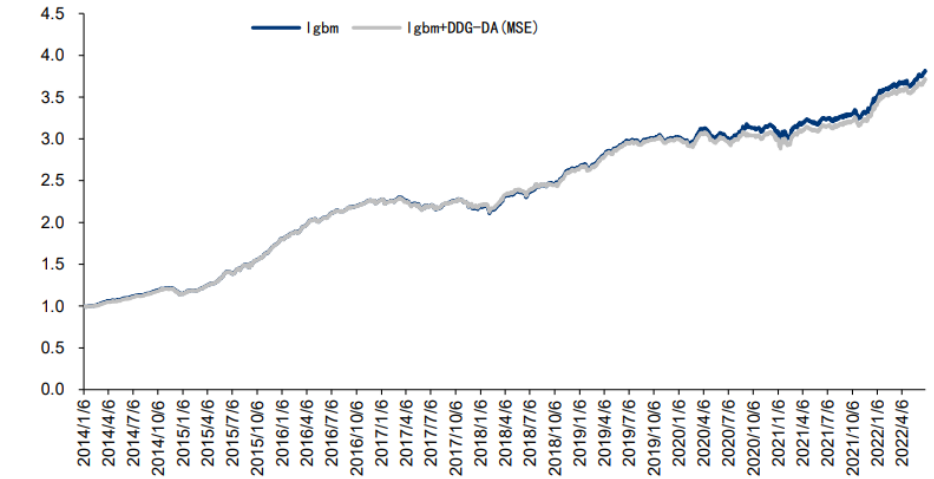
	RankIC均值	IC_IR	IC>0占比	组合1年化超额收益率	TOP组合信息比率
lgbm	0.1023	0.9177	0.8282	0.1813	3.4087
lgbm+DDG-DA(MSE)	0.1027	0.9385	0.8353	0.1775	3.3822

<华泰研究>

DDG-DA 测试结果

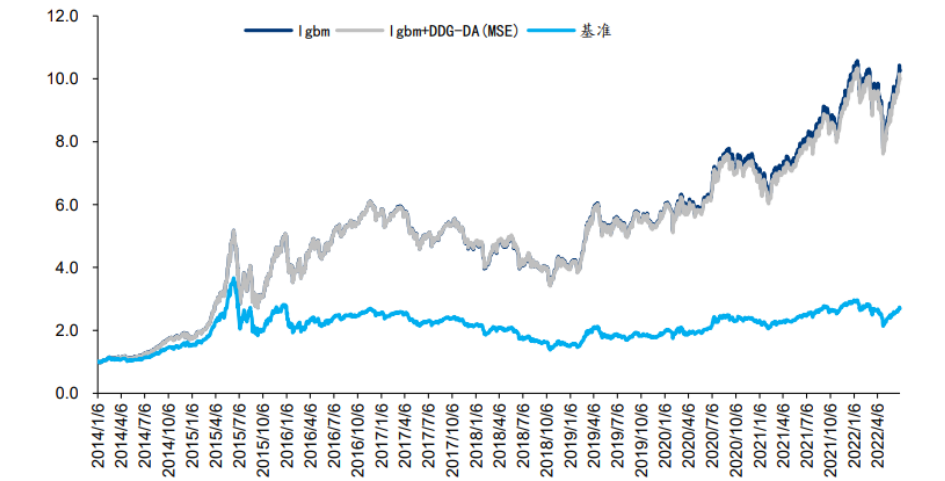
☐ DDG-DA在TOP组合年化超额收益率和TOP组合信息比率反而有所降低。

图表：lgbm 和 lgbm+DDG-DA(MSE)的第 1 层因子分层，手续费千二



资料来源：华泰研究

图表：lgbm 和 lgbm+DDG-DA(MSE)的 top 组合净值，手续费千二



资料来源：华泰研究

	RankIC均值	IC_IR	IC>0占比	组合1年化超额收益率	TOP组合信息比率
lgbm	0.1023	0.9177	0.8282	0.1813	3.4087
lgbm+DDG-DA(MSE)	0.1027	0.9385	0.8353	0.1775	3.3822

<华泰研究>

DDG-DA 测试结果：损失函数的改进

□ **DDG-DA 的损失函数：** 用KL散度来衡量新生成的数据集和测试集的差别：

$$L_\theta^t(task^t) = D_{KL}(\boldsymbol{P}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\boldsymbol{y}|\boldsymbol{x};\theta)||\boldsymbol{P}^{\boldsymbol{t}}_{\boldsymbol{test}}(\boldsymbol{y}|\boldsymbol{x}))$$

假设$\boldsymbol{P}^{\boldsymbol{t}}_{\boldsymbol{test}}(\boldsymbol{y}|\boldsymbol{x})\sim\mathbf{N}(\mathbf{y}^{\boldsymbol{t}}_{\boldsymbol{test}}(\mathbf{x}),\boldsymbol{\sigma}),\boldsymbol{P}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\boldsymbol{y}|\boldsymbol{x};\theta)\sim\mathbf{N}\left(\mathbf{y}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\mathbf{x};\boldsymbol{\theta}),\boldsymbol{\sigma}\right)$，损失函数简化为：

$$L_\theta^t(task^t) = \frac{1}{2}\sum_{(x,y)\in D^t_{test}} \left| \mathbf{y}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\mathbf{x}=\mathbf{x}_{\boldsymbol{test}};\boldsymbol{\theta}) - \mathbf{y} \right|^2$$

□ 除了用MSE来衡量新生成的数据集和测试集的差别，**我们认为在金融场景下，用IC或者强调多头的其他损失函数更加合理**，我们尝试了如下损失函数：

- IC
- $Weighted-IC$
- $RELU$

□ 其中$RELU$损失函数的IC和多头表现最好。

	RankIC均值	IC_IR	IC>0占比	组合1年化超额收益率	TOP组合信息比率
lgbm	0.1023	0.9177	0.8282	0.1813	3.4087
lgbm+DDG-DA(MSE)	0.1027	0.9385	0.8353	0.1775	3.3822
lgbm+DDG-DA(IC)	0.1029	0.9221	0.8306	0.1799	3.3608
lgbm+DDG-DA(Weighted-IC)	0.1031	0.9353	0.8329	0.1791	3.3975
lgbm+DDG-DA(RELU)	0.1034	0.9362	0.8329	0.1823	3.3608

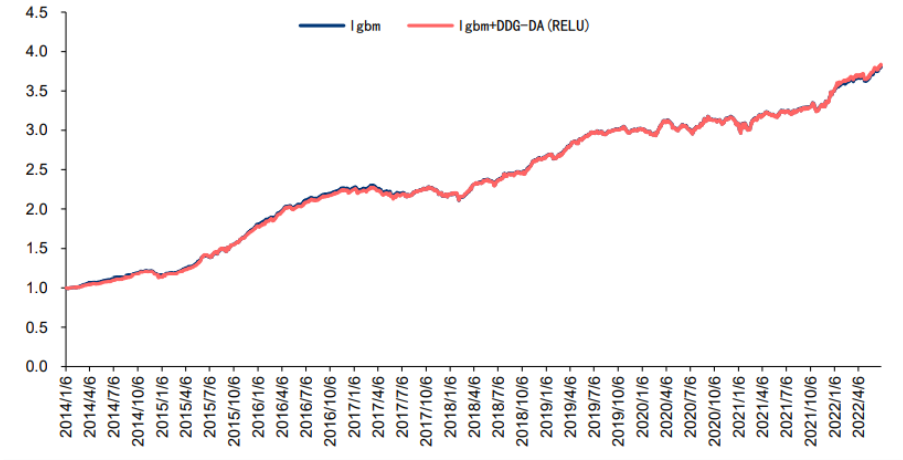
<华泰研究>

DDG-DA 测试结果：损失函数的改进

- *RELU*损失函数是在*MSE*损失函数的平方内层套入一个的RELU函数：

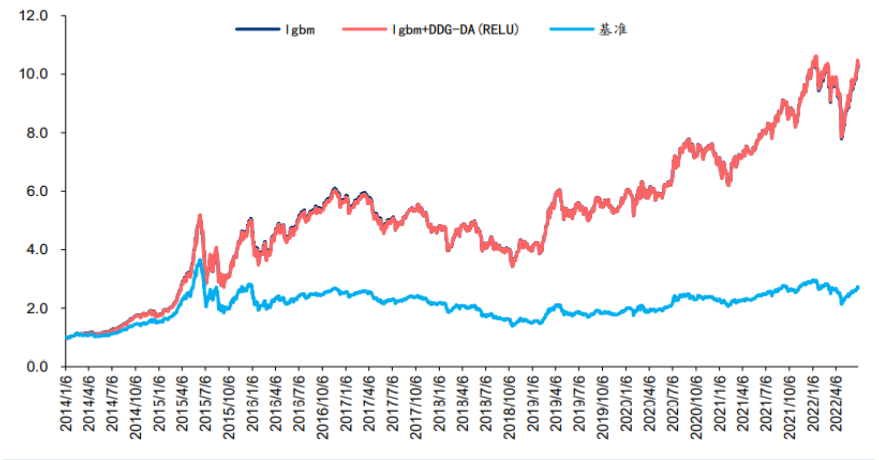
$$L_\theta^t(task^t) = \frac{1}{2}\sum_{(x,y)\in D_{test}^t} \left| \mathbf{Relu}(\mathbf{y}_{resample}^t(\mathbf{x}=\mathbf{x}_{test};\boldsymbol{\theta}) - \mathbf{y}) \right|^2$$

图表：lgbm 和 lgbm+DDG-DA(RELU)的第 1 层因子分层，手续费千二



资料来源：华泰研究

图表：lgbm 和 lgbm+DDG-DA(RELU)的 top 组合净值，手续费千二



资料来源：华泰研究

	RankIC均值	IC_IR	IC>0占比	组合1年化超额收益率	TOP组合信息比率
lgbm	0.1023	0.9177	0.8282	0.1813	3.4087
lgbm+DDG-DA(MSE)	0.1027	0.9385	0.8353	0.1775	3.3822
lgbm+DDG-DA(IC)	0.1029	0.9221	0.8306	0.1799	3.3608
lgbm+DDG-DA(Weighted-IC)	0.1031	0.9353	0.8329	0.1791	3.3975
lgbm+DDG-DA(RELU)	0.1034	0.9362	0.8329	0.1823	3.3608

<华泰研究>

DDG-DA的总结和讨论

- 数据时变现象（Domain shift）会导致多因子模型的样本外泛化能力减弱。
- DDG-DA学习元模型 M_θ 对训练集不同时间点的样本赋权，从而减小样本内外分布的差异。
- DDG-DA核心假设是过去60周截面与历史数据的分布相似情况，和未来几周与历史数据的分布相似情况之间存在一定规律。
- DDG-DA在对损失函数改进后，其表现在RankIC和TOP组合年化超额收益均有提升。

关于多因子选股模型中数据时变规律研究，仍有以下值得尝试的方向：

- DDG-DA提出了一个解决数据时变现象问题的很好的框架，即通过学习元模型 M_θ 对训练集不同时间点的样本赋权，从而减小样本内外分布的差异。但是在核心假设上，相比于目前采用的计算截面与其他时间段的IC，对于分布信息提取能否有更好的指示变量？

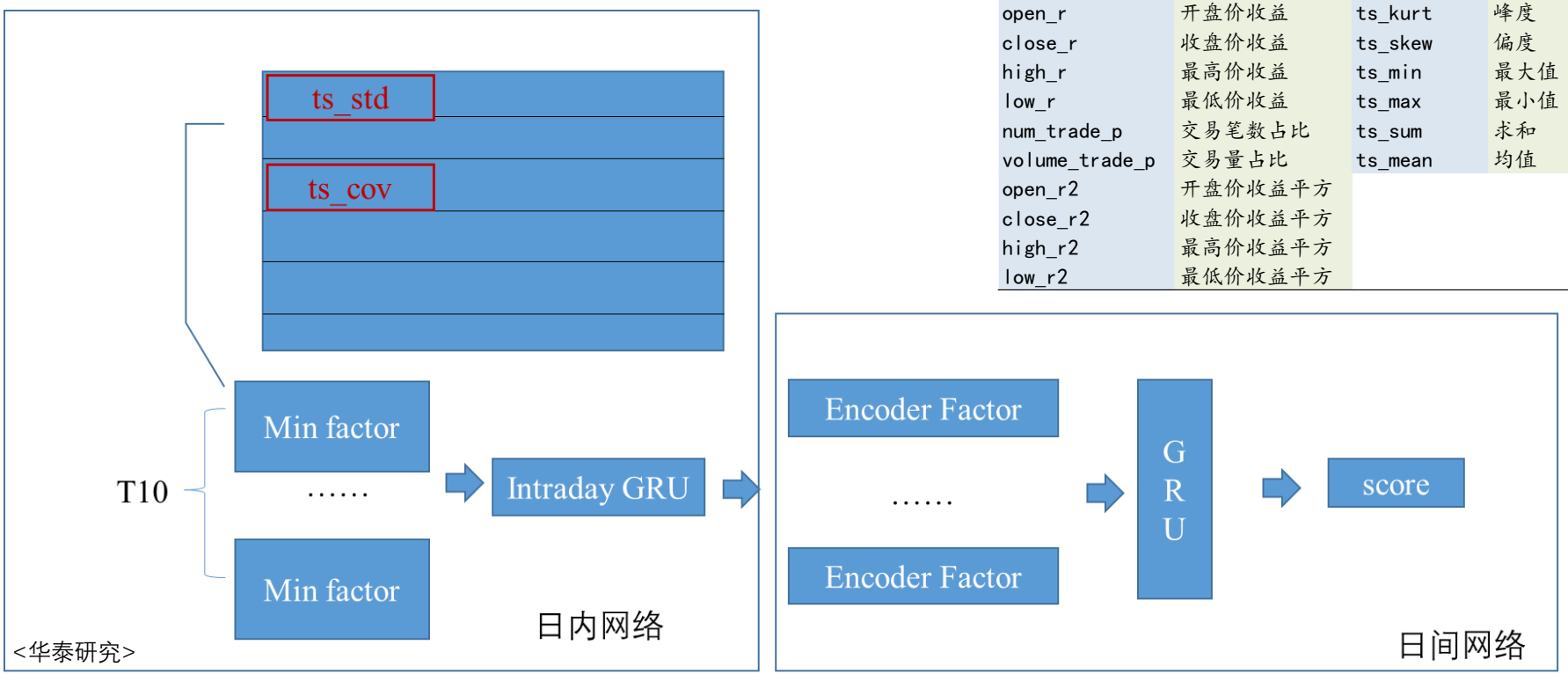
暑期其他研究内容

☐ 基于*AlphaNet*的分钟线数据合成日频因子：

- 市面上常见的高频因子合成日频因子，如尾盘成交额占比因子，可以用基本特征**分钟线内成交量占当日成交量比例** + ***ts_sum*算子**提取。

基础特征		算子	
open	开盘价	ts_pct	按条件筛选占比
close	收盘价	ts_corr	相关系数
high	最高价	ts_cov	协方差
low	最低价	ts_std	标准差
num_trade	交易笔数	ts_zscore	标准化
volume_trade	交易量	ts_deacy	过去N日收益
open_r	开盘价收益	ts_kurt	峰度
close_r	收盘价收益	ts_skew	偏度
high_r	最高价收益	ts_min	最大值
low_r	最低价收益	ts_max	最小值
num_trade_p	交易笔数占比	ts_sum	求和
volume_trade_p	交易量占比	ts_mean	均值
open_r2	开盘价收益平方		
close_r2	收盘价收益平方		
high_r2	最高价收益平方		
low_r2	最低价收益平方		

☐ 通过构建**层级神经网络**进行日频因子的合成。



<华泰研究>

暑期其他小课题

□ 分钟频数据库维护和高频因子更新：

- 重构分钟频数据库，研究*Dask*框架，实现lazy evaluation，有效解决内存爆炸的问题。
- 阅读市面上分钟频因子，复现因子，并更新入库。

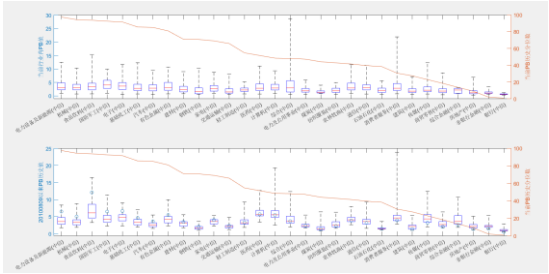
CloseVolPropFactor	DownwardVolaPropFactor	SkewnessKurtosisFactor	VolPriceCorrFacto
尾盘成交占比因子	下行成交量因子	峰度偏度因子	量价相关性因子
AvgOutFlowPropFactor	TimeSectionPropFactor	TimeSectionStatisticFactor	PriceRangeStatisticFactor
平均流出单量因子	时间截面因子	时间序列因子	价格区间因子

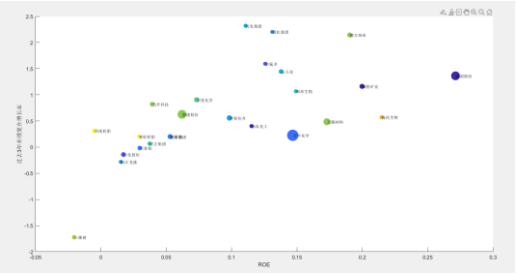
□ 基于估值因子的基金筛选策略：

- 构建低市值和高市值策略
- 编写回测框架

□ 行业板块估值可视化模块开发：

- 一级行业估值分布情况。
- 新能源\化工所有个股的估值散点图。
- 新能源\化工所有个股增速-估值图。





附录： PAC学习理论简化证明

Theorem 2.1 **Learning bounds — finite H, consistent case**

Let H be a finite set of functions mapping from $\mathcal{X}$ to $\mathcal{Y}$. Let $\mathcal{A}$ be an algorithm that for any target concept $c \in H$ and i.i.d. sample S returns a consistent hypothesis h_S: $\widehat{R}(h_S) = 0$. Then, for any $\epsilon, \delta > 0$, the inequality $\Pr_{S \sim D^m}[R(h_S) \leq \epsilon] \geq 1 - \delta$ holds if

$$m \geq \frac{1}{\epsilon}\left(\log |H| + \log \frac{1}{\delta}\right). \tag{2.8}$$

This sample complexity result admits the following equivalent statement as a generalization bound: for any $\epsilon, \delta > 0$, with probability at least $1 - \delta$,

$$R(h_S) \leq \frac{1}{m}\left(\log |H| + \log \frac{1}{\delta}\right). \tag{2.9}$$

附录： PAC学习理论简化证明

Proof Fix $\epsilon > 0$. We do not know which consistent hypothesis $h_S \in H$ is selected by the algorithm $\mathcal{A}$. This hypothesis further depends on the training sample S. Therefore, we need to give a *uniform convergence bound*, that is, a bound that holds for the set of all consistent hypotheses, which a fortiori includes h_S. Thus, we will bound the probability that some $h \in H$ would be consistent and have error more than ϵ:

$$
\begin{aligned}
&\Pr[\exists h \in H \colon \widehat{R}(h) = 0 \wedge R(h) > \epsilon] \\
&= \Pr[(h_1 \in H,\, \widehat{R}(h_1) = 0 \wedge R(h_1) > \epsilon) \vee (h_2 \in H,\, \widehat{R}(h_2) = 0 \wedge R(h_2) > \epsilon) \vee \cdots] \\
&\leq \sum_{h \in H} \Pr[\widehat{R}(h) = 0 \wedge R(h) > \epsilon] && \text{(union bound)} \\
&\leq \sum_{h \in H} \Pr[\widehat{R}(h) = 0 \mid R(h) > \epsilon]. && \text{(definition of conditional probability)}
\end{aligned}
$$

Now, consider any hypothesis $h \in H$ with $R(h) > \epsilon$. Then, the probability that h would be consistent on a training sample S drawn i.i.d., that is, that it would have no error on any point in S, can be bounded as:

$$\Pr[\widehat{R}(h) = 0 \mid R(h) > \epsilon] \leq (1 - \epsilon)^m.$$

附录： PAC学习理论简化证明

The previous inequality implies

$$\Pr[\exists h \in H : \widehat{R}(h) = 0 \wedge R(h) > \epsilon] \leq |H|(1-\epsilon)^m.$$

Setting the right-hand side to be equal to δ and solving for ϵ concludes the proof. ■

The theorem shows that when the hypothesis set H is finite, a consistent algorithm $\mathcal{A}$ is a PAC-learning algorithm, since the sample complexity given by (2.8) is dominated by a polynomial in $1/\epsilon$ and $1/\delta$. As shown by (2.9), the generalization error of consistent hypotheses is upper bounded by a term that decreases as a function of the sample size m. This is a general fact: as expected, learning algorithms benefit from larger labeled training samples. The decrease rate of $O(1/m)$ guaranteed by this theorem, however, is particularly favorable.

The price to pay for coming up with a consistent algorithm is the use of a larger hypothesis set H containing target concepts. Of course, the upper bound (2.9) increases with $|H|$. However, that dependency is only logarithmic. Note that the term $\log |H|$, or the related term $\log_2 |H|$ from which it differs by a constant factor, can be interpreted as the number of bits needed to represent H. Thus, the generalization guarantee of the theorem is controlled by the ratio of this number of bits, $\log_2 |H|$, and the sample size m.

附录： KL散度

K-L散度源于信息论。信息论主要研究如何量化数据中的信息。最重要的信息度量单位是熵Entropy，一般用H表示。分布的熵的公式如下：

$$H = -\sum_{i=1}^{N} p(x_i) \cdot \log p(x_i)$$

只需要稍加修改熵H的计算公式就能得到K-L散度的计算公式。设p为观察得到的概率分布，q为另一分布来近似p，则p、q的K-L散度为：

$$D_{KL}(p||q) = \sum_{i=1}^{N} p(x_i) \cdot log\frac{p(x_i)}{q(x_i)}$$

附录：DDG-DA训练过程的推导

- **输入输出：**DDG-DA元模型$\boldsymbol{M_\Theta}$的输入是历史数据$\boldsymbol{D^t_{train}}$，通过特征提取函数$\boldsymbol{g}$，模型$\boldsymbol{M_\Theta}$将提取其中的数据分布信息，输出不同时间点的样本权重$\boldsymbol{Q^t_{train}}$。DDG-DA元模型可以表示为$\boldsymbol{Q^t_{train}} = \boldsymbol{M_\Theta}\left(\boldsymbol{g}(\boldsymbol{D^t_{train}})\right)$，

- **DDG-DA 的损失函数：**得到模型$\boldsymbol{M_\Theta}$输出不同时间点的样本权重$\boldsymbol{Q^t_{train}}$后，可以得到$\boldsymbol{D^t_{resample}}(\boldsymbol{\theta})$~$\boldsymbol{P^t_{resample}}(\boldsymbol{y}|\boldsymbol{x};\theta)$，我们用$KL$散度来衡量新生成的数据集和测试集的差别：

$$L^t_\theta(task^t) = D_{KL}(\boldsymbol{P^t_{resample}}(\boldsymbol{y}|\boldsymbol{x};\theta)||\boldsymbol{P^t_{test}}(\boldsymbol{y}|\boldsymbol{x}))$$

对KL散度简化，假设$\boldsymbol{P^t_{test}}(\boldsymbol{y}|\boldsymbol{x})$~$\mathbf{N}(\mathbf{y}^t_{test}(\mathbf{x}), \boldsymbol{\sigma})$, $\boldsymbol{P^t_{resample}}(\boldsymbol{y}|\boldsymbol{x};\theta)$~$\mathbf{N}\left(\mathbf{y}^t_{resample}(\mathbf{x};\boldsymbol{\theta}), \boldsymbol{\sigma}\right)$, $\boldsymbol{\sigma}$是一个常数。于是损失函数简化为：

$$L^t_\theta(task^t) = \frac{1}{2}\sum(x,y) \in D^t_{test} \left| \quad \mathbf{y}^t_{resample}(\mathbf{x} = \mathbf{x}_{test};\boldsymbol{\theta}) - \mathbf{y} \right|^2$$

- **DDG-DA 的优化算法：**我们的优化目标是：

$$\Theta^* = argmin \sum task^t \in Task_{train}\ L^t_\theta(task^t)$$

$$= \sum task^t \in Task_{train} \frac{1}{2}\sum(x,y) \in D^t_{test} \left| \quad \mathbf{y}^t_{resample}(\mathbf{x} = \mathbf{x}_{test};\boldsymbol{\theta}) - \mathbf{y} \right|^2$$

附录：DDG-DA训练过程的推导

- DDG-DA元模型$\boldsymbol{M_\theta}$只能输出样本权重$\boldsymbol{Q^t_{train}}$，为了获得$\mathbf{y}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\mathbf{x};\boldsymbol{\theta})$，我们需要引入一个委托模型$\boldsymbol{f}(\phi)$，在$\boldsymbol{D^t_{resample}}(\boldsymbol{\theta})$上拟合后，来估计$\mathbf{y}^{\boldsymbol{t}}_{\boldsymbol{resample}}(\mathbf{x}=\mathbf{x}_{\boldsymbol{test}};\boldsymbol{\theta})= f(x=x_{test};\phi,\theta)$:

$$\Phi^{t^*} = \text{argmin}\sum(x,y)\in D^t_{resmple}(\theta)\left| \quad f(x;\phi,\theta)-y\right|^2$$

- 于是我们的优化目标变为一个两层的优化问题:

$$argmin_\Theta \sum task^t \in Task_{train}(\sum(x,y)\in D^t_{test}\left| \quad f(x=x_{test};\phi^{t^*},\theta)-y\right|^2)$$

$$_{s.t.}\ \Phi^{t^*} = \text{argmin}\sum(x',y')\in D^t_{resmple}(\theta)\left| \quad f(x';\phi,\theta)-y'\right|^2$$

- 为了简化优化难度，我们假设$f(\phi)=x\phi$，我们可以得到解析解

$$\phi^{t^*} = \left((x^t)^T Q^t X^t\right)^{-1}(x^t)^T Q^t y^t$$

- 于是我们的优化目标简化为一个一层的优化问题:

$$argmin_\Theta \sum task^t \in Task_{train}\sum_{(\boldsymbol{x},\boldsymbol{y})\in \boldsymbol{D^t_{test}}}\left|\left((\boldsymbol{x^t_{train}})^T\boldsymbol{M_\theta}\left(\boldsymbol{g}(\boldsymbol{D^t_{train}})\right)\boldsymbol{X^t_{train}}\right)^{-1}(\boldsymbol{x^t_{train}})^T\boldsymbol{M_\theta}\left(\boldsymbol{g}(\boldsymbol{D^t_{train}})\right)\boldsymbol{y^t_{train}}\boldsymbol{x_{test}}-\mathbf{y}_{\boldsymbol{test}}\right|^2$$