

StockTransformer

吴嘉赞 T2018011190

胥嘉政 T2018011300

Main Results

Performance on validation sets.

F0. 5_5	F0. 5_10	F0. 5_20	F0. 5_40	F0. 5_60
0. 355				
PNL_5	PNL_10	PNL_20	PNL_40	PNL_60

* F0. 5在日期范围[64, 79)测试
* PNL在日期范围[74, 79)测试

1. Settings

时间序列建模

问题描述

- 设某一支股票在过去 T 时刻内的因子序列为 $X[0:T]$, $X \in \mathbb{R}^n$ 。
- 目标是预测未来的离散的 L 个交易日的价格变动 $Y[0:L]$, 其中 $Y_l \in \{0, 1, 2\}$ 。

建模为时间序列的分类问题, 使用神经网络 f_θ 进行学习。

Learning Objective

$$\theta = \arg \min_{\theta} D_{KL}(Q(Y|X), P(Y|X; \theta))$$

其中 Q 是数据的真实分布, P 是神经网络的预测分布。

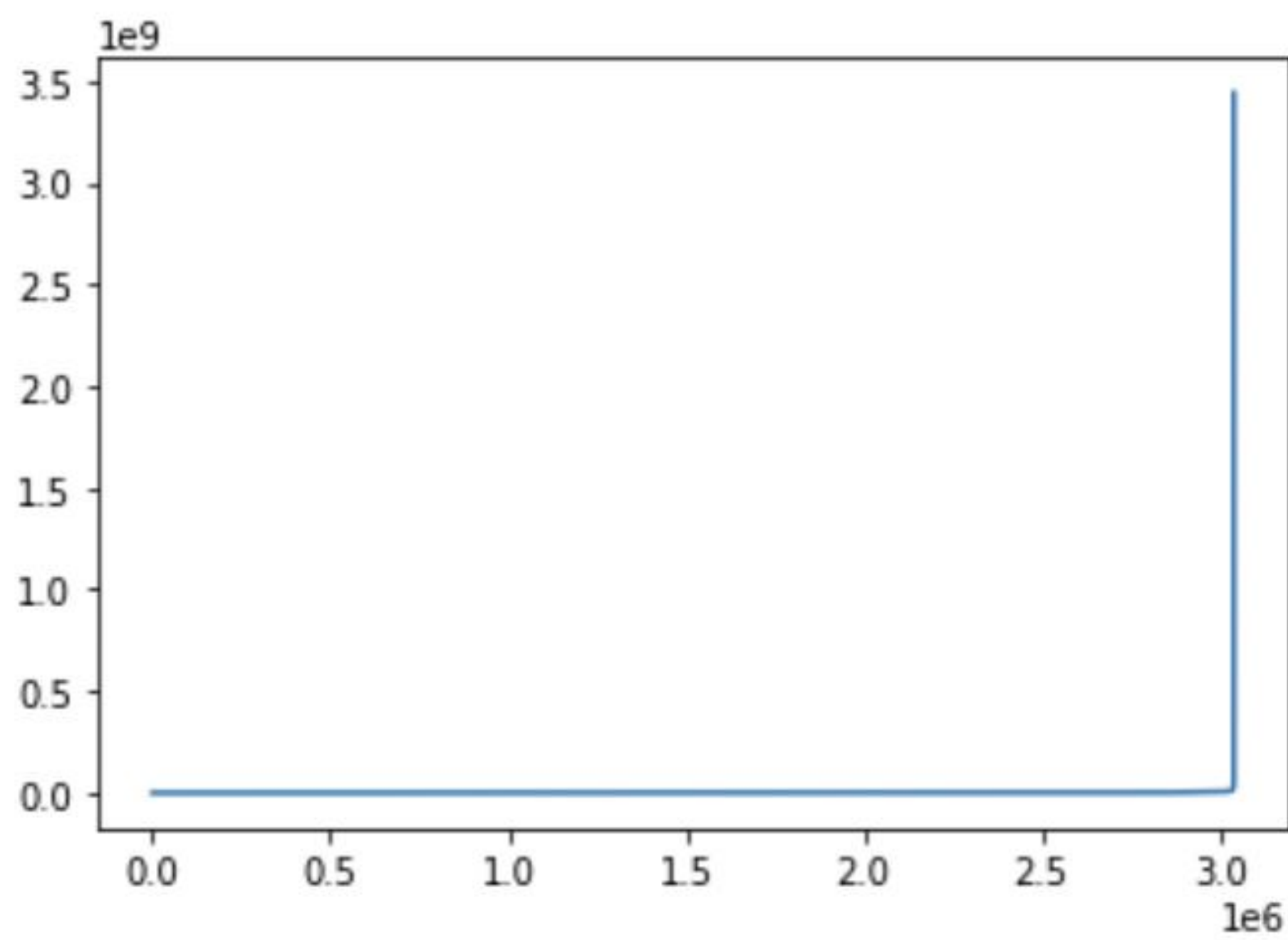
2. Feature Engineering

数据分布

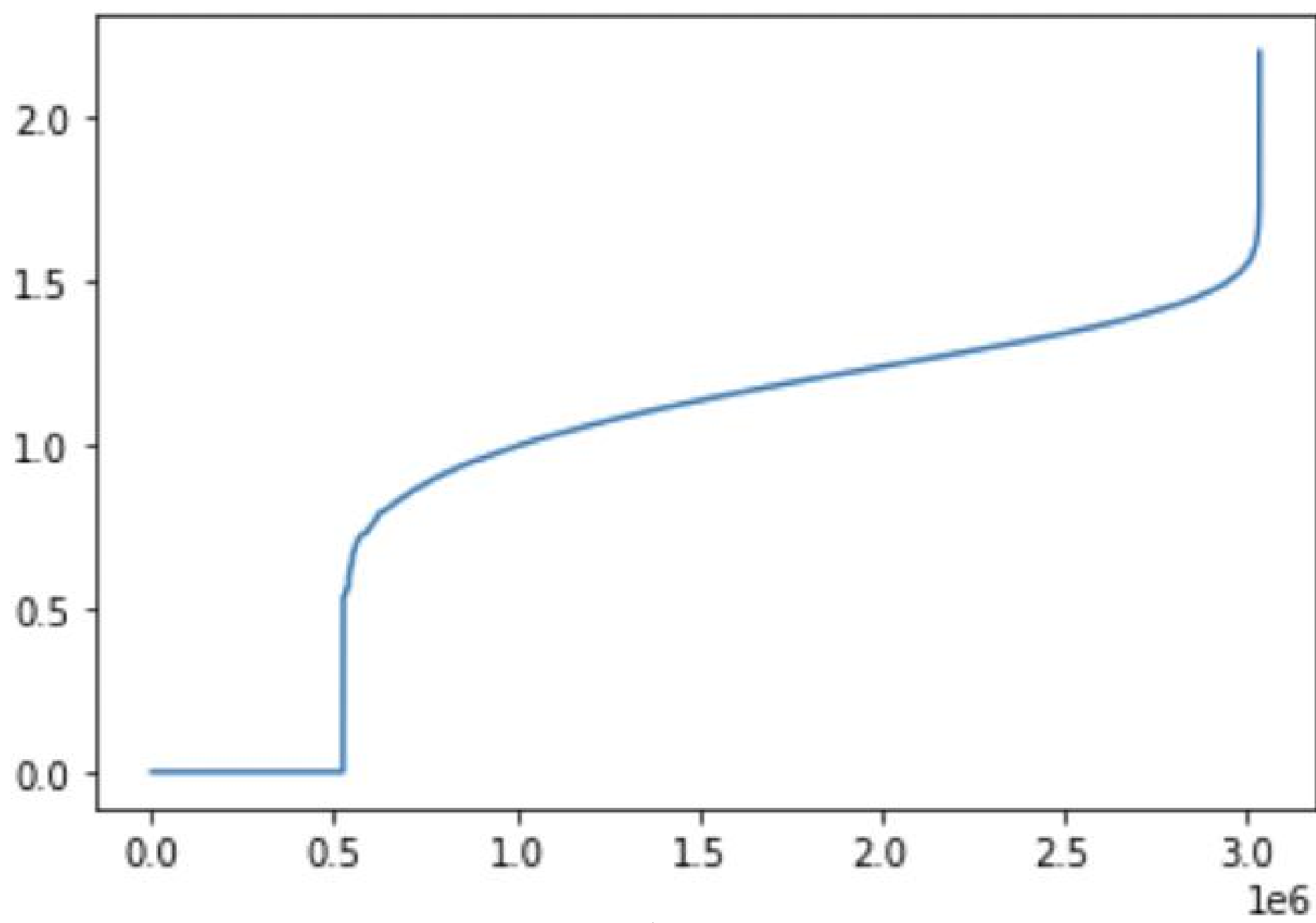
- 成交量数据：增大单位后取对数。
- 换手数据：减小单位后取对数。
- 价格数据：放缩。

数据分布

成交量数据分布



Before



After

数据分布

	date	sym	n_close	amount_delta	n_midprice	n_bid1	n_bsize1	n_bid2	n_bsize2
count	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06
mean	3.943130e+01	4.521368e+00	3.604229e-03	4.824170e+05	3.591785e-03	3.076102e-03	1.182146e-04	1.922592e-03	6.906901e-05
std	2.261346e+01	2.888636e+00	2.443552e-02	2.953133e+06	2.441072e-02	2.423171e-02	9.763794e-04	2.427663e-02	1.299123e-04
min	0.000000e+00	0.000000e+00	-1.001495e-01	0.000000e+00	-1.001495e-01	-1.001495e-01	0.000000e+00	-1.001495e-01	0.000000e+00
25%	2.000000e+01	2.000000e+00	-9.888752e-03	6.955000e+03	-9.722222e-03	-1.035197e-02	1.353481e-06	-1.154529e-02	2.471911e-06
50%	3.900000e+01	4.000000e+00	2.220577e-03	8.634500e+04	1.968504e-03	1.418440e-03	6.426969e-06	0.000000e+00	1.112887e-05
75%	5.900000e+01	7.000000e+00	1.472701e-02	4.016260e+05	1.471405e-02	1.411765e-02	3.168687e-05	1.304348e-02	5.421011e-05
max	7.800000e+01	9.000000e+00	1.005464e-01	3.440177e+09	1.005464e-01	1.005464e-01	2.775392e-02	9.987086e-02	9.185008e-03

Before

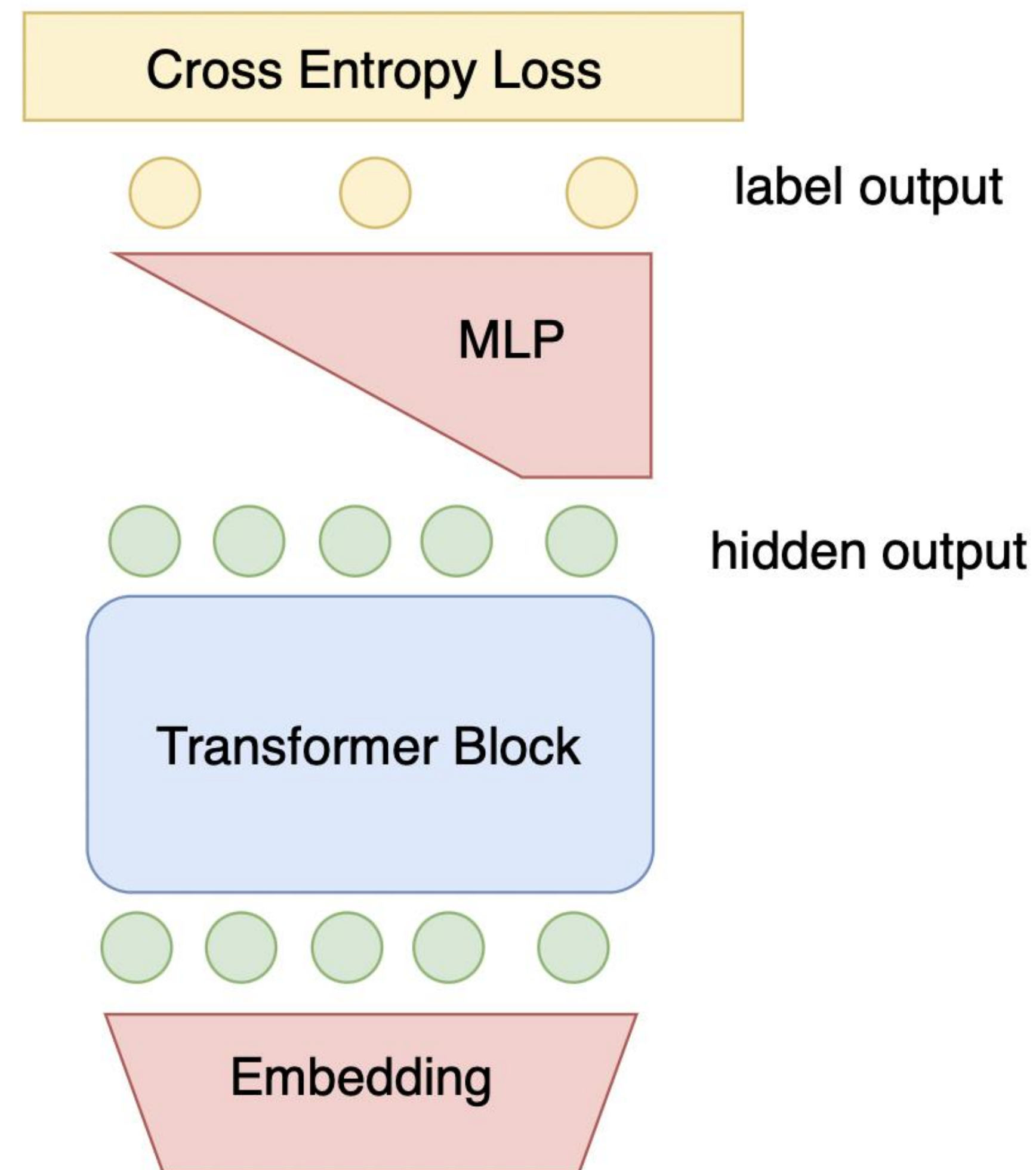
	date	sym	n_close	amount_delta	n_midprice	n_bid1	n_bsize1	n_bid2	n_bsize2
count	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06	3.040479e+06
mean	3.943130e+01	4.521368e+00	3.604229e-02	9.697427e-01	3.591785e-02	3.076102e-02	2.350361e-01	1.922592e-02	2.725919e-01
std	2.261346e+01	2.888636e+00	2.443552e-01	4.793614e-01	2.441072e-01	2.423171e-01	1.813635e-01	2.427663e-01	1.808014e-01
min	0.000000e+00	0.000000e+00	-1.001495e+00	0.000000e+00	-1.001495e+00	-1.001495e+00	0.000000e+00	-1.001495e+00	0.000000e+00
25%	2.000000e+01	2.000000e+00	-9.888752e-02	8.847360e-01	-9.722222e-02	-1.035197e-01	8.558953e-02	-1.154529e-01	1.244705e-01
50%	3.900000e+01	4.000000e+00	2.220577e-02	1.136612e+00	1.968504e-02	1.418440e-02	2.005118e-01	0.000000e+00	2.495588e-01
75%	5.900000e+01	7.000000e+00	1.472701e-01	1.290328e+00	1.471405e-01	1.411765e-01	3.486973e-01	1.304348e-01	4.011146e-01
max	7.800000e+01	9.000000e+00	1.005464e+00	2.195879e+00	1.005464e+00	1.005464e+00	1.023117e+00	9.987086e-01	9.125437e-01

After

3. Model

Structure

- 基于经典的时间序列模型Transformer
- 多任务模型，同时输出未来5个时间点的label
- K-dimensional Cross Entropy Loss



Transformer

- Positional Embedding描述时序的不对称性

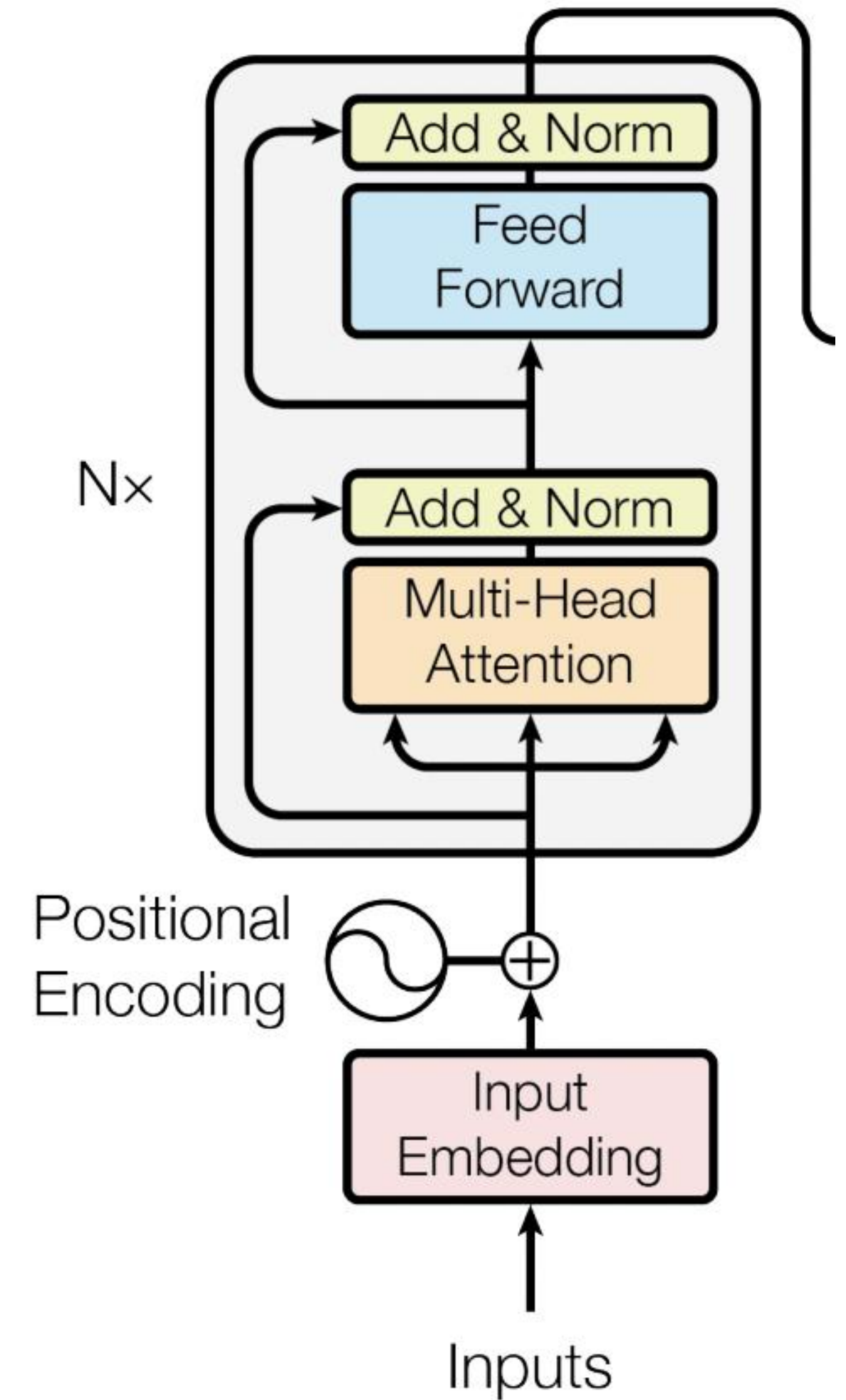
$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}})$$

- Self Attention捕捉时序上的微观特征
 - Attention Map计算时间序列的Covariance Matrix

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

- 残差连接防止梯度消失



Transformer Encoder Layer

