股指期货价差预测

代码：

https://github.com/nickhuangxinyu/zxrk

数据处理：

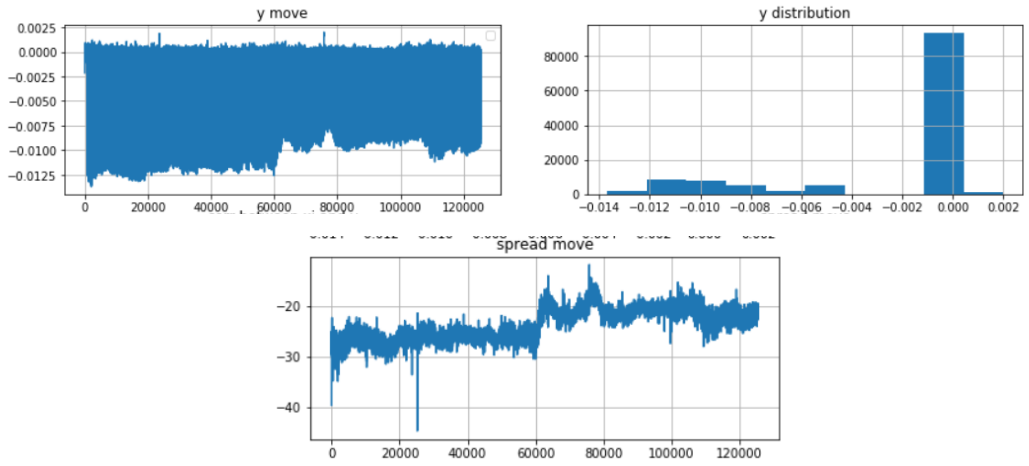
数据：

股指期货 IF 三天的 tick 数据

预测目标：

10 个 tick 之后主力和次主力收益率之差变动的方向，二分类问题

数据分布：



特征工程：

特征表达式：

Mid_spread = mp_x-mp_y

Mid_spread_gap = (mid_spread-mid_spread.rolling(t).mean())/mid_spread.rolling(t).std()

Wp = (bp0*bs0+ap0*as0)/(bs0+as0)

rwp = (bp0*as0+ap0*bs0)/(bs0+as0)

atp = turnover.diff(1)/volume.diff(1)

atp-mp_diff=(atp-mp).diff(t)

mid_macd：

Rwp-mp_diff= (rwp-mp).diff(t)

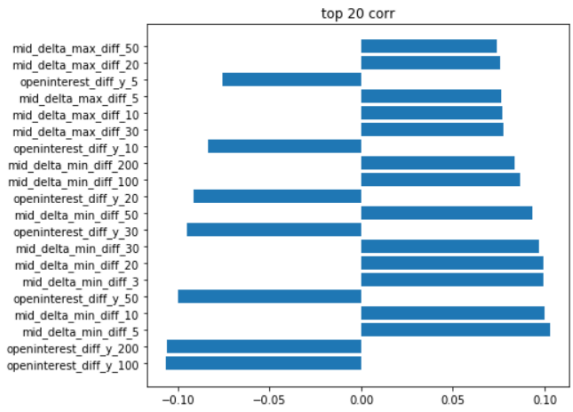
……

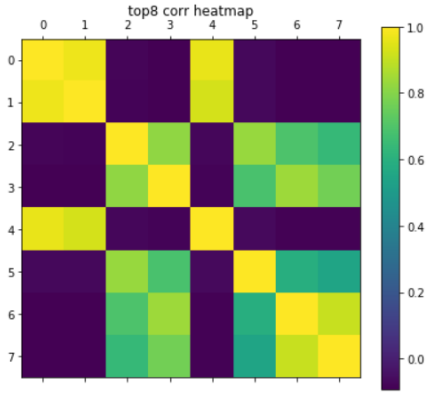
一共构造了 173 个特征

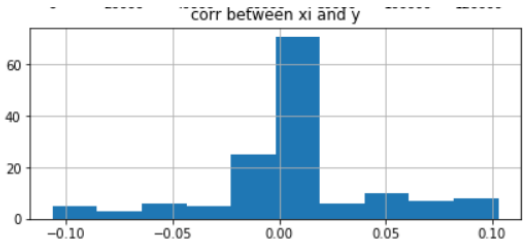
特征分布：（箱线图）：

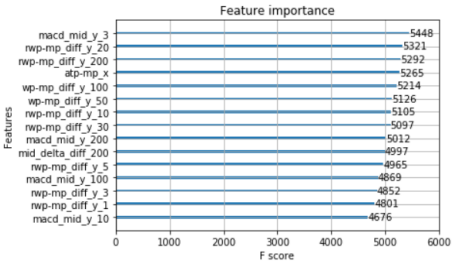
略，部分特征异常值比较严重，故数据归一化采用 RobustScaler 方式

特征相关性检验：









训练集测试集划分：

模型均采用 train:valid:test = 6:2:2 的比例随机划分，训练模型时，使用 train 数据，用 valid 的 early_stop 来防止过拟合，在 test 数据上验证结果。

模型：

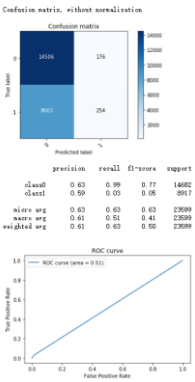
1. 线性模型 ElasticNet
   采用 grid_search，参数 alphas=[0.0,0.0001, 0.0005, 0.001, 0.01, 0.1, 1, 10], l1_ratio=[0.0, .01, .1, .5, .9, .99, 0.1]
2. Xgboost
   目标函数为 logistic
3. Mlp
   神经元个数为[50,30,20,2],激活函数采用 relu，输出层为 sigmoid，目标函数选取了 CategoricalCrossEntropy 和 focal_loss 两种，early_stop=5
4. lstm
   seq_length=32，用之前 32 个数据，来预测当前未来的目标值，过滤了开盘和收盘的数据，lstm 层神经元数量为 50，后面全连接结构，激活函数 relu

结果:

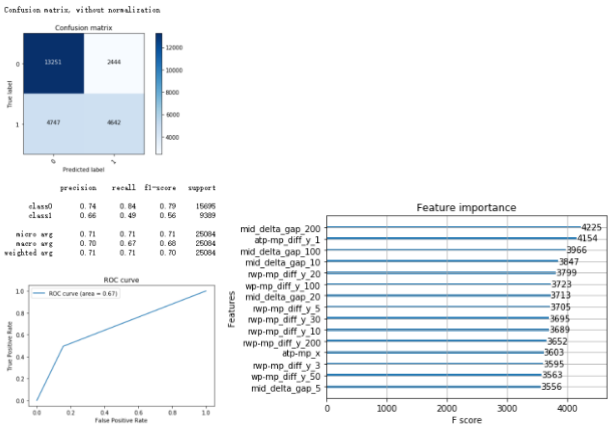
衡量标准：

Precision_rate, recall_rate, f1

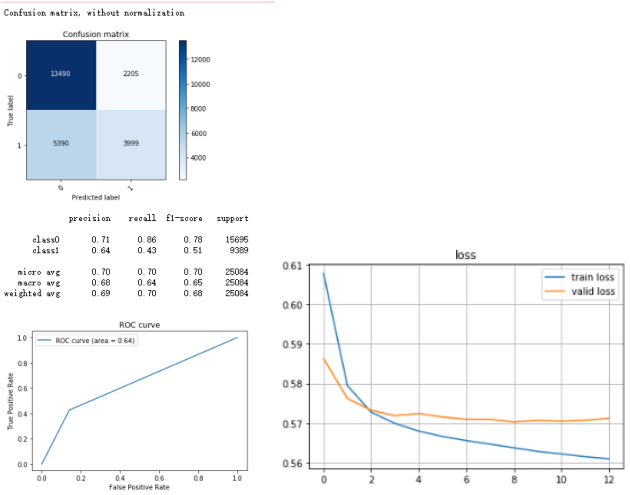
线性模型 ElasticNet



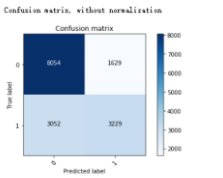
Xgboost



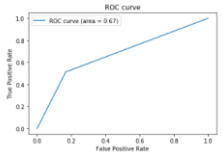
Mlp

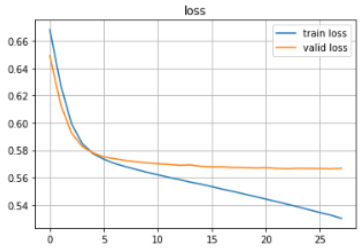


lstm



	precision	recall	f1-score	support
class0	0.73	0.83	0.77	9683
class1	0.66	0.51	0.58	6281
micro avg	0.71	0.71	0.71	15964
macro avg	0.69	0.67	0.68	15964
weighted avg	0.70	0.71	0.70	15964





结论:

	Precision_rate		Recall_rate		F1		Auc
LR	0.63	0.59	**0.99**	0.03	0.77	0.05	0.51
XGBOOST	**0.74**	0.66	0.84	0.49	**0.79**	0.56	0.67
MLP	0.71	0.64	0.86	0.43	0.78	0.51	0.64
LSTM	0.73	**0.66**	0.83	**0.51**	0.77	**0.58**	**0.67**

由于特征工程中找到的特征和目标值的线性相关性较低，线性模型表现不好，倾向于偷懒给出同一预测

XGBOOT 和 LSTM 效果比较好，LSTM 在正例上的召回率较高。