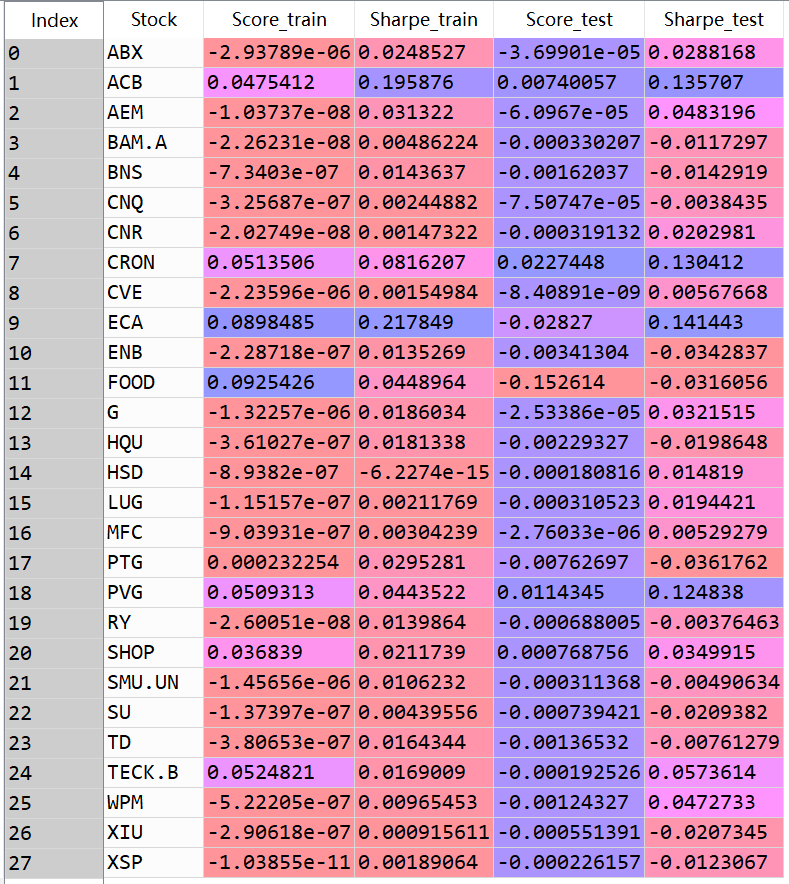
1. RF, n = 100, max\_depth = 2

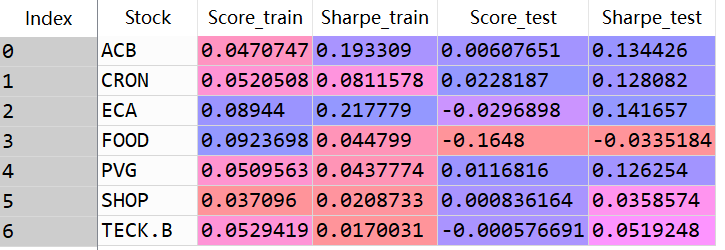


Select the ones with train score > 0.01:

ACB, CRON, ECA, FOOD, PVG, SHOP, TECK.B are good

1. Look at each’s feature importance, select the ones with score>0.01

Refit random forest



1. The result seems to be not promising, so we try to tune the parameter.

Unfortunately, as we tune the parameter, we find it extremely hard to improve the performance on test set score. “ACB”, “CRON”, “PVG” always seem to have better performance than the other stocks as the parameter changes.

Final model is:

model\_param = {'max\_depth': 2, 'random\_state': 0, 'n\_estimators': 100}

ACB, 'mid\_momentum\_20ord', 'transaction\_spread\_5ord', **'volum\_imbalance'**,

CRON, 'mid\_momentum\_10ord', 'mid\_momentum\_30s', **'mid\_momentum\_50ord'**,

'smart\_price\_momentum\_50ord', 'smart\_price\_snapshot', 'spread\_10s',

**'spread\_30s'**, 'spread\_5s', 'spread\_diff\_10s', 'spread\_diff\_30s',

'trade\_sign\_10ord', 'trade\_sign\_2s', 'transaction\_spread\_10ord',

**'volum\_imbalance'**

PVG, 'mid\_momentum\_10ord', 'mid\_momentum\_10s', 'mid\_momentum\_30s',

**'mid\_momentum\_50ord'**, **'mid\_snapshot'**, 'smart\_price\_momentum\_10ord',

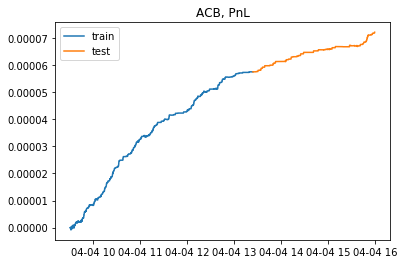
'smart\_price\_momentum\_10s', 'smart\_price\_momentum\_20ord',

'smart\_price\_momentum\_30s', 'smart\_price\_momentum\_50ord',

'smart\_price\_momentum\_5ord', 'smart\_price\_snapshot', **'spread\_10s'**,

'spread\_30s', 'spread\_50ord', 'transaction\_spread\_10ord',

'volum\_imbalance'

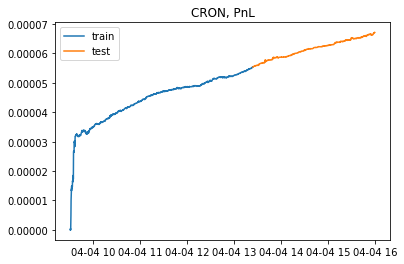


('Train Score: ', 0.04592)

('Train Sharpe: ', 0.19169)

('Test Score: ', 0.00549)

('Test Sharpe: ', 0.1336)



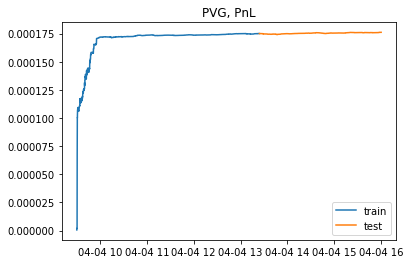
Cron:

('Train Score: ', 0.04081)

('Train Sharpe: ', 0.07381)

('Test Score: ', 0.01828)

('Test Sharpe: ', 0.14117)



('Train Score: ', 0.04106)

('Train Sharpe: ', 0.04137)

('Test Score: ', -0.00015)

('Test Sharpe: ', 0.02286)