1.程序介绍

（1）仿真程序需要在linux系统并且是python3.7的环境中才能运行起来，因为.so文件是支撑整个仿真环境的核心。

（2）提供了两个策略，分别是规则和随机，run\_log.py中可以对策略进行配置，使用--my\_ai来进行解析。

（3）环境目前来看是单智能体，应该和时序相关。

（4）主程序是run\_log.py，可以根据这个实现自己的AI代码训练和评估。

2.观察量

        [{'observation': {'serverTime': 93004406.0, 'eventTime': 93004990.0, 'code': 2.0, 'signal0': -0.5843887380437967,

        'signal1': -0.69046247746311, 'signal2': -1.693096976594504, 'ap0': 4598.62, 'bp0': 4596.55, 'av0': 8.0,

        'bv0': 3.0, 'ap1': 4600.0, 'bp1': 4594.411, 'av1': 22.0, 'bv1': 4.0, 'ap2': 4604.599999999999,

        'bp2': 4594.365, 'av2': 1.0, 'bv2': 2.0, 'ap3': 4605.658, 'bp3': 4594.3189999999995, 'av3': 7.0,

        'bv3': 1.0, 'ap4': 4606.900000000001, 'bp4': 4586.269, 'av4': 6.0, 'bv4': 5.0, 'code\_net\_position': 0,

        'ap0\_t0': 4597.469999999999}, 'new\_game': False}]

如图所示是观察量的结构，详细信息参考这个[科目详情 - 及第 Jidi (jidiai.cn)](http://www.jidiai.cn/env_detail?envid=104)，

Code表示股票代码；signal是不同预期时间的涨跌；bv是买入量；av是卖出量；ap是卖出价格；bp是买入价格。不同的后缀表示不同的档次，都可以直接作为观察量的特征。（第一档的买入价会是所有买家中出价最高的，而第一档的卖出价是所有卖家中出价最低的。更高档次的买入价较低，而更高档次的卖出价则较高。这些信息对于理解市场供需平衡、价格走势和潜在的交易机会非常重要）

3.控制量

        joint\_act: [[[0, 0, 1], array([17.525919], dtype=float32), array([8072.1045], dtype=float32)]]

控制量第一个one-hot编码可取0， 1， 2，分别代表买入、什么都不做和卖出，初步判断[0,0,1]为卖出。‘volume’和‘price’分别指订单中的交易量（单位：手数）和价格（单位：元/股）

4.决策空间约束

[科目详情 - 及第 Jidi (jidiai.cn)](http://www.jidiai.cn/env_detail?envid=104)参考这个页面，约束决策空间。



5、方案：

初步判断PPO，不过最后一层决策网络分三个，分别用于表现离散动作、连续动作、连续动作。可能和过去的时序相关。