量化 live 2 笔记

策略和 infastructure 的开发是合二为一的

定义类的例子 (java)

```java
※ public class Bin {
    public calendar openTime;
    public calendar closeTime;
    public double openBid;
    public double highBid;
    public double lowBid;
    public double closeBid;

    public double openAsk;
    public double highAsk;
    public double lowAsk;
    public double volumn;
    public double pnl;
    public int length; }
```

```java
public class Trade {

    private UUID uuid;
    private String instrument;

    private double entryPrice;
    private double exitPrice;

    private double stoploss Price;
    private double profitTargetPrice;

    private double commissionrate;
    private double commission;

    private int quantity;

    private Calendar entryTime;
    private Calendar exitTime;

    private DirectionType direction;
}
```
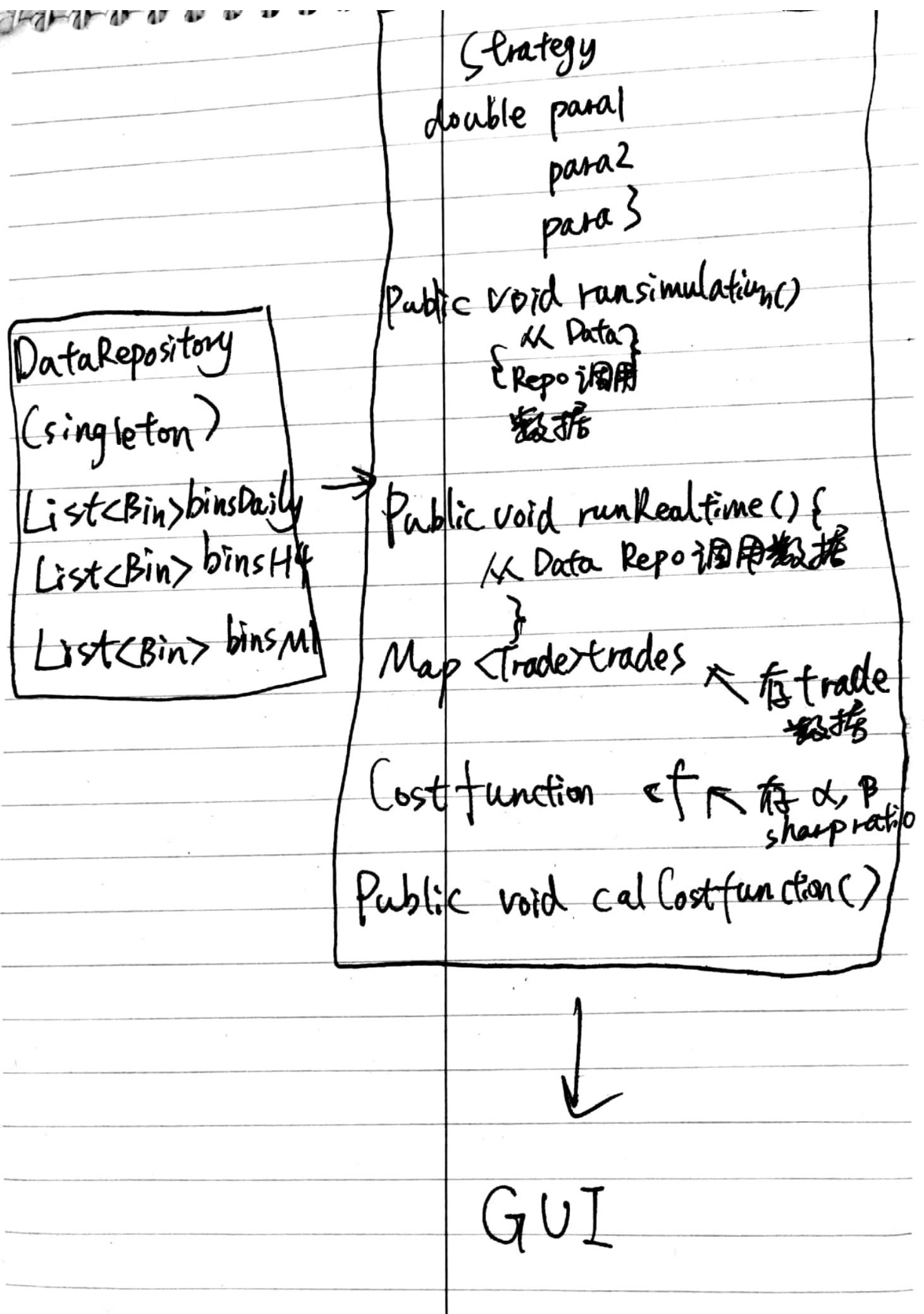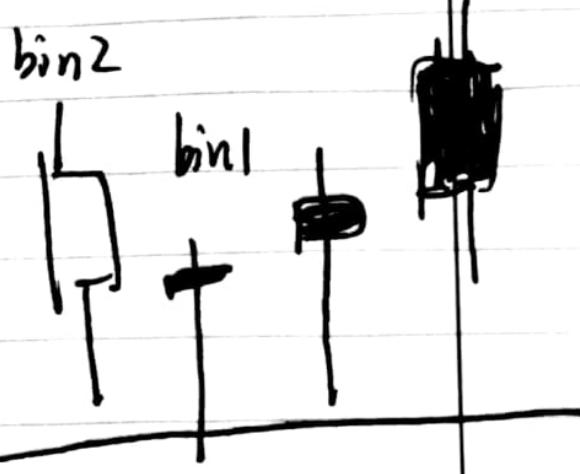
Strategy

double para1
       para2
       para3

Public void runsimulation()
        // Data?
        { Repo 调用
        数据

Public void runRealtime() {
   // Data Repo 调用数据
        }

Map <Trade> trades   ← 存 trade
                       数据

Cost function ←f ← 存 α, β
                      sharp ratio

Public void cal Costfunction()

DataRepository
(singleton)

List<Bin>binsDaily
List<Bin> binsH
List<Bin> binsM

GUI

```
for binsDaily{
    for(Bin bin:binsM1) {

        if(达到入场价){

            Trade trade = new Trade()
            Trades.put(trade); }

        For(trades){

            if(打止赢,打止损){

                Trade.close()
            }
        }
    }
}
```
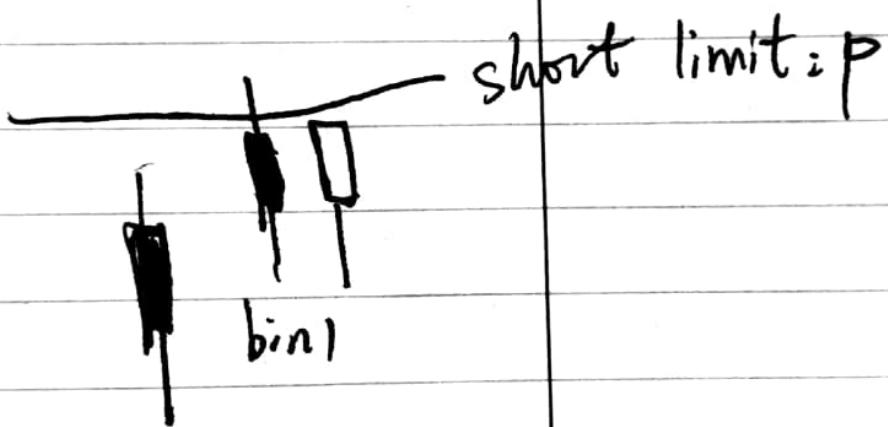
bin2

bin1

long limit: p

entry condition: close ask > p

bin1. low ask <= p && bin2. ~~close~~

做多

short limit: p

bin1

bin2

entry ~~position~~ condition:

bin1. high bid >= p

&& bin2. close bid < p

做空

注: 假设没有滑点.

cost function 需要了解！

比如 Sharp, soto sotino
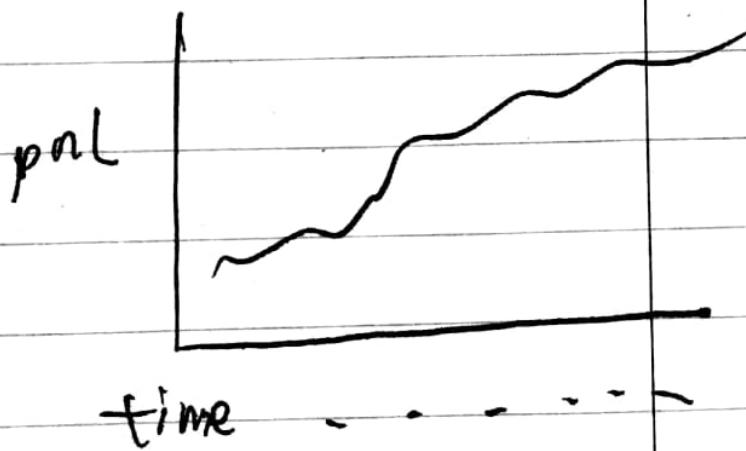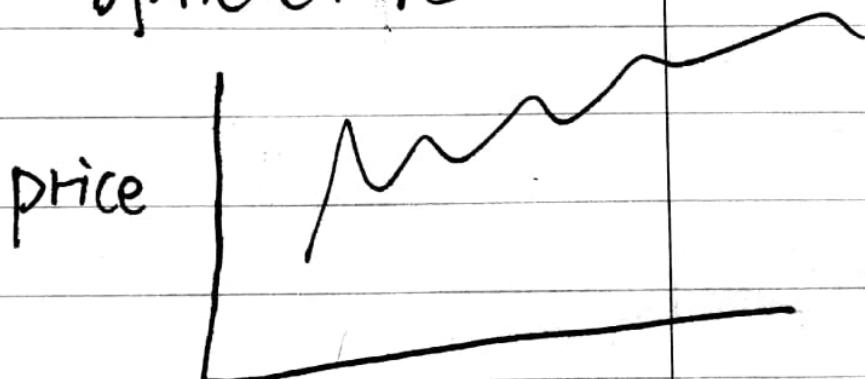
胜率, 盈亏比, 最大回测,

最大连续亏损数, 交易次数,

交易频率, 平均持仓时间

可视化例子

price

pnl

time

gui 须贴
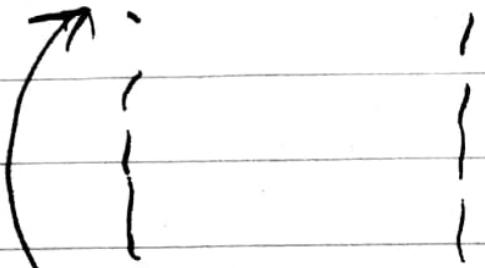
uuid	entry time	exit time	entryPrice	exitprice
235	08/05/2017	0.7	0.8	0.6
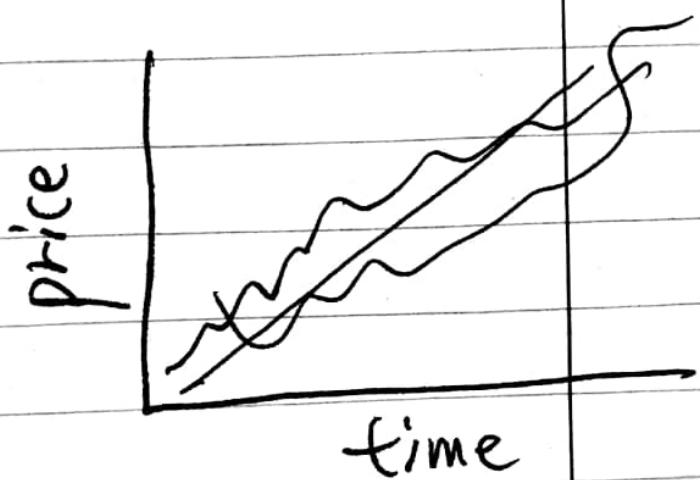
点击 event → 显示交易 ask, bid 细节

图片最好!

多个交易的图最好可以 同时显示

可加一些技术指标, 如MA, kdj

（回测）结果需要检查！

需要实现一个 ~~函数~~ 调参函数

Parameters | Strategies

设置 ~~step~~ step { para1, para2 ——→ Strategies
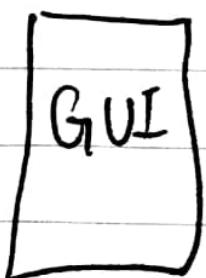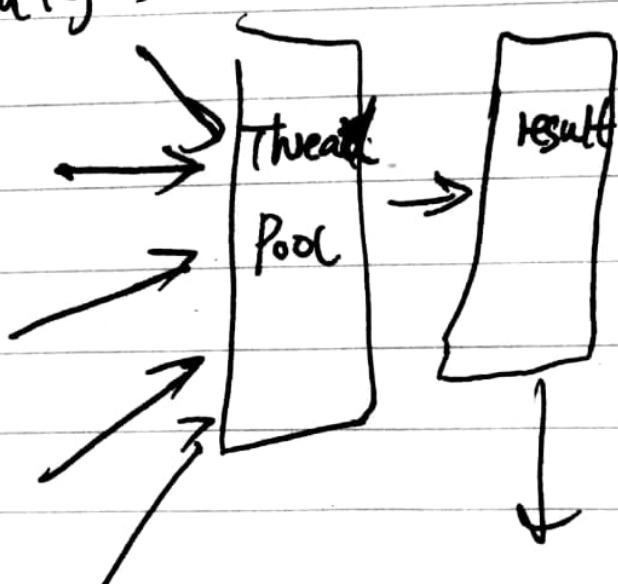
p2, q2 ——run-Simulation——→ S

p3, q3 ——→ S

p4, q4 ——→ S

pn, qn ——→ S 进

Thread Pool → result → GUI

多线程。（可能用 GPU??）

我的感觉: （需要学 CUDA 可能）

运算结果需要存在表里，然后排序

比如：

profit	sharpe	maxdd	mwl	win rate	trade no

total profit	total loss	profit factor

策略开发的两个方向

① 先有想法，再开发

② 先挖掘，再开发