

法律声明

本课件包括：演示文稿，示例，代码，题库，视频和声音等，小象学院拥有完全知识产权的权利；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或机构不得盗版、复制、仿造其中的创意，我们将保留一切通过法律手段追究违反者的权利。



关注 小象学院

第八课

技术型和基本面因子的编写

系统化构建量化交易体系：

模块3：量化交易策略逻辑的深入讨论

内容介绍



使用财务评分卡框架实现盈利能力及质量评价因子

上市公司对财报中利润的操纵

改进型ROE的实现

Boll中轨支撑反弹的实现

“次日机会”量价因子的实现

使用财务评分卡框架实现盈利能力及质量评价因子

盈利能力及质量评价因子的制作

财务分析分类

风险

运营效率

盈利能力及质量

前景与评估

风险

- ☐ 资产负债率
- ☐ 净资产负债率
- ☐ 流动比率
- ☐ 速动比率
- ☐ ...

运营效率

- ☐ 总资产周转率
- ☐ 固定资产周转率
- ☐ 存货周转率
- ☐ 应收账款周转率
- ☐ ...

盈利能力及质量

- ☐ 净资产收益率
- ☐ 总资产收益率
- ☐ 每股收益
- ☐ 销售收现比率
- ☐ 净利润现金含量比率
- ☐ 毛利率、净利率
- ☐ ...

前景与评估

- ☐ 营业收入增长率
- ☐ 净利润增长率
- ☐ 总资产增长率
- ☐ 净资产增长率
- ☐ ...

财务评分卡框架

类型	权重	总分	财务指标	分数
风险	10%	100	资产负债率	40
			流动比率	60
运营效率	25%	100	总资产周转率	50
			存货周转率	50
盈利能力及质量	40%	100	净资产收益率	20
			每股收益	30
			销售收现比率	20
			净利润现金含量比率	30
前景与评估	25%	100	营业收入增长率	50
			净利润增长率	50

盈利能力及质量评价因子

因子评分 = 风险得分*10% + 运营效率得分*25% + 盈利能力及质量得分*40% + 前景与评估得分*25%

	总评分排名占比	评级
财务评级	前10%	A级
	前10%—30%	B级
	中30%—60%	C级
	后60%—80%	D级
	后80%—100%	E级

财务指标计算-资产负债率

财务指标	指标值	得分
资产负债率	40%—60%	40
	30%—40%	30
	0—30%	20
	60-80%	10
	80%—100%	5

财务指标计算-流动比率

财务指标	指标值排名占比	得分
流动比率	前10%	60
	前10%—30%	50
	中30%—60%	40
	后60%—80%	25
	后80%—100%	10

财务指标计算-总资产周转率

财务指标	指标值排名占比	得分
总资产周转率	前10%	50
	前10%—30%	40
	中30%—60%	30
	后60%—80%	20
	后80%—100%	10

财务指标计算-存货周转率

财务指标	指标值排名占比	得分
存货周转率	前10%	50
	前10%—30%	40
	中30%—60%	30
	后60%—80%	20
	后80%—100%	10

财务指标计算-净资产收益率

财务指标	指标值排名占比	得分
净资产收益率	前10%	20
	前10%—30%	15
	中30%—60%	10
	后60%—80%	5
	后80%—100%	1

财务指标计算-每股收益

财务指标	指标值排名占比	得分
每股收益	前10%	30
	前10%—30%	25
	中30%—60%	15
	后60%—80%	10
	后80%—100%	5

财务指标计算-销售收现比率

财务指标	指标值	得分
销售收现比率	大于等于1	20
	小于1	10

财务指标计算-净利润现金含量比率

财务指标	指标值排名占比	得分
净利润现金含量比率	前10%	30
	前10%—30%	25
	中30%—60%	15
	后60%—80%	10
	后80%—100%	5

财务指标计算-营业收入增长率

财务指标	指标值排名占比	得分
营业收入增长率	前10%	50
	前10%—30%	40
	中30%—60%	30
	后60%—80%	20
	后80%—100%	10

财务指标计算-净利润增长率

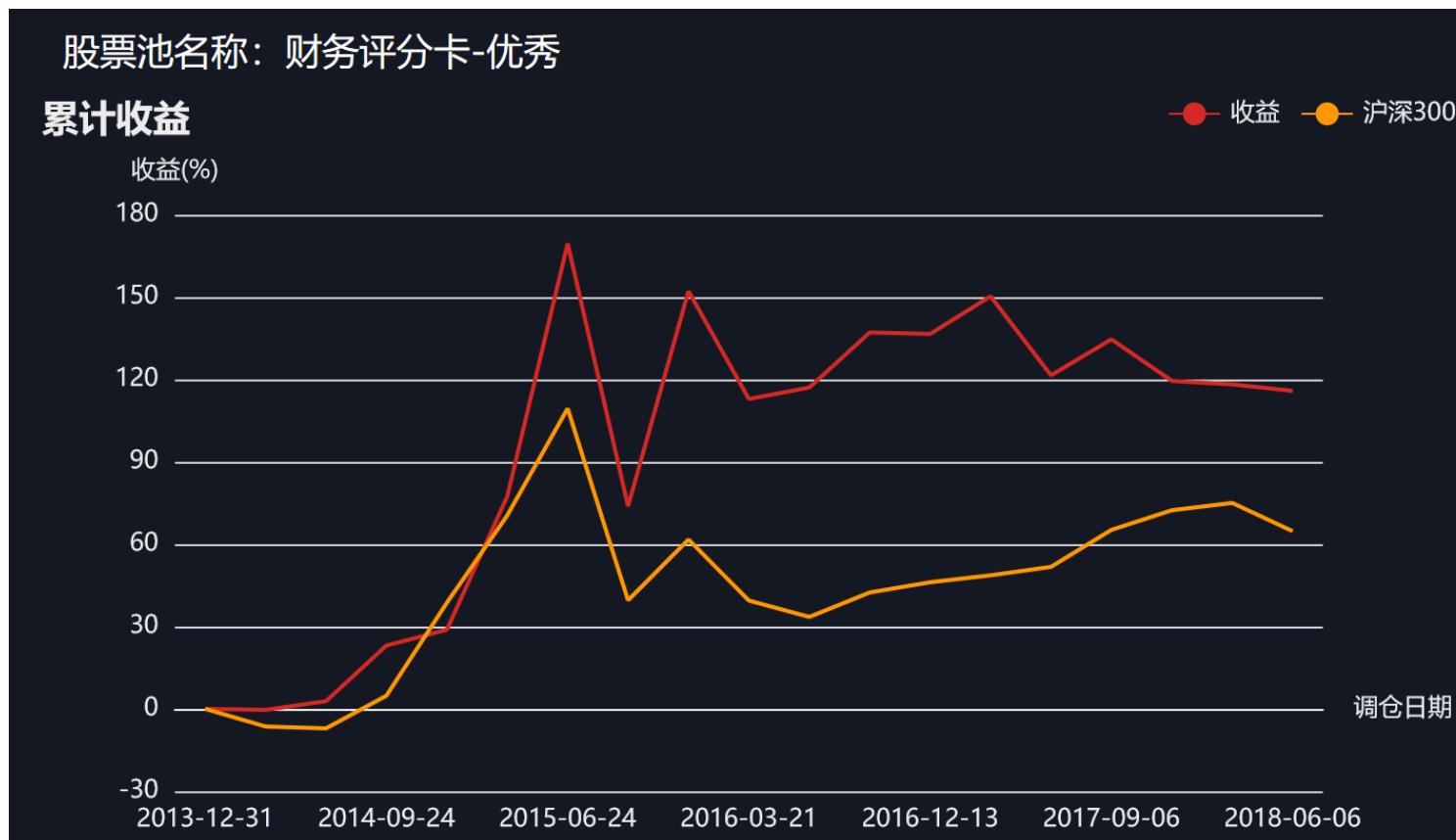
财务指标	指标值排名占比	得分
净利润增长率	前10%	50
	前10%—30%	40
	中30%—60%	30
	后60%—80%	20
	后80%—100%	10

实例-恒生电子

资产负债率	35%	30
流动比率	排名35%	40
得分	$(30+40)*0.1=7$ 分	
总资产周转率	排名25%	40
存货周转率	排名40%	30
得分	$(40+30)*0.25=17.5$ 分	
净资产收益率	排名50%	10
每股收益	排名32%	15
销售收现比率	大于1	20
净利润现金含量比率	排名46%	15
得分	$(10+15+20+15)*0.4=24$ 分	
营业收入增长率	排名32%	30
净利润增长率	排名70%	20
得分	$(30+20)*0.25=12.5$ 分	
总得分	$7+17.5+24+12.5=61$ ，排名35%，C级	

股票池实例

财务评级：A级 股票池调整周期：60个交易日



使用财务评分卡框架实现盈利能力及质量评价因子

上市公司对财报中利润的操纵

如何调节**净利润**？

+

- 营业收入
- 公允价值变动损益
- 投资收益
- 营业外收入

-

- 营业成本
- 营业税金及附加
- 销售费用
- 管理费用
- 财务费用
- 资产减值损失
- 营业外支出
- 所得税费用

调节方法

调节方法列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

其他

调节方法

调节 方法 列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

定义

会计估计

- 依据信息对不确定事项作出判断
- 可依据信息发生改变，则可变更会计估计

会计政策

- 会计核算及编制会计报表采用的方法
- 提供更合适的会计信息，则可变更会计政策

应收坏账准备

变更前公司采用账龄分析法计提坏账准备如下：

天沃科技

账龄	应收账款计提比例（%）	其他应收款计提比例（%）
1 年以内（含 1 年）	3	3
1—2 年	10	10
2—3 年	30	30
3—4 年	50	50
4—5 年	80	80
5 年以上	100	100

变更后公司采用账龄分析法计提坏账准备如下：

账龄	应收账款计提比例（%）	其他应收款计提比例（%）
1 年以内（含 1 年）	3	3
1—2 年	5	5
2—3 年	20	20
3—4 年	40	40
4—5 年	60	60
5 年以上	100	100

固定资产折旧

- ❑ 固定资产折旧方法变更可以影响变更以后每年的利润
- ❑ 寅吃卯粮：前几年增加利润，后几年减少利润

鹏博士

固定资产类别	调整前折旧年限（年）	调整后折旧年限（年）
线路资产	8	8-15

减少2017年折旧金额3.54亿，增加2017年净利润3亿元

投资性房地产的后续计量

- 企业在资产负债表日对投资性房地产进行后续计量所采用的会计处理。例如企业对投资性房地产的后续计量是采用成本模式，还是采用公允价值模式。模式的改变会影响会计处理方式，从而影响净利润。
- 中安消：2016年增加了2.6亿的净利润

其他方法

- 合同完工进度的确定：延迟转固
- 长期股权投资：成本法/权益法
- 借款费用的处理：资本化/费用化
- 研发阶段无形资产的确认：资本化/费用化

调节方法

调节 方法 列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

资产减值损失

- 因资产的账面价值高于其可收回金额而造成的损失
- 会计期末，企业要复核账上资产的账面价值，进行计提减值

提前足额计提

柳州化工 - 洗大澡

2016 年度公司计提各类资产减值准备共计 23,518.69 万元人民币，具体计提项目明细如下：

项目	2016 年度计提金额（万元）
固定资产减值准备	19,487.72
坏帐准备	328.79
存货跌价准备	3,702.18
合计	23,518.69

调节方法

调节 方法 列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

投资收益

- 企业合并准则中，通过多次交易分步实现的企业合并中，在合并财务报表中，对于购买日之前持有的被购买方的股权，应当按照该股权在购买日的公允价值进行重新计量，公允价值与其账面价值的差额计入当期投资收益。
- 股权增值

举例

- 2016年3月公告，星辉娱乐以参考96,951.40万元购买趣丸网络67%股权，趣丸网络的估值14.3亿元，由于此前星辉娱乐持有趣丸网络33%的股权，且股权取得较早，成本较低，因此本次确认了高达2.7亿的投资收益
- 投资收益占当年净利润的70%

调节方法

调节 方法 列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

商誉

- 商誉通常是指企业在同等条件下，能获得高于正常投资报酬率所形成的价值
- 企业合并所形成的商誉，应当在每年年度终了进行减值测试
- 利用商誉减值洗大澡

举例

□ 三泰电子

- 公司使用人民币7.5亿元收购烟台伟岸信息科技有限公司100%股权，该收购事项形成商誉67,755.03万元
- 公司因当年扭亏无望，便通过大额计提商誉减值6亿来为公司“洗大澡”，为未来业绩增长铺平道路

调节方法

调节 方法 列表

变更会计估计/会计政策

资产减值损失

投资收益

商誉

关联交易

关联交易

□ 关联交易是指企业关联方之间的交易，
是公司运作中经常出现的而又易于发生
不公平结果的交易

关联交易的作用

- 价格公允比较难以判断，包括成本和收入两个方向
- 隐蔽的关联方不容易察觉，容易产生关联交易非关联化

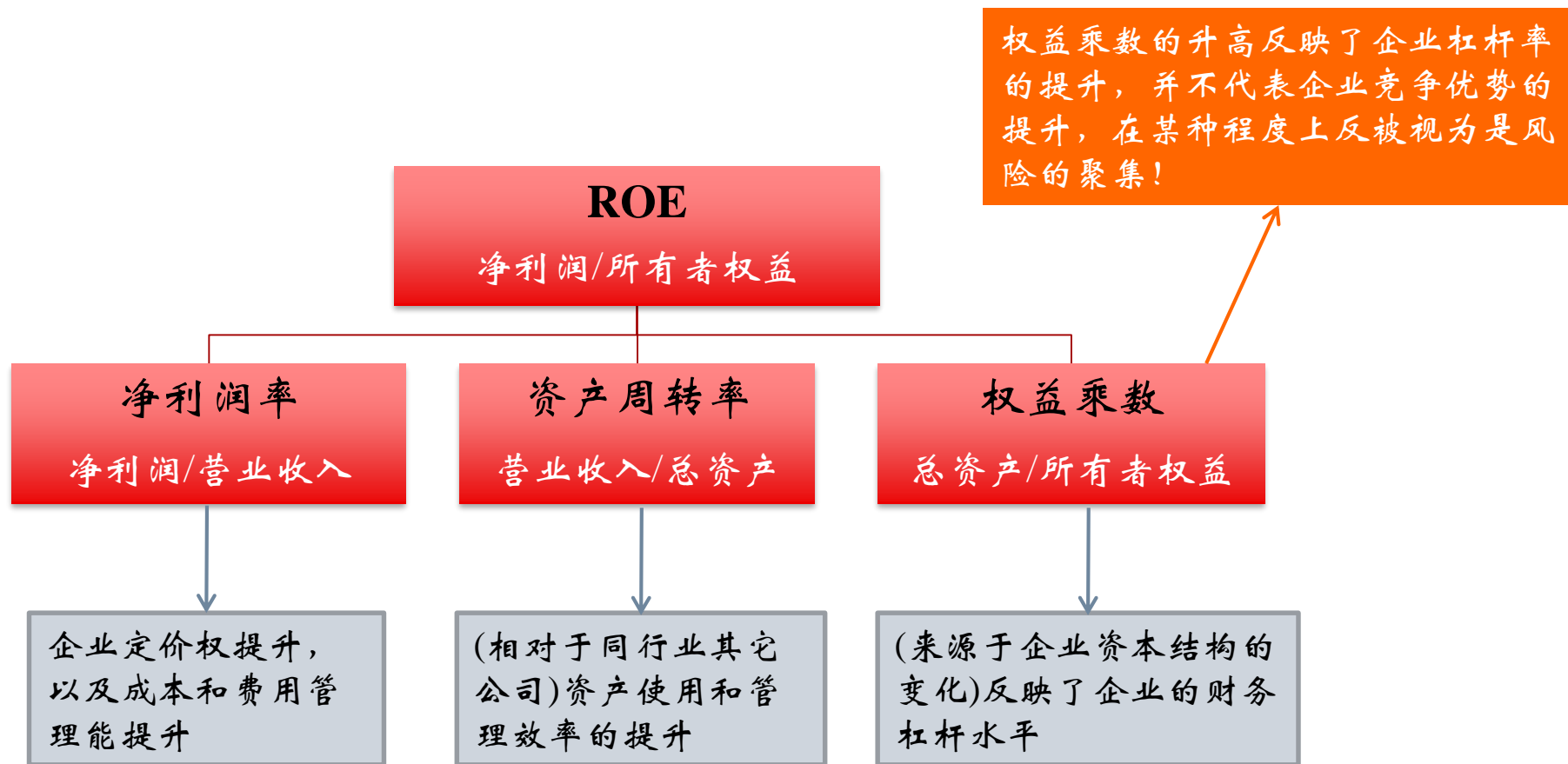
举例

□ 神雾环保

- 2016年，乌海神雾煤化科技有限公司和新疆胜沃能源是神雾环保最大的两个客户，2016年承接的销售额达到23亿，占神雾环保当年总销售额的74%
- 2017年，公司披露的关联方收入约5.65亿元，占当期收入的25%，加上未披露的乌海神雾的8.67亿，关联方销售合计也达到70%

改进型ROE的实现

ROE的杜邦分解



改进型的计算公式

$$\text{ROE} = \text{净利润} / \text{所有者权益}$$

$$= \text{净利润} / \text{营业收入} \times \text{营业收入} / \text{总资产} \times \text{总资产} / \text{所有者权益}$$

$$= \text{净利润率} \times \text{资产周转率} \times \text{权益乘数}$$

$$\text{改进型ROE} = \text{净利润率} \times \text{资产周转率}$$

$$= \text{ROE} / \text{权益乘数}$$


```

class ImprovedRoeFactor(BaseFactor):
    def __init__(self):
        BaseFactor.__init__(self, 'improved_roe')

    def compute(self, begin_date, end_date):
        codes = get_all_codes()

        all_dates = get_trading_dates(begin_date=begin_date, end_date=end_date)

        for code in codes:
            update_requests = []
            for date in all_dates:
                lrb = DB_CONN['CWBB_LRB'].find_one(
                    {'code': code, 'announced_date': {'$lte': date}, 'report_date':
{'$regex': '\d{4}-12-31$'}},
                    sort=[('announced_date', DESCENDING)],
                    projection={'parentnetprofit': True}
                )

                # 如果没有利润表信息，则跳过
                if lrb is None:
                    continue

                zcfzb = DB_CONN['CWBB_ZCFZB'].find_one(
                    {'code': code, 'announced_date': {'$lte': date}, 'report_date':
{'$regex': '\d{4}-12-31$'}},
                    sort=[('announced_date', DESCENDING)],
                    projection={'sumasset': True})

```

改进型的ROE

```
if zcfzb is None:  
    continue
```

```
improved_roe = round(lrb['parentnetprofit'] / zcfzb['sumasset'], 2)
```

```
update_requests.append(UpdateOne(  
    {'code': code, 'date': date},  
    {'$set': {'code': code, 'date': date, 'roe': improved_roe}},  
    upsert=True))
```

改进型的ROE

```
if len(update_requests) > 0:  
    save_result = self.collection.bulk_write(update_requests, ordered=False)  
    print('股票代码: %s, 因子: %s, 插入: %4d, 更新: %4d' %  
          (code, self.name, save_result.upserted_count,  
            save_result.modified_count), flush=True)
```

BOLL中轨支撑反弹的实现

Boll

□ 基于统计学的标准差原理

□ 三条轨道

■ 上轨：压力线

■ 中轨：价格均线

■ 下轨：支撑线

计算公式：价格均值和标准差

$$MA = \frac{1}{N} \sum_{i=0}^N CLOSE_i \quad (1)$$

N=20

$$STD = \sqrt{\frac{1}{N} \sum_{i=0}^N (CLOSE_i - MA)^2} \quad (2)$$

计算公式

$$MB = \frac{1}{N-1} \sum_{i=0}^{N-1} CLOSE_i$$

中轨

盘中实时计算，当日收盘价不稳定，所以计算的是前一天的均价

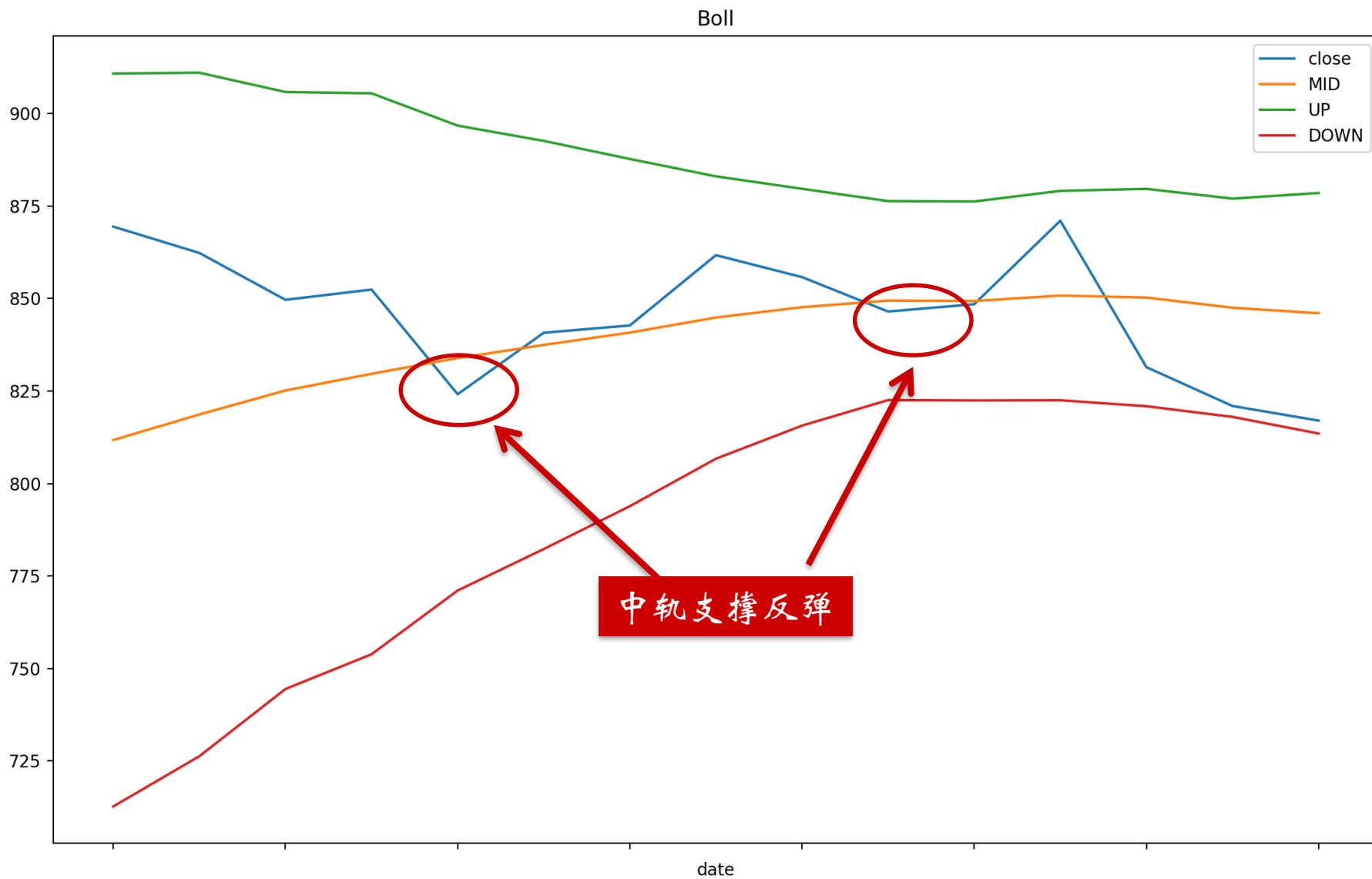
$$UP = MB + k * STD$$

上轨

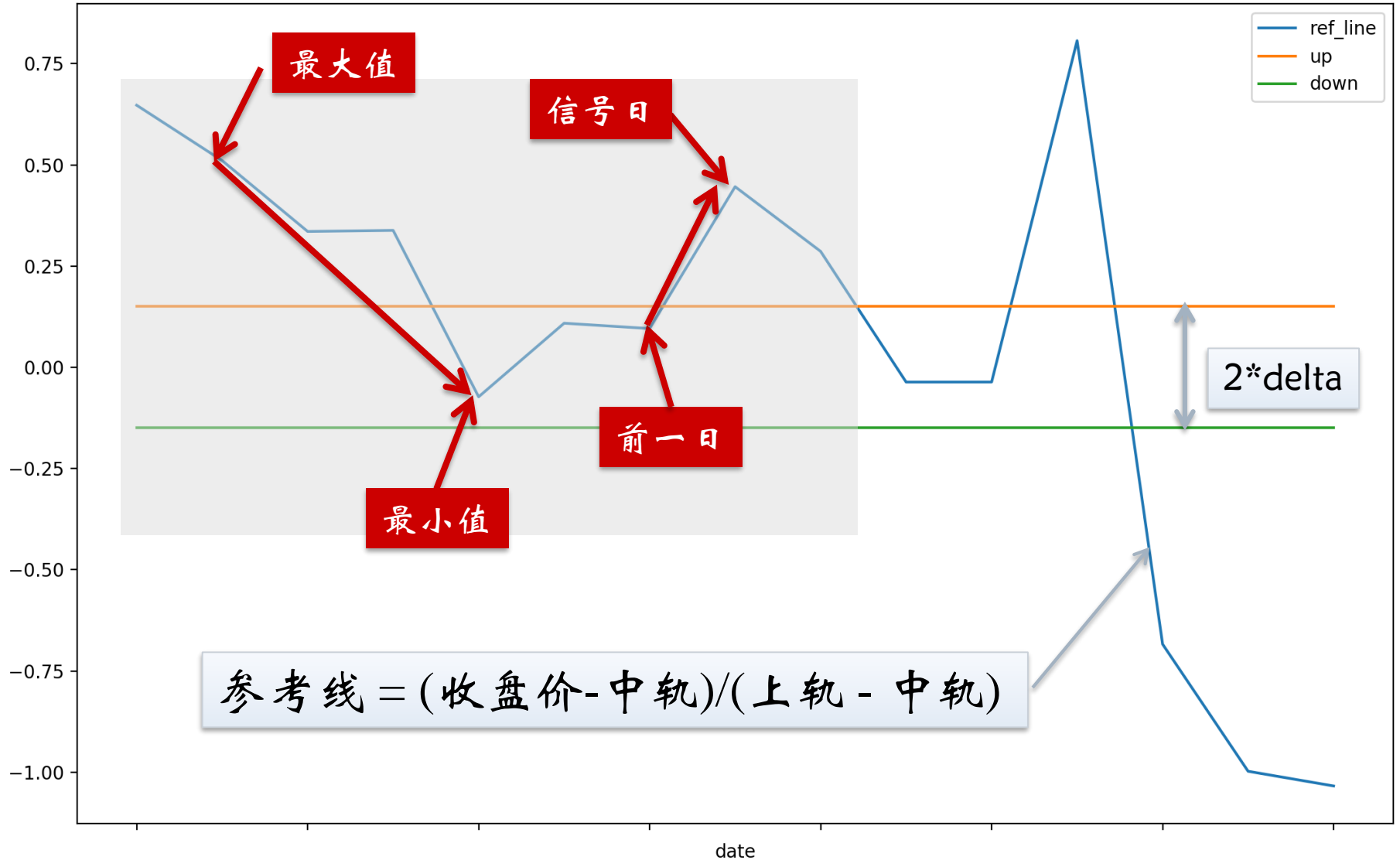
$$k=2$$

$$DOWN = MB - k * STD$$

下轨



Rebound




```
import traceback

import numpy as np
from pymongo import UpdateOne

from data.data_module import DataModule
from trading.signal.computer.base_signal_computer import BaseSignalComputer
from util.stock_util import get_all_codes

class BollSignalComputer(BaseSignalComputer):
    def __init__(self):
        BaseSignalComputer.__init__(self, 'boll_signal')

    def compute(self, begin_date, end_date):
        """
        计算指定日期内的信号
        :param begin_date: 开始日期
        :param end_date: 结束日期
        """
        all_codes = get_all_codes()

        dm = DataModule()

        N = 20
        k = 2

        for code in all_codes:
```

Boll

try:

```
df_daily = dm.get_k_data(code, autype='hfq', begin_date=begin_date, end_date=end_date)

# 计算MB，盘后计算，这里用当日的Close
df_daily['MID'] = df_daily['close'].rolling(N).mean()
# 计算STD20
df_daily['std'] = df_daily['close'].rolling(N).std()
# 计算UP
df_daily['UP'] = df_daily['MID'] + k * df_daily['std']
# 计算down
df_daily['DOWN'] = df_daily['MID'] - k * df_daily['std']

# 将日期作为索引
df_daily.set_index(['date'], inplace=True)

# 上轨和中轨右移一位
shifted_up = df_daily['UP'].shift(1)
shifted_middle = df_daily['MID'].shift(1)

# 收盘价突破或者跌破中轨的幅度占上轨和中轨宽度的比例
ref_line = (df_daily['close'] - shifted_middle) / (shifted_up - shifted_middle)

ref_prev = ref_line.shift(1)
```

Boll

```
# 找到时间窗口内的最小值
min_val = ref_line.rolling(10).min()
```

Boll

```
# 找到时间窗口内最低点前的最大值
max_leading_value = ref_line.rolling(10).apply(lambda vec:
                                                vec[:np.argmin(vec) + 1].max().astype(float), raw=True)
```

```
# 中轨支撑的作用的范围
delta = 0.15
```

```
# 判断是否存在中轨支撑反弹的信号，要求：
```

```
# 时间窗口的最低点之前的最大值大于delta，最小值的绝对值小于delta，就有一个穿越阈值分界线的动作；
```

```
# 当前日期在也在阈值之上，表示又从最低点穿越到阈值分界线之上；
```

```
# 而判断前一日在阈值分界线之下，表示穿越是在当前交易日完成
```

```
m_rebound_mask = (abs(min_val) <= delta) & (ref_line > delta) & (ref_prev <= delta) & \
                  (max_leading_value > delta)
```

```
# 将信号保存到数据库
```

```
update_requests = []
```

```
df_daily['m_rebound_mask'] = m_rebound_mask
```

```
df_daily = df_daily[df_daily['m_rebound_mask']]
```

```
for date in df_daily.index:
```

```
    doc = {'code': code, 'date': date, 'signal': 'mid_rebound'}
```

```
    update_requests.append(UpdateOne(
        doc, {'$set': doc}, upsert=True
    ))
```

```
if len(update_requests) > 0:
```

```
    update_result = self.collection.bulk_write(update_requests, ordered=False)
```

```
    print('%s, upserted: %4d, modified: %4d' %
```

```
        (code, update_result.upserted_count, update_result.modified_count),
        flush=True)
```

```
except:
```

```
    traceback.print_exc()
```

思考

- 实时价格的不稳定性
 - MB: 前一日的收盘价
- 周期
 - 日
 - 周
 - 月
 - 分钟
- 可变参数
 - N
 - k

“次日机会”量价因子的实现

要点

第二日上涨的概率较大

- 当日放量下跌
- 收的阴线实体不大
- 大部分时间在昨日收盘之上运行

图形化分析

当日下跌放量，
阴线实体不大

大部分时间在昨日
收盘以上运行



指标量化

□ 当日放量下跌

- 下跌：涨跌幅 < 0
- 放量：今日成交量 $> 1.5 \times$ 上一日成交量

□ 收的阴线实体不大

- 阴线实体：(开盘价-收盘价)/收盘价
- 不大：阴线实体 $< 3\%$

□ 大部分时间在昨日收盘之上运行

- 1分钟K线
- 昨日收盘之上：1分钟K线收盘价涨跌幅 > 0
- 大部分时间：1分钟K线数 > 150


```

class TomorrowChanceFactor(BaseFactor):
    def __init__(self):
        BaseFactor.__init__(self, 'tomorrow_chance')

    def compute(self, begin_date, end_date):
        codes = get_all_codes()
        dm = DataModule()

        for code in codes:
            df_daily = dm.get_k_data(code, begin_date=begin_date, end_date=end_date)

            if df_daily.index.size == 0:
                continue

            # 当日放量下跌
            df_daily['change'] = df_daily['close'] - df_daily['pre_close']
            df_daily = df_daily[df_daily['change'] < 0]
            df_daily['last_volume'] = df_daily['volume'].shift(1)
            df_daily.dropna(inplace=True)
            df_daily['volume_change'] = round(df_daily['volume']/df_daily['last_volume'],

2)

            df_daily = df_daily[df_daily['volume_change'] > 1.5]

            # 收的阴线实体不大
            df_daily['entity'] = round(abs((df_daily['open'] -df_daily['close']))) *
100/df_daily['close'], 2)
            df_daily = df_daily[df_daily['entity'] < 3]

```

次日机会

```
# 大部分时间在昨日之上运行
df_daily.set_index(['date'], 1, inplace=True)
```

```
update_requests = []
for date in df_daily.index:
```

次日机会

```
    # 大部分时间在昨日之上运行
```

```
    pre_close = df_daily.loc[date]['pre_close']
```

```
    df_minute = dm.get_k_data(code, period='M1', begin_date=date,
```

```
end_date=date)
```

```
    df_minute = df_minute[df_minute['close'] > pre_close]
```

```
    if df_minute.index.size > 150:
```

```
        update_requests.append(UpdateOne({
            'code': code, 'date': date},
            {'$set': {'code': code, 'date': date}},
            upsert=True))
```

```
    if len(update_requests) > 0:
```

```
        save_result = self.collection.bulk_write(update_requests, ordered=False)
```

```
        print('股票代码: %s, 因子: %s, 插入: %4d, 更新: %4d' %
```

```
              (code, self.name, save_result.upserted_count,
```

```
save_result.modified_count), flush=True)
```

总结

- 财务评分卡的框架
- 盈利能力及质量评价因子的原理
- 上市公司对财报中利润的操纵的手法
- 改进型ROE的实现
- Boll中轨支撑反弹的实现
- “次日机会”量价因子的实现

课后练习

- 根据本课程中财务评分卡的原理，编写代码实现盈利能力及质量评价因子的计算。

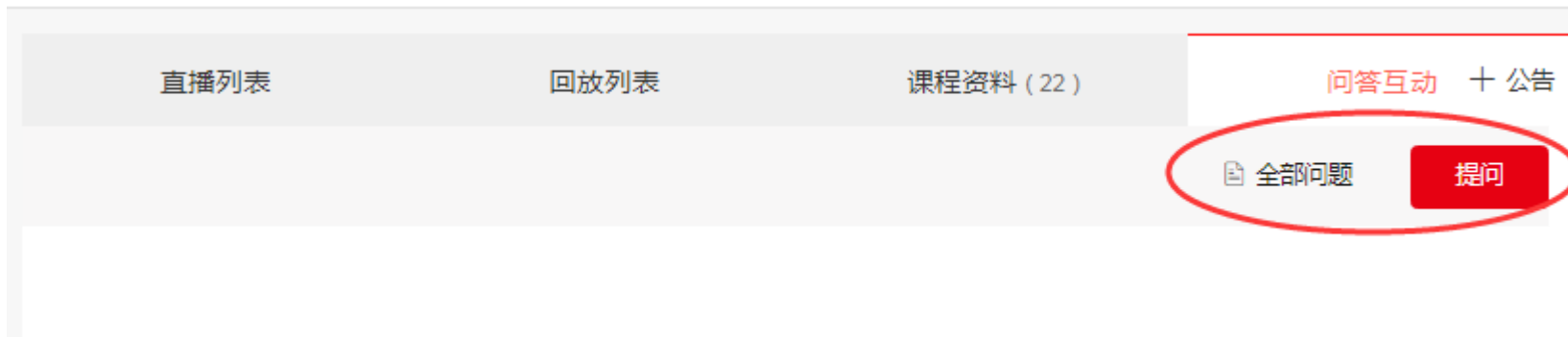
下节课预告

□ 题目：趋势型策略的改进和实战要点

问答互动

在所报课的课程页面，

- 1、点击“全部问题”显示本课程所有学员提问的问题。
- 2、点击“提问”即可向该课程的老师 and 助教提问问题。



联系我们

小象学院：互联网新技术在线教育领航者

— 微信公众号：**小象学院**



THANKS