

CZSC 官方文档 (0.6.8)

原创 置顶 zengbin93 于 2021-03-06 17:52:33 发布 11704 收藏 45

版权

分类专栏: 缠中说禅



缠中说禅 专栏收录该内容

109 订阅 11 篇文章 订阅

CZSC 是缠中说禅技术分析理论的 python 实现, 该理论源自 缠中说禅 博客。

注意: 文档写的比较匆忙, 很多地方可能没有讲清楚。有什么疑问, 可以到 常见问题 (FAQ) 看看。看完还有疑问, 可以直接在FAQ 链接下面跟帖提问。

此文档仅适用于 `czsc == 0.6.8`, 地址: <https://github.com/zengbin93/czsc>

CZSC 官方文档 (0.6.8)

1. 缠中说禅技术分析理论基础
2. 安装、更新与卸载
3. 基础信息
 - 3.1 符号说明
 - 3.2 基础形态定义
4. 简单用法
 - 4.1 查看分析结果
 - 4.2 单级别形态选股
 - 4.3 实时监控并推送消息到企业微信
5. 高级用法
 - 5.1 使用掘金量化终端进行缠论策略研究
 - 5.2 对接K线数据

1. 缠中说禅技术分析理论基础

- 缠中说禅重新编排版《论语》(整理版)
- 缠中说禅交易指南
- 缠中说禅技术原理
- 缠中说禅图解分析示范

以上四篇文章, 建议按顺序读一遍, 如果读完后感觉缠中说禅技术分析理论还不错, 推荐阅读原文。

2. 安装、更新与卸载

`czsc` 是一个python第三方库, 安装、更新、卸载与任何 python 第三方库都是一样的。 `czsc` 目前已经实现了缠中说禅技术分析理论中的一些核心概念, 包括分型、笔等。基于此, 设计并实现了N个缠中说禅技术分析信号, 用于辅助策略设计和多因子建模。

python 版本必须大于等于 3.7

直接从 [github](#) 安装与更新：

```
1 | pip install git+git://github.com/zengbin93/czsc.git -U
```

从 [pypi](#) 安装与更新：

```
1 | pip install czsc -U -i https://pypi.python.org/simple
```

快速卸载：

```
1 | pip uninstall czsc
```

3. 基础信息

3.1 符号说明

在信号和因子设计过程中，为了方便表达，引入以下符号，不区分大小写：

符号	说明
BI	笔对象
FX	分型对象
—	—
X5	五笔形态
X7	七笔形态
X9	九笔形态
L	多头信号，Long 的首字母
S	空头信号，Short 的首字母
—	—
C6	日线级别
C5	60分钟级别
C4	30分钟级别
C3	15分钟级别
C2	5分钟级别
C1	1分钟级别
—	—
L1	向下笔转折右侧
L2	向下笔转折左侧
L3	向上笔中继右侧
L4	向上笔中继左侧
S1	向上笔转折右侧

符号	说明
S2	向上笔转折左侧
S3	向下笔中继右侧
S4	向下笔中继左侧
—	—
R1	倒数第1笔
R2	倒数第2笔
R3	倒数第3笔
R4	倒数第4笔
R5	倒数第5笔

符合举例：
X5LA0 表示五笔形态为aAb式底背弛基础型，具体的说，X5 = 五笔形态，L = 多头，A0 = aAb式底背弛基础型

3.2 基础形态定义

在信号层面，定义了若干种常见形态，具体请参考：

- 缠中说禅形态挖掘之五笔形态
- 缠中说禅形态挖掘之七笔形态
- 缠中说禅形态挖掘之九笔形态

4. 简单用法

czsc 库目前内置了聚宽数据的获取方法，聚宽数据的用户可以很方便的使用一些基础功能。

4.1 查看分析结果

CzscTrader 是依赖聚宽数据实现的多级别分析类，聚宽数据用户可以直接使用。

```
1  # 首次使用需要设置聚宽账户
2  # from czsc.data.jq import set_token
3  # set_token("phone number", 'password') # 第一个参数是JQData的手机号，第二个参数是登录密码
4
5  from datetime import datetime
6  import czsc
7  from czsc.trader import CzscTrader
8
9  assert czsc.__version__ == '0.6.8'
10
11 # 在默认浏览器中打开最新分析结果，
12 ct = CzscTrader(symbol="000001.XSHG", end_date=datetime.now())
13 ct.open_in_browser(width="1400px", height="580px")
14 # open_in_browser 方法可以在windows系统中使用，如果无法使用，可以直接保存结果到 html 文件
15 # ct.take_snapshot(file_html="czsc_results.html", width="1400px", height="580px")
16
17
18 # 在默认浏览器中打开指定结束日期的分析结果)
19 from datetime import datetime
20 ct = CzscTrader(symbol="000001.XSHG", end_date=datetime.strptime("2024-03-04", "%Y-%m-%d"))
```

```
20 | ct = CZSCrader(symbol="000001.XSHG", end_date="2021-03-04")
    | ct.open_in_browser(width="1400px", height="580px")
```

4.2 单级别形态选股

以三买为例，单级别形态选股的实现方法如下：

```
1 | # 首次使用需要设置聚宽账户
2 | # from czsc.data.jq import set_token
3 | # set_token("phone number", 'password') # 第一个参数是JQData的手机号，第二个参数是登录密码
4 |
5 | from datetime import datetime
6 | from typing import List
7 | from czsc.data.jq import get_kline, get_index_stocks
8 | import czsc
9 | from czsc.analyze import CZSC
10 | from czsc.enum import Signals
11 |
12 | assert czsc.__version__ == '0.6.8'
13 |
14 | def is_third_buy(symbol):
15 |     """判断一个股票现在是否有日线三买"""
16 |     bars = get_kline(symbol, freq="D", end_date=datetime.now(), count=1000)
17 |     c = CZSC(bars, freq="日线")
18 |
19 |     # 在这里判断是否有五笔三买形态，也可以换成自己感兴趣的形态
20 |     if c.signals['倒1五笔'] in [Signals.X5LB0.value, Signals.X5LB1.value]:
21 |         return True
22 |     else:
23 |         return False
24 |
25 |
26 | if __name__ == '__main__':
27 |     # 获取上证50最新成分股列表，这里可以换成自己的股票池
28 |     symbols: List = get_index_stocks("000016.XSHG")
29 |     for symbol in symbols:
30 |         try:
31 |             if is_third_buy(symbol):
32 |                 print("{} - 日线三买".format(symbol))
33 |         except:
34 |             print("{} - 执行失败".format(symbol))
```

4.3 实时监控并推送消息到企业微信

CZSC 内部实现了企业微信群聊机器人的实时消息推送方法，可以推送文本消息和文件，关于企业微信群聊机器人的使用方法，请参考腾讯文档：<https://work.weixin.qq.com/api/doc/90000/90136/91770>

下面是一个实时监控因子的案例，以日线笔因子为例：

```
1 | # 首次使用需要设置聚宽账户
2 | # from czsc.data.jq import set_token
3 | # set_token("phone number", 'password') # 第一个参数是JQData的手机号，第二个参数是登录密码
4 | import traceback
5 | import time
6 | import shutil
7 | import os
8 | import pickle
9 | from datetime import datetime
10 |
```

```

11 from czsc.trader import CzscTrader, Factors
12 from czsc.utils.qywx import push_text, push_file
13
14 # =====
15 # 基础参数配置
16 ct_path = os.path.join(".", "czsc_traders")
17 os.makedirs(ct_path, exist_ok=True)
18 # 关于企业微信群聊机器人的使用文档, 参考: https://work.weixin.qq.com/api/doc/90000/90136/91770
19 # 企业微信群聊机器人的key
20 qywx_key = "4ad2e226-2519-4893-8670-*****"
21
22 # 定义需要监控的股票列表
23 symbols = ["300033.XSHE", "300803.XSHE", "002739.XSHE"]
24 # =====
25
26
27 def save_pkl(data, file):
28     with open(file, "wb") as f:
29         pickle.dump(data, f)
30
31 def read_pkl(file):
32     with open(file, "rb") as f:
33         data = pickle.load(f)
34     return data
35
36 def monitor(use_cache=True):
37     push_text("自选股CZSC因子监控启动 @ {}".format(datetime.now().strftime("%Y-%m-%d %H:%M")), qywx_key)
38     moni_path = os.path.join(ct_path, "monitor")
39     # 首先清空历史快照
40     if os.path.exists(moni_path):
41         shutil.rmtree(moni_path)
42     os.makedirs(moni_path, exist_ok=True)
43
44     for s in symbols:
45         print(s)
46         try:
47             file_ct = os.path.join(ct_path, "{}.ct".format(s))
48             if os.path.exists(file_ct) and use_cache:
49                 ct: CzscTrader = read_pkl(file_ct)
50                 ct.update_factors()
51             else:
52                 ct = CzscTrader(s, max_count=1000)
53             save_pkl(ct, file_ct)
54
55             # 每次执行, 会在moni_path下面保存一份快照
56             file_html_ = os.path.join(moni_path, f"{ct.symbol}_{ct.kf.end_dt.strftime('%Y%m%d%H%M')}.htm")
57             ct.take_snapshot(file_html_, width="1400px", height="580px")
58
59             if ct.s['日线因子'] != Factors.Other.value:
60                 msg = "{} - {}\n".format(s, ct.s['日线因子'])
61                 msg += "同花顺F10: http://basic.10jqka.com.cn/{}".format(s[:6])
62                 push_text(msg, key=qywx_key)
63                 file_html_new = os.path.join(moni_path, f"{ct.symbol}_{ct.kf.end_dt.strftime('%Y%m%d%H%M')}")
64                 shutil.copyfile(file_html_, file_html_new)
65                 push_file(file_html_new, key=qywx_key)
66
67         except Exception as e:
68             traceback.print_exc()
69             print("{} 执行失败 - {}".format(s, e))
70
71     push_text("自选股CZSC因子监控结束 @ {}".format(datetime.now().strftime("%Y-%m-%d %H:%M")), qywx_key)
72
73 def run_monitor():
74     mdt = ["09:30", "10:00", "10:30", "11:00", "11:20", "13:00", "13:30", "14:00", "14:30", "14:50"]
75     monitor()
76     while 1:

```

```

77         print(datetime.now().strftime("%H:%M"))
78         if datetime.now().strftime("%H:%M") in mdt:
79             monitor()
80             time.sleep(3)
81
82
83 if __name__ == '__main__':
    run_monitor()

```

5. 高级用法

5.1 使用掘金量化终端进行缠论策略研究

使用掘金量化终端进行缠论策略研究的代码稍微有点复杂，主要是需要实现一些掘金量化平台的一些回调函数和下单方法，具体的内容，点击访问：https://github.com/zengbin93/czsc_gm

5.2 对接K线数据

缠中说禅技术分析理论只需要K线，因此，任何第三方数据源只要转换一下K线获取方法就可以使用。下面以 **Tushare** 数据为例，介绍下如何进行K线数据转换，重点是 **format_kline** 方法。

```

1
2 import pandas as pd
3 import tushare as ts
4 from datetime import datetime, timedelta
5 from typing import List
6 import czsc
7 from czsc.analyze import CZSC, RawBar
8 from czsc.enum import Signals
9
10 assert czsc.__version__ == '0.6.8'
11
12 # 使用第三方数据，只需要定义一个K线转换函数
13 def format_kline(kline: pd.DataFrame) -> List[RawBar]:
14     """
15
16     :param kline: Tushare 数据接口返回的K线数据
17     :return: 转换好的K线数据
18     """
19     bars = []
20     records = kline.to_dict('records')
21     for record in records:
22         # 将每一根K线转换成 RawBar 对象
23         bar = RawBar(symbol=record['ts_code'], dt=pd.to_datetime(record['trade_date']), open=record['open'],
24                       close=record['close'], high=record['high'], low=record['low'], vol=record['vol'])
25         bars.append(bar)
26     return bars
27
28
29 def is_third_buy(ts_code):
30     """判断一个股票现在是否有日线三买"""
31     # 调用tushare的K线获取方法，Tushare数据的使用方法，请参考: https://tushare.pro/document/2
32     end_date = datetime.now()
33     start_date = end_date - timedelta(days=1000)
34     df = ts.pro_bar(ts_code=ts_code, adj='qfq', asset="E",
35                   start_date=start_date.strftime("%Y%m%d"),
36

```

```
37         end_date=end_date.strftime("%Y%m%d"))
38     bars = format_kline(df)
39     c = CZSC(bars, freq="日线")
40
41     # 在这里判断是否有五笔三买形态, 也可以换成自己感兴趣的形态
42     if c.signals['倒1五笔'] in [Signals.X5LB0.value, Signals.X5LB1.value]:
43         return True
44     else:
45         return False
46
47
48 if __name__ == '__main__':
49     # 这里可以换成自己的股票池
50     ts_codes = ['603259.SH', '603288.SH', '603501.SH', '603986.SH']
51     for ts_code in ts_codes:
52         try:
53             if is_third_buy(ts_code):
54                 print("{} - 日线三买".format(ts_code))
55         except:
56             print("{} - 执行失败".format(ts_code))
```



zengbin93

关注

14



45



11



专栏目录

python缠论代码_缠论dll(czsc - 缠中说禅技术分析工具)

weixin_29554849的博客 2343

【实例简介】dll插件【实例截图】【核心代码】czsc-master | LICENSE | README.md | czsc | __init__.py | analyz...

评论 11 条 >

5



danielhgb

热评

setup.py 出现错误: usage: get_data_czsc.py [global_opts] cmd1 [cmd1_opts] [cmd2 [cmd2_opts] ...] or: get_data_czsc.py [cmd1 cmd2 ...] or: get_data_czsc.py --help-commands or: get_data_czsc.py cmd --help error: no commands supplied 请解决, 谢谢

fico-CZSCfico-CZSC

06-23

fico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSCfico-CZSC

fico-CZSC2fico-CZSC2fico-CZSC2

06-23

fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-CZSC2fico-C...

欢迎你来到CSDN 热门推荐

CSDN 官方博客 2万+

哈罗, 欢迎你来到CSDN! 完成新手任务可获得新星勋章, 电子书体验周卡 (本活动仅限于首次在CSDN发布原创博文的用户)

webpack性能优化——DLL

weixin_30549657的博客 187

Webpack性能优化的方式有很多种, 本文之所以将 dll 单独讲解, 是因为 dll 是一种最简单粗暴并且极其有效的优化方式。在通常的打包...

文档编制规范 最新发布

Candy5204的博客 93

本文简述文档分类, 文档编号, 根据页面布局, 目录, 正文, 小标题级别, 图片与表格, 功能点与列表完善文档编写的格式要求。

SpringMVC下载word文档

xue2xue的博客 1638

@RequestMapping("/downloadCzsc") public void downloadCzsc(HttpServletRequest request, HttpServletResponse response) { try { ...

get和post区别

czsc03的博客 18

请求缓存: GET 会被缓存, 而post不会 保留浏览器历史记录: GET可以, 而POST不能 用处: get常用于取回数据, post用于提交数据 安...

2016SCZSC Day6

MintGreen 288

T1 看电影 n个人围成一圈, 在n个人中每次随机删掉一个位置的人, 若已经被删, 则顺时针找接下来第一个没有被删的人。直到剩k人, 问...

死锁的解决方案

czsc03的博客 73

造成死锁有四个方面: 资源互斥, 不可剥夺, 保持并请求, 循环等待 预防死锁的方法: 资源共享, 可剥夺, 一次性全部申请, 顺序资源分...

前端 Webpack 工程化的最佳实践

CSDN资讯 1万+

作者 | 阿里文娱前端开发专家范苏责编 | 屠敏头图 | CSDN 下载自视觉中国引言 前端构建工具的演变回想在2015-2016年的时候, 开发者...

zsczsc1011的专栏

代码重用和接口重用 4356

在面试中，面向对象语言中经常会提到代码重用和接口重用的概念，有些同学会很疑惑，其实仔细翻阅下书本就不难发现，代码重用就是...

uwsgi安装报错Command errored out with exit status 1: weixin_41486438的博客 44

从yum源上下载安装对应python版本的python-devel sudo yum install python3-devel

用python编写缠论中枢_【缠论】python 实现 weixin_40002224的博客 3140

缠论来源于缠中说缠博客，欢迎加微信探讨，我的微信号是 zengbin93考虑到我自己对缠论的理解还在“四不像”水平，**CZSC** 这个库只有参...

1.8新特性提升 zxczsczxc的博客 28

python文件命名的坑导致引入模块方法失败 yu_kang的博客 320

python文件命名的坑导致引入模块失败 本人python入门。。。在练习枚举enum类型时，创建了个名为enum.py文件 结果。。 ImportError...

python使用pip安装库_使用pip安装python库的多种方式 weixin_30847063的博客 1344

操作系统： CentOS7.5.1804_x64python 版本：3.6.81、使用pip在线安装1.1 安装单个package格式如下：pip install SomePackage示例...

安装文件怎么写？ xiuluomenshen1986的专栏 1699

；脚本用 Inno Setup 脚本向导生成。；查阅**文档**获取创建 INNO SETUP 脚本文件详细资料![Setup]AppName=科发医院管理系统AppVerNa...

“相关推荐”对你有帮助么？