

Ai全自动编写策略：牛起指标，收益率326%，年化73%，胜率63%

 **松鼠Quant**
公号：松鼠Quant 量化交易知识分享

已关注

8 人赞同了该文章

工具推荐

· 参数筛选工具
mp.weixin.qq.com/s?__biz=Mjm5MzE5NzUxNw==&m...

Ai帮你编写策略
mp.weixin.qq.com/s?__biz=Mjm5MzE5NzUxNw==&m...

· 订单流图表
mp.weixin.qq.com/s?__biz=Mjm5MzE5NzUxNw==&m...

· 加入2025俱乐部
mp.weixin.qq.com/s?__biz=Mjm5MzE5NzUxNw==&m...

· Ai投研助手课程
mp.weixin.qq.com/s?__biz=Mjm5MzE5NzUxNw==&m...

Ai编写策略是量化领域的新常态，松鼠Quant在早些时候推出了开源项目[《gitee官方推荐，松鼠QuantAi+回测框架详细版教程\(图文\)》](#)让Ai自动编写策略，自动回测，给出总结和迭代意见。
现在我们将[通达信](#)指标：牛起指标，让Ai自动完成python编写，然后自动回测总结。

关于作者

 **松鼠Quant**
公号：松鼠Quant 量

AI创新屋、宽客与人生、Ch
注了他

回答	文章	1
354	347	1

已关注

发



通达信：

```
VAR1+:=MA(V,5);
VAR2+:=MA(V,10);
VAR3+:=V>REF(HHV(V,2),1) AND RANGE(VAR1,VAR2,V) AND UPDAY(VAR1,2) AND UPDAY(VAR
VAR4+:=H>REF(HHV(H,2),1);VAR8+:=MAX(C,0);
VAR5+:=VAR8>REF(HHV(VAR8,2),1);
历史大底:=VAR3 AND VAR4 AND VAR5;
VAR6:=BARSLAST(历史大底=1);
VAR7+:=REF(REF(H,1),VAR6);
DB:=IF(历史大底,0,IF(C<VAR7,1,0));
LJ:=REF(HHV(H,VAR6+1),BARSLAST(DB));
NOTEXT:LJ COLORLIMAGENTA;
STICKLINE(IF(C>LJ,IF(REF(VAR8<LJ,1),1,0),0),0,C,2.2,0),COLORYELLOW;
DRAWICON(IF(C>LJ,IF(REF(VAR8<LJ,1),1,0),0),L,25);
```

接下来让Ai帮我们转化到**松鼠Quant**回测框架⁺内，我们不需要写一行代码！！

github地址(国外):

[github.com/songshuquant...](https://github.com/songshuquant)

gitee地址(国内):

gitee.com/ssquant/ssqua...

如何使用，手把手教程：《[gitee官方推荐，松鼠QuantAi+回测框架详细版教程\(图文\)](#)》

Python源码：

```
import pandas as pd
import numpy as np
from ssquant.backtest import MultiSourceBacktest, MultiSourceRealtime
from ssquant.api.:
```

```

def initialize(api: StrategyAPI)
:   api.log("成交量突破+价格突破策略初始化")
   api.entry_price = 0 # 添加入场价格变量

def strategy_function(api: StrategyAPI):
    # 参数和数据验证
    current_idx = api.get_idx()
    if current_idx < 20: # 确保有足够的历史数据计算指标
        return

# 获取数据
    close = api.get_close()
    high = api.get_high()
    low = api.get_low()
    open_ = api.get_open()
    volume = api.get_volume()
    current_pos = api.get_pos()

    # 增强数据验证
    if any(data is None or len(data) == 0 for data in [close, high, low, open_, volume]):
        api.log("警告: 数据为空")
        return
    if current_idx >= len(close):
        api.log("警告: 索引超出数据范围")
        return

    # 计算指标
    # VAR1: 5日成交量均线
    var1 = volume.rolling(5).mean()
    # VAR2: 10日成交量均线
    var2 = volume.rolling(10).mean()
    # VAR3: 成交量突破条件 - 进一步简化
    var3 = volume > volume.rolling(3).mean()
# 改为3日均量, 不要求突破最大值
    # VAR4: 价格突破2日最高价
    var4 = high > high.rolling(2).max().shift(1)
    # VAR8: 当日K线实体高点
    var8 = np.maximum(close, open_)
    # VAR5: K线实体高点突破2日最高
    var5 = var8 > var8.rolling(2).max().shift(1)
    # 历史大底信号
    history_bottom = var3 & var4 & var5

    # 添加调试日志
    if current_idx % 20 == 0: # 每20个周期打印一次日志
        api.log(f"==== 调试信息 [idx={current_idx}] =====")
        api.log(f"成交量突破(var3): {var3.iloc[current_idx]}")
        api.log(f"价格突破(var4): {var4.iloc[current_idx]}")
        api.log(f"实体高点突破(var5): {var5.iloc[current_idx]}")
        api.log(f"历史大底信号: {history_bottom.iloc[current_idx]}")
        api.log(f"当前成交量: {volume.iloc[current_idx]}")
        api.log(f"5日均量: {var1.iloc[current_idx]}")
        api.log(f"10日均量: {var2.iloc[current_idx]}")
        api.log(f"2日最高价: {high.rolling(2).max().shift(1).iloc[current_idx]}")

    # 计算最近的

```

```

recent_history = history_bottom.iloc[max(0, current_idx-5):current_idx+1] # i
var6 = 0
if recent_history.any():
    var6 = len(recent_history) - 1 - recent_history[::-1].values.argmax()

    # 添加var6调试日志
if current_idx % 20 == 0:
    api.log(f"历史大底距离(var6): {var6}")

    # 修改lj的计算逻辑
lj = 0
if var6 > 0:
    # 从历史大底到当前的最高价
    start_idx = current_idx - var6
    lj = high.iloc[start_idx:current_idx+1].max()

    # 添加lj调试日志
if current_idx % 20 == 0:
    api.log(f"历史大底高点(lj): {lj}")
    api.log("=====")
# 交易信号
if lj > 0:
    current_close = close.iloc[current_idx]
    current_high = high.iloc[current_idx]
    prev_close = close.iloc[current_idx-1]

    # 进一步放宽买入条件:
    # 1. 当日最高价接近历史高点（允许10%的差距）
    # 2. 收盘价站稳在历史高点附近（允许15%的回撤）
    # 3. 相对前一日有上涨
    price_threshold = lj * 0.85 # 允许15%的回撤
    high_threshold = lj * 0.90 # 允许10%的差距
    buy_signal = (current_high > high_threshold) and \
                 (current_close > price_threshold) and \
                 (current_close > prev_close * 1.005) # 要求至少0.5%的涨幅

    # 添加买入条件的详细日志
if current_idx % 5 == 0 or buy_signal: # 每5个周期或有买入信号时打印
    api.log(f"买入条件检查 [idx={current_idx}]:")
    api.log(f"当日最高价: {current_high:.2f}")
    api.log(f"当前收盘价: {current_close:.2f}")
    api.log(f"前一日收盘价: {prev_close:.2f}")
    api.log(f"历史高点(lj): {lj:.2f}")
    api.log(f"价格阈值(85%): {price_threshold:.2f}")
    api.log(f"最高价阈值(90%): {high_threshold:.2f}")
    api.log(f"条件1(最高价>阈值): {current_high > high_threshold}")
    api.log(f"条件2(收盘价>阈值): {current_close > price_threshold}")
    api.log(f"条件3(上涨>0.5%): {current_close > prev_close * 1.005}")
    api.log(f"买入信号: {buy_signal}")
else:
    buy_signal = False
    current_close = close.iloc[current_idx]
    prev_close = close.iloc[current_idx-1]
# 执行交易
if current_pos:
    # 计算当前:

```

```

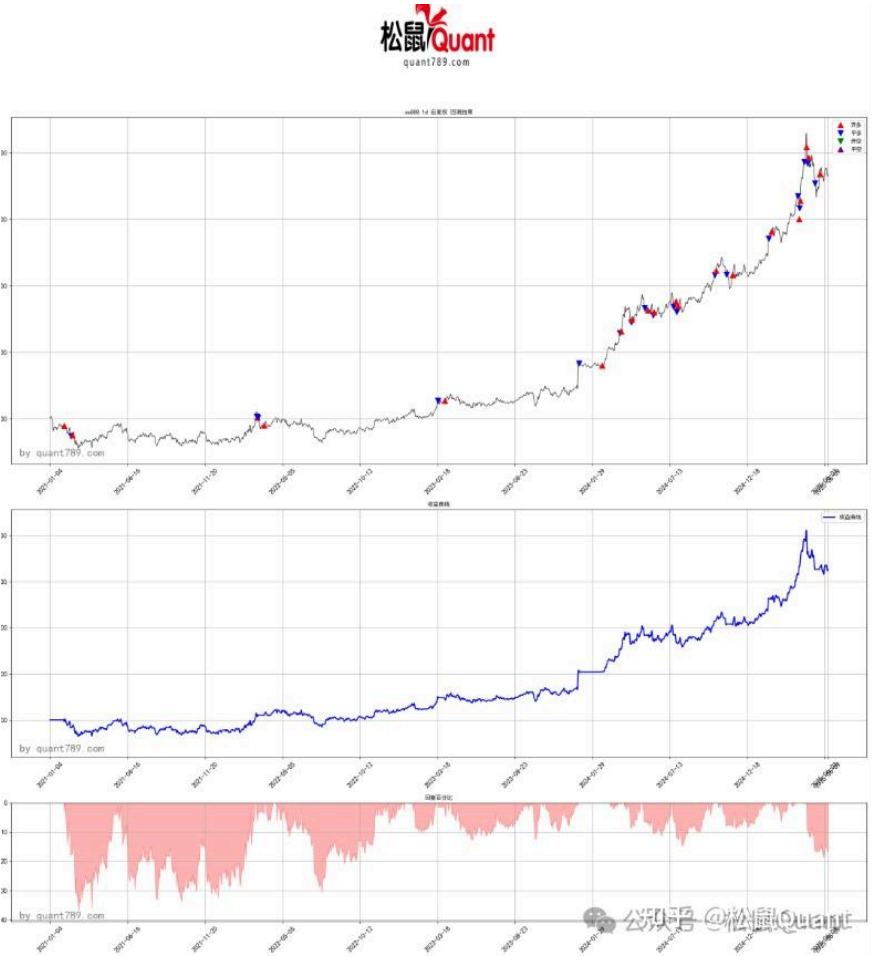
        if api.entry_price > 0: # 确保有入场价格
            profit_ratio = (current_close - api.entry_price) / api.entry_price
# 平仓条件:
    # 1. 止损: 跌破历史高点15%
    # 2. 止盈: 盈利超过8%
    # 3. 破位: 收盘价跌破5日均线且跌幅超过2%
    ma5 = close.rolling(5).mean()
    sell_signal = (current_close < lj * 0.85) or \
        (profit_ratio > 0.08) or \
        ((current_close < ma5.iloc[current_idx]) and \
         (current_close < prev_close * 0.98)) # 破位
    if sell_signal:
        api.close_all(order_type='next_bar_open')
        if current_close < lj * 0.85:
            api.log(f"止损平仓, 当前价: {current_close:.2f}")
        elif profit_ratio > 0.08:
            api.log(f"止盈平仓, 盈利: {profit_ratio*100:.1f}%")
        else:
            api.log(f"破位平仓, 当前价: {current_close:.2f}")
        api.entry_price = 0 # 重置入场价格

    elif buy_signal and current_pos <= 0: # 没有持仓且有买入信号
        api.close_all(order_type='next_bar_open')
        api.buy(volume=1, order_type='next_bar_open')
        api.entry_price = current_close # 记录入场价格
        api.log(f"买入信号触发, 价格接近历史高点: {lj:.2f}")
if __name__ == "__main__":
    # 导入API认证信息
    try:
        from ssquant.config.auth_config import get_api_auth
        API_USERNAME, API_PASSWORD = get_api_auth()
    except ImportError:
        print("警告: 未找到 auth_config.py 文件, 请在下方填写您的认证信息: API_USERNAME, API_PASSWORD")
        API_USERNAME = "your_username" # 替换为您的实际用户名
        API_PASSWORD = "your_password" # 替换为您的实际密码
# 回测配置
backtester = MultiSourceBacktester()
backtester.set_base_config({
    'username': API_USERNAME, # 添加用户名
    'password': API_PASSWORD, # 添加密码
    'use_cache': True,
    'save_data': True,
    'debug': True
})
backtester.add_symbol_config(
    symbol='au888',
    config={
        'start_date': '2021-01-01',
        'end_date': '2025-12-31',
        'initial_capital': 100000.0,
        'commission': 0.0002, # 手续费万分之0.2
        'margin_rate': 0.08, # 保证金率8%
        'contract_multiplier': 1000, # 沪金一手1000克
        'periods': [
            {'

```

```
}  
)  
  
    results = backtester.run(  
        strategy=strategy_function,  
        initialize=initialize,  
        strategy_params={},  
    )
```

回测绩效



数据源 #0 (au888 1d) 回测结果:
总交易次数: 19
盈利交易: 12, 亏损交易: 7
胜率: 63.16%
初始权益: 100000.00
期末权益: 426592.73
净值: 4.2659
总点数盈亏: 331.44
总金额盈亏: 331440.00
总手续费: 4527.27
总净盈亏: 327066.37
平均盈利: 36550.65
平均亏损: -15934.48
盈亏比: 2.29
最大回撤: 95820.21 (36.86%)
年化收益率: 73.83%
夏普比率: 1.07

 知乎 @松鼠Quant

----- 回测结果摘要 -----

[2025-06-09 21:47:54]

总收益率: 326.59%

[2025-06-09 21:47:54]

年化收益率: 73.83%

[2025-06-09 21:47:54]

最大回撤: 95820.21

[2025-06-09 21:47:54]

夏普比率: 1.07

[2025-06-09 21:47:54]

胜率: 63.16%

[2025-06-09 21:47:54]

总交易次数: 19

[2025-06-09 21:47:54]

盈利交易: 12

[2025-06-09 21:47:54]

亏损交易: 7

[2025-06-09 21:47:54]

 知乎 @松鼠Quant

交易明细:

1. 2021-01-30 00:00:00 开多 1手 价格:389.57 手续费:77.91

2. 2021-02-19 00:00:00 平多 1手 价格:373.86 点数盈亏:-15.71 金额盈亏:-15710.00 手续费:74.77 净盈亏:-15862.69 ROI:-50.90%

3. 2021-02-22 00:00:00 开多 1手 价格:375.42 手续费:75.08

4. 2022-03-09 00:00:00 平多 1手 价格:403.11 点数盈亏:27.69 金额盈亏:27690.00 手续费:80.62 净盈亏:27534.29 ROI:85.38%

5. 2022-03-10 00:00:00 开多 1手 价格:402.09 手续费:80.42

6. 2022-03-11 00:00:00 平多 1手 价格:401.44 点数盈亏:-0.65 金额盈亏:-650.00 手续费:80.29 净盈亏:-810.71 ROI:-2.52%

7. 2022-03-23 00:00:00 开多 1手 价格:389.66 手续费:77.93

8. 2023-03-20 00:00:00 平多 1手 价格:427.26 点数盈亏:37.60 金额盈亏:37600.00 手续费:85.45 净盈亏:37436.62 ROI:109.53%

9. 2023-04-01 00:00:00 开多 1手 价格:427.03 手续费:85.41

10. 2024-01-03 00:00:00 平多 1手 价格:483.32 点数盈亏:56.29 金额盈亏:56290.00 手续费:96.66 净盈亏:56107.93 ROI:145.11%

11. 2024-02-26 00:00:00 开多 1手 价格:480.10 手续费:96.02

12. 2024-03-30 00:00:00 平多 1手 价格:528.84 点数盈亏:48.74 金额盈亏:48740.00 手续费:105.77 净盈亏:48538.21 ROI:114.73%

13. 2024-04-02 00:00:00 开多 1手 价格:531.33 手续费:106.27

14. 2024-04-24 00:00:00 平多 1手 价格:545.21 点数盈亏:13.88 金额盈亏:13880.00 手续费:109.04 净盈亏:13664.69 ROI:31.33%

15. 2024-04-25 00:00:00 开多 1手 价格:549.14 手续费:109.83

16. 2024-05-24 00:00:00 平多 1手 价格:567.05 点数盈亏:17.91 金额盈亏:17910.00 手续费:113.41 净盈亏:17686.76 ROI:38.99%

17. 2024-05-30 00:00:00 开多 1手 价格:563.58 手续费:112.72

18. 2024-06-12 00:00:00 平多 1手 价格:555.94 点数盈亏:-7.64 金额盈亏:-7640.00 手续费:111.19 净盈亏:-7863.90 ROI:-17.44%

19. 2024-06-13 00:00:00 开多 1手 价格:560.30 手续费:112.06

20. 2024-07-20 00:00:00 平多 1手 价格:568.56 点数盈亏:18.26 金额盈亏:8260.00 手续费:113.71 净盈亏:8034.23 ROI:17.66%

21. 2024-07-25 00:00:00 开多 1手 价格:576.94 手续费:115.39

22. 2024-07-26 00:00:00 平多 1手 价格:560.84 点数盈亏:-16.10 金额盈亏:-16100.00 手续费:112.17 净盈亏:-16327.56 ROI:-35.38%

23. 2024-07-29 00:00:00 开多 1手 价格:569.88 手续费:113.98

24. 2024-10-17 00:00:00 平多 1手 价格:616.57 点数盈亏:46.69 金额盈亏:46690.00 手续费:123.31 净盈亏:46452.71 ROI:94.18%

25. 2024-10-19 00:00:00 开多 1手 价格:623.21 手续费:124.64

26. 2024-11-08 00:00:00 平多 1手 价格:617.25 点数盈亏:-5.96 金额盈亏:-5960.00 手续费:123.45 净盈亏:-6208.09 ROI:-12.45%

27. 2024-11-20 00:00:00 开多 1手 价格:616.73 手续费:123.35

28. 2025-02-06 00:00:00 平多 1手 价格:671.66 点数盈亏:54.93 金额盈亏:54930.00 手续费:134.33 净盈亏:54672.32 ROI:101.75%

29. 2025-02-11 00:00:00 开多 1手 价格:681.73 手续费:136.35

30. 2025-04-02 00:00:00 平多 1手 价格:735.68 点数盈亏:53.95 金额盈亏:53950.00 手续费:147.14 净盈亏:53666.52 ROI:91.19%

31. 2025-04-07 00:00:00 开多 1手 价格:700.64 手续费:140.13

32. 2025-04-09 00:00:00 平多 1手 价格:717.10 点数盈亏:16.46 金额盈亏:16460.00 手续费:143.42 净盈亏:16176.45 ROI:28.20%

33. 2025-04-10 00:00:00 开多 1手 价格:728.00 手续费:145.60

34. 2025-04-18 00:00:00 平多 1手 价格:786.94 点数盈亏:58.94 金额盈亏:58940.00 手续费:157.39 净盈亏:58637.01 ROI:93.14%


35. 2025-04-22 00:00:00 开多 1手 价格:809.32 手续费:161.86

36. 2025-04-24 00:00:00 平多 1手 价格:784.72 点数盈亏:-24.60 金额盈亏:-24600.00 手续费:156.94 净盈亏:-24918.81 ROI:-38.49%

37. 2025-04-25 00:00:00 开多 1手 价格:793.69 手续费:158.74

38. 2025-05-13 00:00:00 平多 1手 价格:754.45 点数盈亏:-39.24 金额盈亏:-39240.00 手续费:150.89 净盈亏:-39549.63 ROI:-62.29%

39. 2025-05-22 00:00:00 开多 1手 价格:768.19 手续费:153.64

 知乎 @松鼠Quant

没错，整个开发过程我们没有手写一行代码，松鼠QuantAi助手会编写，回测，分析，修改。基于松鼠Quant的MCP服务器+Deepseek开启全新量化学习的路径，即使0基础也能完成策略的编写与验证！

如果你喜欢这个系列，请点赞，转发，在看三连。我们根据受欢迎程度决定是否继续更新，感谢您的支持。

发布于 2025-06-12 11:02 · 山西

交易策略 AI 期货



理性发言，友善互动

1 条评论

默认 最新



博者不知

ai怎么解决漂移
06-14 · 吉林

...

回复 喜欢

推荐阅读

基于python的用户行为分析

今天咱们先介绍一下，几个指标，从几个维度进行分析？到了具体的模型会给大家介绍其原理。此项目数源是来源于阿里天池数据集，我的这个有12256906条数据，是2014.11.18-2014.12.18；在分...

我叫武月 发表于数据分析那...

TabNet

量化投资入门系列（九）——
TabNet解析与使用

李浩然 发表于量化投资从...

Python之营销预测模型

本文讲解如何使用Python工具制作会员营销预测的模型，希望能通过数据预测在下一次营销活动时，响应活动会员的具体名单和响应概率，以此来制定针对性的营销策略。当然了，也可以基于现有的...

大米糕糕