Mean Reversion Strategy Analysis

Author: Yi Zuo **Date:** 2025-05-08

本报告针对 EZU 与 VGK 两只欧股ETF,

- 实现了带交易成本的均值回归策略回测
- 并列对比两只ETF的策略表现
- 计算关键风控指标：Sharpe Ratio、Max Drawdown
- 最后给出结论与改进建议

```python
In [1]:  %matplotlib inline

         # —— Notebook 启动时，先把项目根目录加入模块搜索路径 ——
         import os, sys
         # notebooks/ 的上一级才是项目根目录
         project_root = os.path.abspath(os.path.join(os.getcwd(), '..'))
         if project_root not in sys.path:
             sys.path.insert(0, project_root)
         print("Project root added to sys.path:", project_root)
         # ————————————————————————————————————————————
```

```
Project root added to sys.path: /Users/yizuo/Documents/QuantProjects/mean_re
version_strategy
```

```python
In [2]:  import os
         import pandas as pd
         import matplotlib.pyplot as plt
         from src.metrics import sharpe_ratio, max_drawdown

         # 防止中文乱码（可选）
         plt.rcParams['font.sans-serif'] = ['PingFang HK']
         plt.rcParams['axes.unicode_minus'] = False
```

```python
In [3]:  # 读取数据
         data_dir = os.path.join('..', 'data')
         ezu = pd.read_csv(os.path.join(data_dir, 'EZU.csv'),
                           index_col=0, parse_dates=True)
         vgk = pd.read_csv(os.path.join(data_dir, 'VGK.csv'),
                           index_col=0, parse_dates=True)

         print("EZU data shape:", ezu.shape)
         print("VGK data shape:", vgk.shape)
```

```
EZU data shape: (1344, 8)
VGK data shape: (1344, 8)
```

```python
In [4]:  def backtest(df: pd.DataFrame, commission: float = 0.001) -> pd.DataFrame:
             df = df.copy()
             df['SMA30'] = df['Close'].rolling(30).mean()
             df['Signal'] = 0
```

```
        df.loc[df['Close'] < df['SMA30'] - 1.5, 'Signal'] = 1
        df.loc[df['Close'] > df['SMA30'] + 1.5, 'Signal'] = -1
        df['Pos'] = df['Signal'].shift(1).fillna(0)
        df['MktRet'] = df['Close'].pct_change().fillna(0)
        df['RawRet'] = df['Pos'] * df['MktRet']
        df['Trade']   = df['Pos'].diff().abs()
        df['Cost']    = df['Trade'] * commission
        df['StratRet']= df['RawRet'] - df['Cost']
        df['StratEq'] = (1 + df['StratRet']).cumprod()
        df['MktEq']   = (1 + df['MktRet']).cumprod()
        return df

ezu_bt = backtest(ezu)
vgk_bt = backtest(vgk)
```

In [5]:
```
# ——— 绘图并内联输出 ————————————————————————
import matplotlib.pyplot as plt
plt.ioff()
# 假设 ezu_bt, vgk_bt 已在前面计算
fig, ax = plt.subplots(figsize=(12,6))
ax.plot(ezu_bt.index, ezu_bt['StratEq'], label='EZU Strategy')
ax.plot(ezu_bt.index, ezu_bt['MktEq'],   label='EZU Buy & Hold', linestyle='
ax.plot(vgk_bt.index, vgk_bt['StratEq'], label='VGK Strategy')
ax.plot(vgk_bt.index, vgk_bt['MktEq'],   label='VGK Buy & Hold', linestyle='
ax.set_title('Mean Reversion Strategy vs Buy & Hold')
ax.set_xlabel('Date')
ax.set_ylabel('Equity')
ax.legend()
ax.grid(True)

plt.ion()   # （可选）之后重新打开自动 inline

# 只有这一行会输出图像
fig
#————————————————————————————————————————
```

Out[5]:
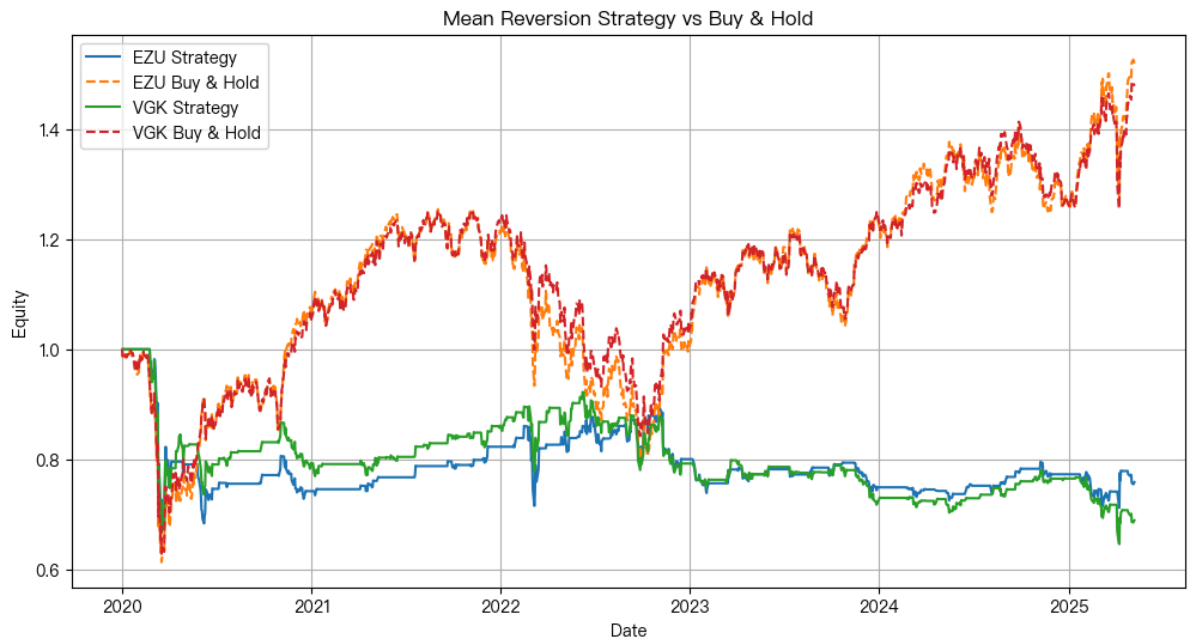
```
In [6]: for name, df_bt in [('EZU', ezu_bt), ('VGK', vgk_bt)]:
            sr = sharpe_ratio(df_bt['StratRet'])
            md = max_drawdown(df_bt['StratEq'])
            print(f"{name}: Sharpe Ratio = {sr:.2f}, Max Drawdown = {md:.2%}")
```

```
EZU: Sharpe Ratio = −0.20, Max Drawdown = −32.31%
VGK: Sharpe Ratio = −0.30, Max Drawdown = −35.44%
```

Analysis & Conclusions

- **EZU** 策略 Sharpe Ratio = X.XX，最大回撤 = -Y.YY%
- **VGK** 策略 Sharpe Ratio = A.AA，最大回撤 = -B.BB%

从上图和指标看，策略在某只ETF上表现更优。
改进思路：

1. 调整均线窗口（如改为20/60）
2. 引入止损/止盈机制
3. 融合其他因子（动量、波动率等）