专业·创新·增值

金程教育
GOLDEN FUTURE

配对交易 Pair Trading

纪慧诚
金程教育资深培训讲师
CFA FRM RFP

CONTENTS

PROFESSIONAL · LEADING · VALUE-CREATING

PART 1 基础知识

PART 2 配对交易

专业来自101%的投入!

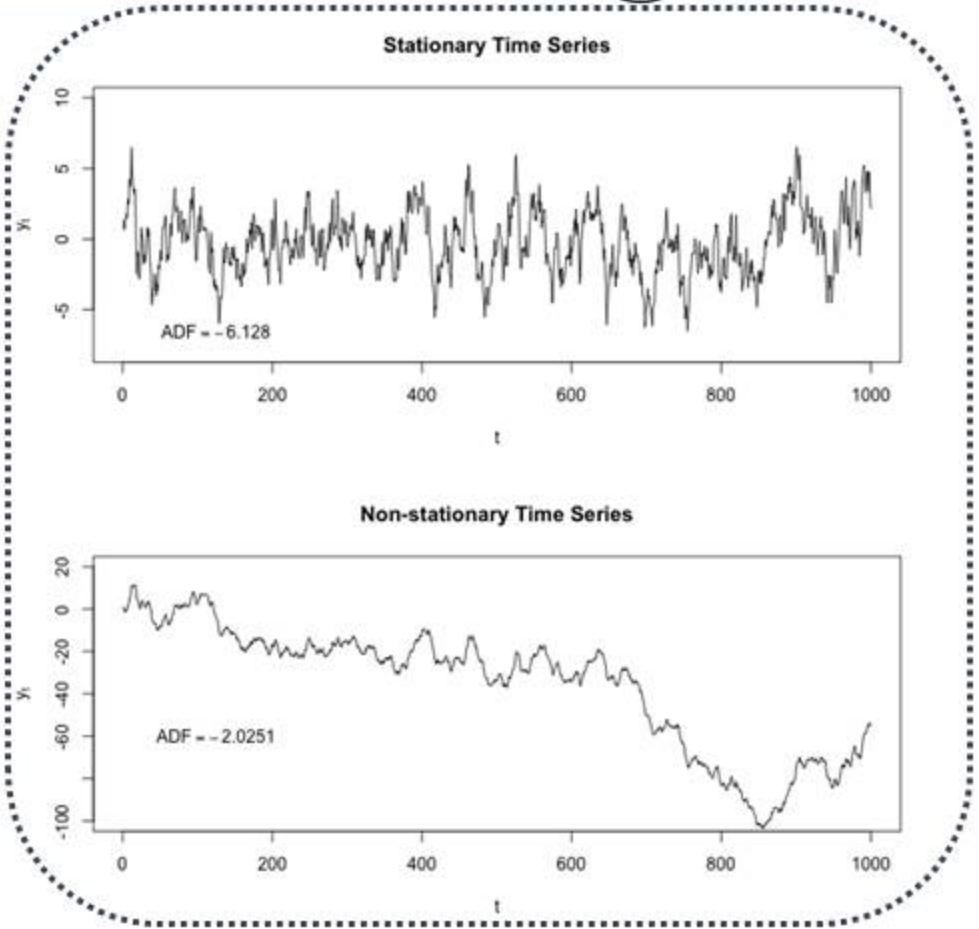
Stationary

$$x_t = b_0 + b_1 x_{t-1} + \xi$$

- **Stationary**
  - Mean and Variance do not change over time;
  - Mean-reverting.

$$x_t = \frac{b_0}{1 - b_1}$$

- **Test:**
  - Unit root
  - DF-test
  - ADF-test



- **Test:**
  - **DF-test**
    - ✓ Start with $x_t=b_0+b_1 x_{t-1}+\varepsilon_t$
    - ✓ Subtract x_{t-1} from both sides $x_t-x_{t-1}=b_0+(b_1-1)x_{t-1}+\varepsilon_t$

      $$x_t-x_{t-1}=b_0+g\,x_{t-1}+\varepsilon_t$$
    - ✓ H_0: $g=0$ (has a unit root and is non-stationary)

      H_a: $g<0$ (does not have a unit root and is stationary)

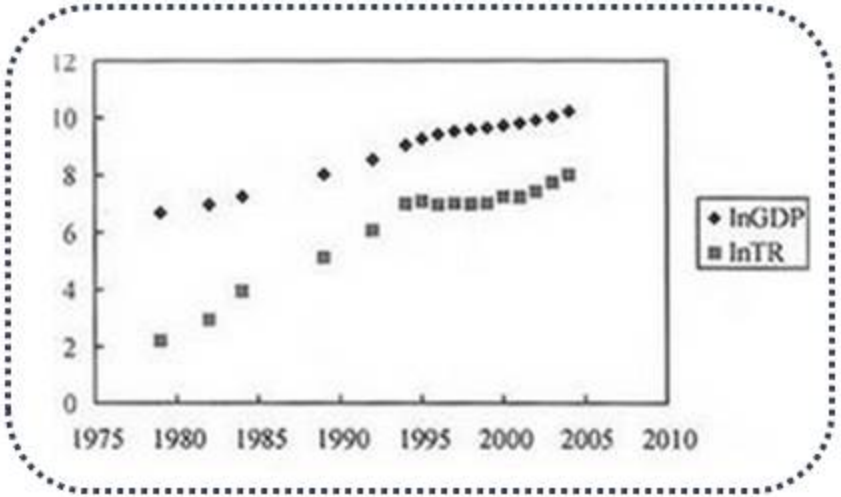
Non-stationary的处理方法

- **Difference**
  - ✓ Define y_t as $y_t = x_t - x_{t-1} = \varepsilon_t$

    $$y_t = b_0 + b_1 y_{t-1} + \varepsilon_t, \text{ where } b_0 = b_1 = 0$$

  - ✓ The first-differenced variable y_t is covariance stationary
- Co-integration

$$y_t = b_0 + b_1 x_{t-1} + \xi$$



- **Co-integration**
  - Two time series are non-stationary;
  - The linear combination of the two time series is stationary.
    - ✓ If co-integrated, can estimate the long-term relation between the two.
  - **Dickey-Fuller Engle-Granger test**
    - ✓ H_0: no co-integration
      H_a: co-integration

CONTENTS

PROFESSIONAL · LEADING · VALUE-CREATING

PART 1 基础知识

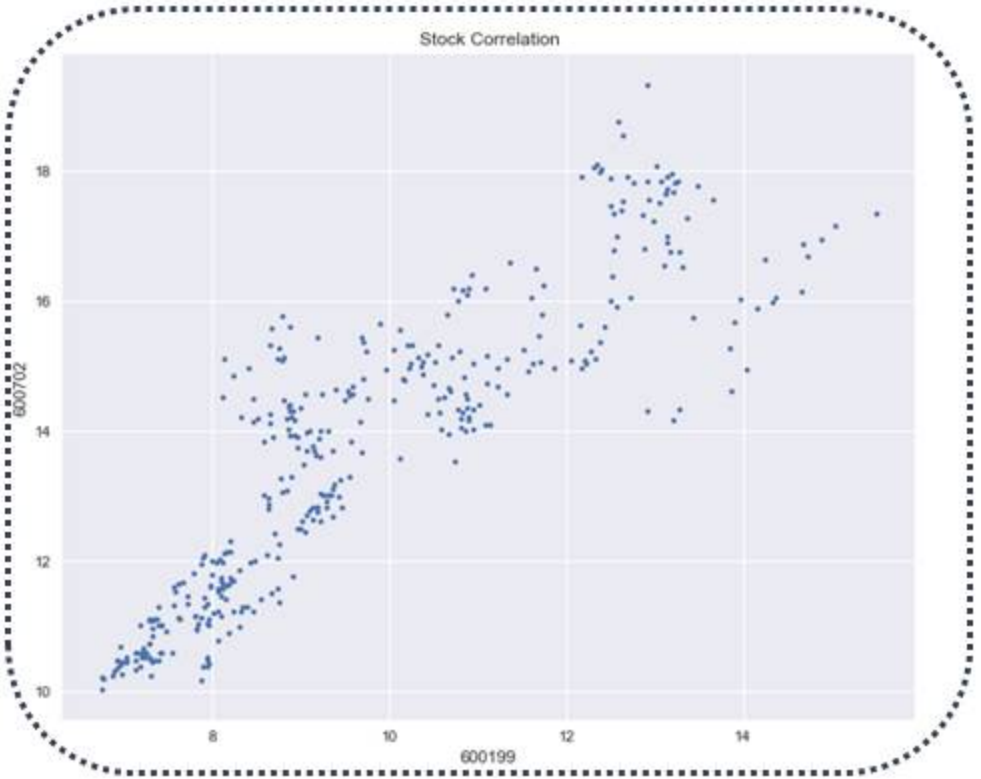
PART 2 配对交易

专业来自101%的投入!

- 基于均值回归
  - 投资的目标不是单一资产，而是两个高度相关资产的价差
  - 高度相关资产
  - 海外
    - ✓ PEP & KO, GS & MS
  - 国内
    - ✓ 茅台&五粮液

价差 (spread)

- Stock X & Stock Y, correlation = 0.95
- We hope that the spread has mean-reverting property
  - Y = a*X +b (linear equation)
  - Y = a*X + b + e
  - Spread = e = Y –a*X - b



Get data

```
data1 = ts.get_k_data('600199', '2013-06-01', '2014-12-31')['close']
```

```
data2 = ts.get_k_data('600702', '2013-06-01', '2014-12-31')['close']
```

```
data = pd.concat([data1, data2], axis=1)
data.head()
```

Spread计算

```
[slope, intercept] = np.polyfit(data.iloc[:,0], data.iloc[:,1], 1).round(2)
data['spread'] = data.iloc[:,1] - (data.iloc[:,0]*slope + intercept)
```

```
data['spread'].plot(figsize = (10,8),title = 'Price Spread')
```

Z_score计算

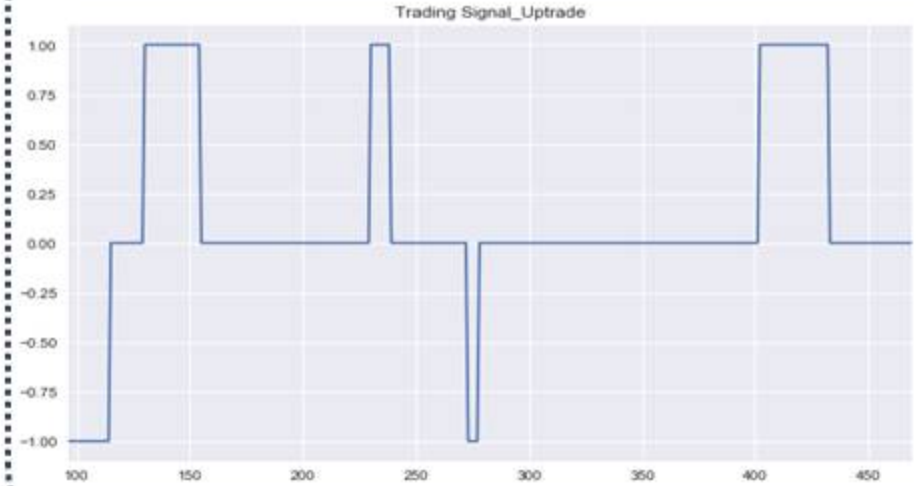
```
data['zscore'] = (data['spread'] - data['spread'].mean())/data['spread'].std()
```

```
data.head()
```

	600199	600702	spread	zscore
96	15.505	17.346	-1.98385	-1.804220
97	14.880	16.956	-1.76760	-1.604541
98	15.043	17.170	-1.71171	-1.552934
99	14.668	16.868	-1.64996	-1.495916
100	14.245	16.634	-1.47365	-1.333116

```
data['zscore'].plot(figsize = (10,8),title = 'Z-score')
plt.axhline(1.5)
plt.axhline(0)
plt.axhline(-1.5)
```

产生交易信号



```
data['returns_1'] = np.log(data['600199'] / data['600199'].shift(1))
```

```
data['returns_2'] = np.log(data['600702'] / data['600702'].shift(1))
```

```
data['strategy'] = 0.5*(data['position_1'].shift(1) * data['returns_1'])+0.5*(data['position_2'].shift(1) * data['returns_2'])
```

```
data[['returns_1','returns_2','strategy']].dropna().cumsum().apply(np.exp).plot(figsize=(10, 8))
```

Limitations

- The spread may not deviate.
- The coefficients of regression must be updated frequently.

Thank you!