

# 《学Python量化，通往财务自由》 | 选股实战之CAPM模型

主要内容：

一、什么是CAPM

二、CAPM的实际应用

三、Python实战CAPM选股

**注意：**这里的讨论仅限于学习Python量化技术，不构成任何投资建议！

Markowitz的均值-方差模型告诉我们如何构建自己的投资组合，并且他本人凭借这一贡献获得了诺贝尔经济学奖。其核心目标是在达成投资目标的前提下，最小化资产的风险。

不过由于其计算量大、难度高、成本高（在当时的条件下），因此部分学者基于Markowitz的框架推导出了资本资产定价模型（CAPM），CAPM可以说奠定了现代投资学的基础。

## 什么是CAPM

经典的CAPM模型如下：

$$E(R_q) - R_f = \beta_{qm}[E(R_m) - R_f]$$

其中：

- $R_m$  是市场投资组合的收益率，也就是整个市场所有风险资产的收益情况。它的作用是作为一个基准，评估目标投资组合相对于基准的提升量。在实际应用中，我们常用大盘指数来作为  $R_m$ 。 $R_q$  代表我们的投资组合的收益率， $R_f$  代表无风险资产收益率，比如国债等固收资产。
- $\beta_{qm}$  是投资组合q的Beta值，这里的 $\beta$ 与我们常提到的 $\beta$ 是一回事， $|\beta|$  大于1代表我们的投资组合比大盘的波动更剧烈， $|\beta|$  小于1代表我们的投资组合的波动小于大盘，如果 $\beta$ 为负，则说明我们的投资组合与大盘的波动方向相反。当然，我们的投资组合也可以是个股
- $E(R_q) - R_f$  是我们的风险投资组合q比无风险资产高出的期望收益率，这部分被我们称作风险溢价。这部分额外的期望收益率是由于我们的投资组合q的风险与大盘不同决定的。

这里额外提一下 $\beta$ 的计算公式：

$$\beta_{qm} = \frac{\sigma(R_q, R_m)}{\sigma^2(R_m)}$$

头条 @数据科学探索

从CAPM模型的公式中可以看出，我们的投资组合的收益与 $\beta$ 之间是线性的关系，当大盘的风险溢价为正时， $\beta$ 越大，我们的收益越大；反之，当大盘的风险溢价为负时， $\beta$ 越大，我们的亏损越大。在实际的投资过程中，机构一般会用对冲的方式抵消大盘带来的系统性风险，从而获得投资组合跑赢大盘部分的收益。

## CAPM的实际应用

Jensen在实际应用过程中，为CAPM模型引入了 $\alpha$ 参数，于是CAPM模型可以写成如下形式：

其中下标“it”代表了这是序列数据，最后一项代表误差项。在前边的CAPM公式中，所有的收益率数据都是期望数据，在实际应用时，我们往往需要使用Jensen的公式来求解出 $\alpha$ 和 $\beta$ 。

我们再来仔细看一下这个公式，是不是发现跟简单线性回归的公式几乎是一样的？没错，我们可以使用最小二乘法的方式直接求解，得出某只股票

的风险溢价相对于大盘风险溢价的关系。

那么这里的 $\alpha$ 怎么理解呢？阿尔法代表了收益率胜过大盘的部分，也就是说，这部分最为考验我们选股的能力，假如选到了走势明显强于大盘的股票，那 $\alpha$ 值就会明显大于0，反之 $\alpha$ 则会小于0。 $\alpha$ 常作为衡量基金经理人绩效的指标。

我们选股的目标，就是在同一风险水平下找最高的 $\alpha$ ，我们再来看CAPM的公式，假如我们的目标收益率已经确定，当 $\alpha$ 越大，那么 $\beta$ 部分就越小，也就是风险越小。这就是Markowitz模型思想的精髓。

关于简单线性回归的最小二乘解法，这里就不赘述了。在我的《三步教你从零掌握简单线性回归》一文中，从理论推导、使用现有工具完成拟合到自己来实现线性回归都有详细介绍，感兴趣的可以去阅读下。

## Python实战CAPM选股

接下来就要到大家最关心的Python实战了。我们先挑选五只股票：万科A、中国平安、贵州茅台、万华化学和科大讯飞，然后我们以沪深300作为市场基准。

## 获取数据并进行简单探索

首先我们要获取数据，并进行一定的清洗：

```
import pandas as pd
import tushare as ts
# 获取数据
pro = ts.pro_api()
wanke = pro.daily(ts_code='000002.SZ', start_date='20170101')
pingan = pro.daily(ts_code='601318.SH', start_date='20170101')
maotai = pro.daily(ts_code='600519.SH', start_date='20170101')
wanhua = pro.daily(ts_code='002415.SZ', start_date='20170101')
keda = pro.daily(ts_code='002230.SZ', start_date='20170101')
hs300 = pro.index_daily(ts_code='000300.SH', start_date='20170101')
# 仅保留收益率数据，且用日期作为index
# 然后按照日期排序（增序）
stock_list = [wanke, pingan, maotai, wanhua, keda, hs300]
for stock in stock_list:
    stock.index = pd.to_datetime(stock.trade_date)
```

```
df = pd.concat([stock.pct_chg / 100 for stock in stock_list], axis=1)
df.columns = ['wanke', 'pingan', 'maotai', 'wanhua', 'keda', 'hs300']
df = df.sort_index(ascending=True)
df.describe()
```

```
import pandas as pd
import tushare as ts

# 获取数据
pro = ts.pro_api()
wanke = pro.daily(ts_code='000002.SZ', start_date='20170101')
pingan = pro.daily(ts_code='601318.SH', start_date='20170101')
maotai = pro.daily(ts_code='600519.SH', start_date='20170101')
wanhua = pro.daily(ts_code='002415.SZ', start_date='20170101')
keda = pro.daily(ts_code='002230.SZ', start_date='20170101')
hs300 = pro.index_daily(ts_code='000300.SH', start_date='20170101')

# 仅保留收益率数据, 且用日期作为index
# 然后按照日期排序 (增序)
stock_list = [wanke, pingan, maotai, wanhua, keda, hs300]
for stock in stock_list:
    stock.index = pd.to_datetime(stock.trade_date)
df = pd.concat([stock.pct_chg / 100 for stock in stock_list], axis=1)
df.columns = ['wanke', 'pingan', 'maotai', 'wanhua', 'keda', 'hs300']
df = df.sort_index(ascending=True)
df.describe()
```

executed in 577ms, finished 12:33:27 2019-01-10

	wanke	pingan	maotai	wanhua	keda	hs300
count	488.000000	493.000000	493.000000	493.000000	493.000000	493.000000
mean	0.000895	0.001233	0.001497	0.001519	0.001422	-0.000091
std	0.025409	0.018448	0.019874	0.024587	0.030133	0.010609
min	-0.092593	-0.066100	-0.100000	-0.099861	-0.100177	-0.047990
25%	-0.012391	-0.008981	-0.009400	-0.011700	-0.013700	-0.005195
50%	-0.000900	0.000300	0.000243	0.000800	0.001000	0.000277
75%	0.010225	0.010806	0.012900	0.013986	0.015200	0.005183
max	0.100100	0.057200	0.082100	0.100157	0.100100	0.043167

头条 @数据科学探索

接下来我们先看下它们的累计收益率和方差情况：

```
df = df.fillna(0)
```

```

returns = (df + 1).product() - 1
print('累计收益率:
', returns)
print('
标准差:
', df.std())

```

```

df = df.fillna(0)
returns = (df + 1).product() - 1
print('累计收益率: \n', returns)
print('\n标准差: \n', df.std())

```

---

executed in 7ms, finished 12:33:32 2019-01-10

```

累计收益率:
wanke      0.324316
pingan     0.688657
maotai     0.898056
wanhua     0.821639
keda       0.613060
hs300     -0.069971
dtype: float64

标准差:
wanke      0.025279
pingan     0.018448
maotai     0.019874
wanhua     0.024587
keda       0.030133
hs300     0.010609
dtype: float64

```

头条 @数据科学探索

我们看下两年来的收益率波动情况：

```

import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib as mpl
sns.set()
mpl.rcParams['font.family'] = 'sans-serif'
mpl.rcParams['font.sans-serif'] = 'SimHei'
plt.figure(figsize=(10, 5))
for col in df.columns:
    plt.plot(df[col], label=col)
plt.title('日收益率时序图(2017至今)', fontsize=20)
plt.legend();

```



日收益率时序图(2017至今)



虽然线条过于密集，但是我们还是能看出来万华化学、科大讯飞的波动很剧烈，但是沪深三百的波动很稳健。

那我们看下累计收益率的时序图：

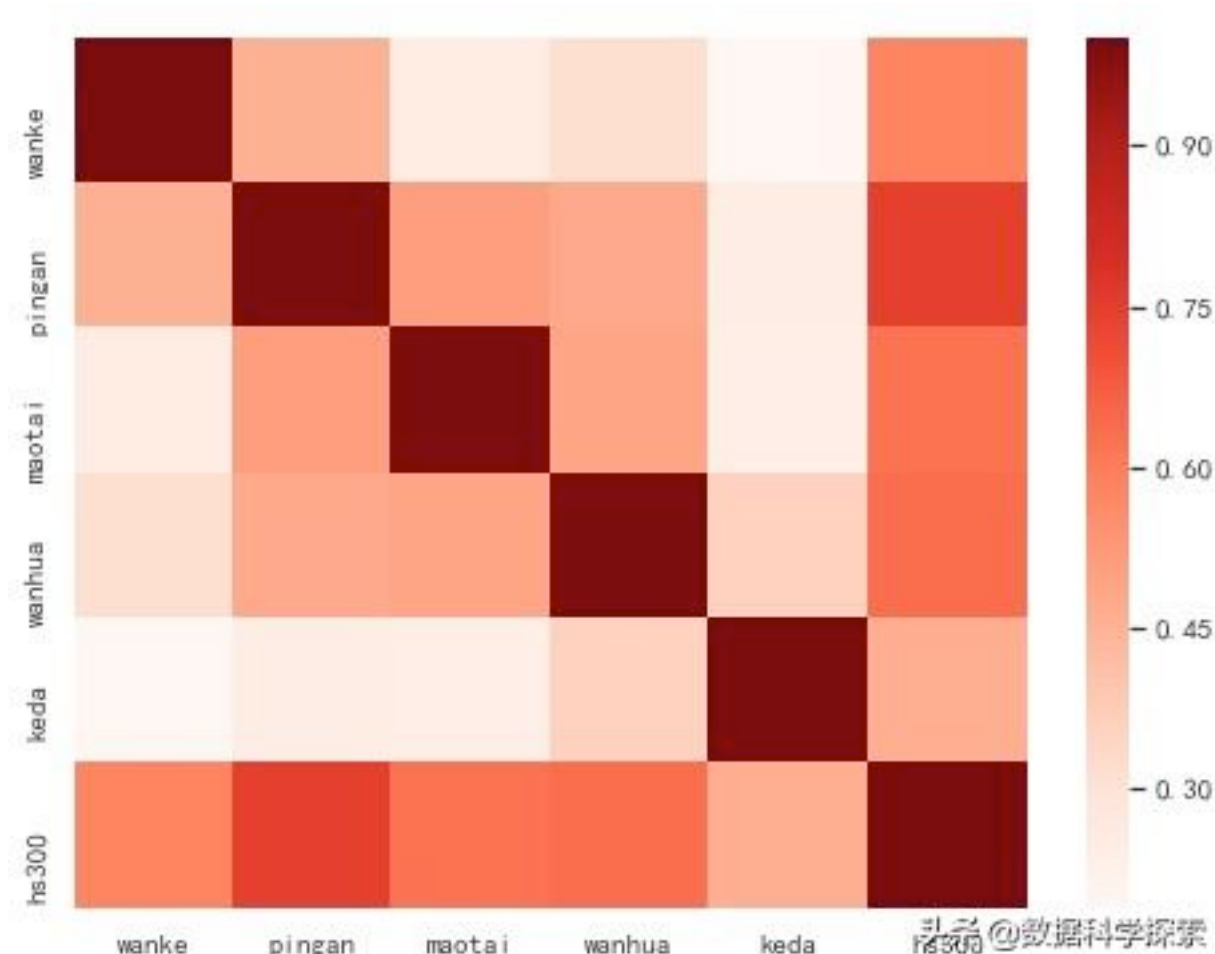
```
plt.figure(figsize=(10, 5))
for col in df.columns:
    plt.plot((df[col]+1).cumprod()-1, label=col)
plt.title('累计收益率时序图(2017至今)', fontsize=20)
plt.legend();
```

累计收益率时序图(2017至今)



符合我们上一张图中获得的预期，万华化学和科大讯飞的波动范围比较大，沪深三百则非常平稳。

我们再来看一下它们之间的相关性：



可以看到，各只股票和沪深三表都有明显的相关性，其中，科大讯飞与沪深三百的关联最弱。

## CAPM选股

接下来，我们就来看一下哪只股票更好。

我们先假设无风险固定收益为3.2%，那么平均每日的无风险收益率为：

```
rf = 1.032 ** (1/360) - 1
print(rf)
```

```
rf = 1.032 ** (1/360) - 1
print(rf)

executed in 4ms, finished 13:16:36 2019-01-10

8.750012529978868e-05
```

然后我们需要计算出这些股票和沪深300各自的风险溢价。

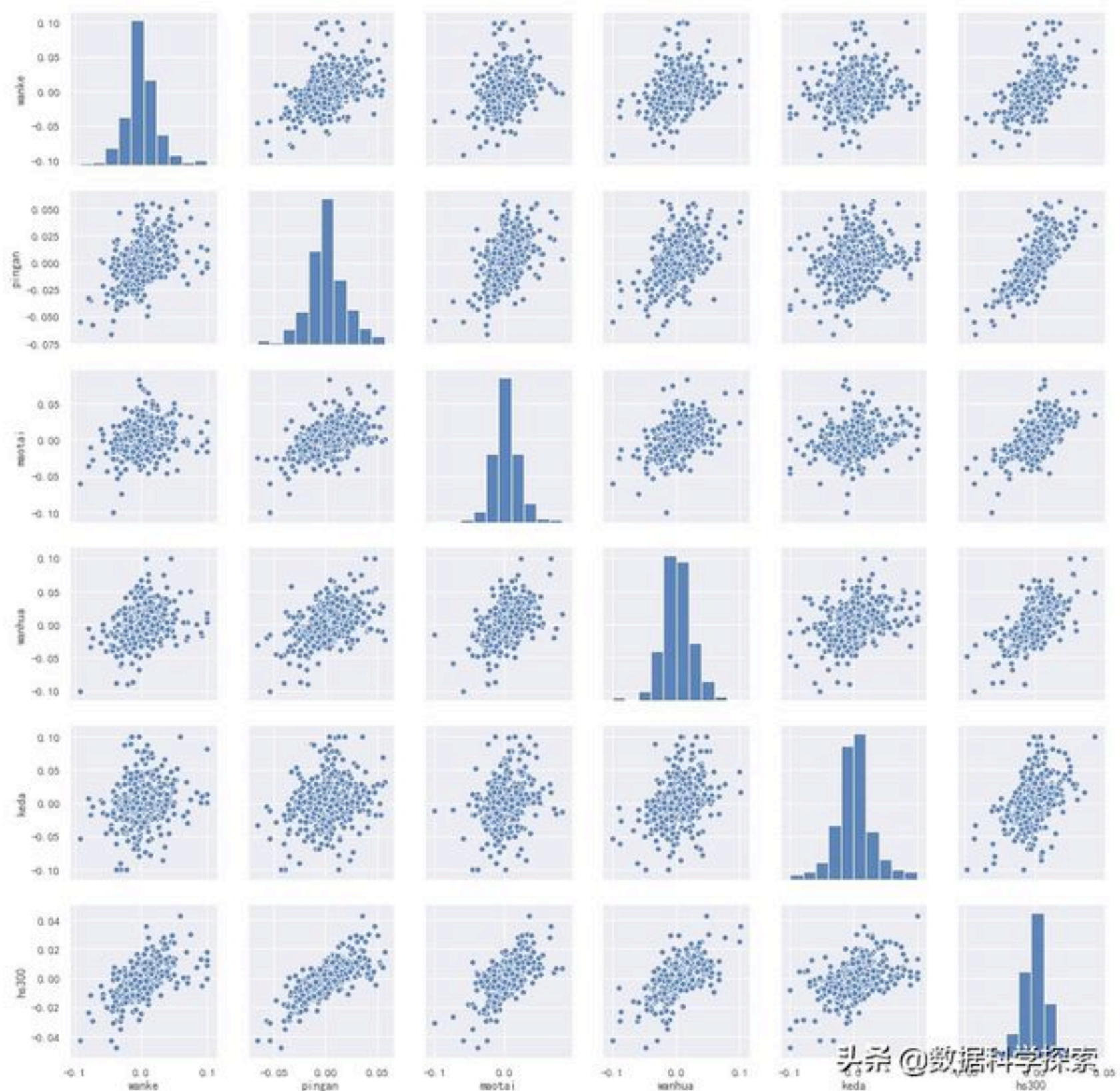
```
df_rp = df - rf
df_rp.head()
```



接下来，我们看下它们各自风险溢价之间的关系：

```
sns.pairplot(df_rp);
```





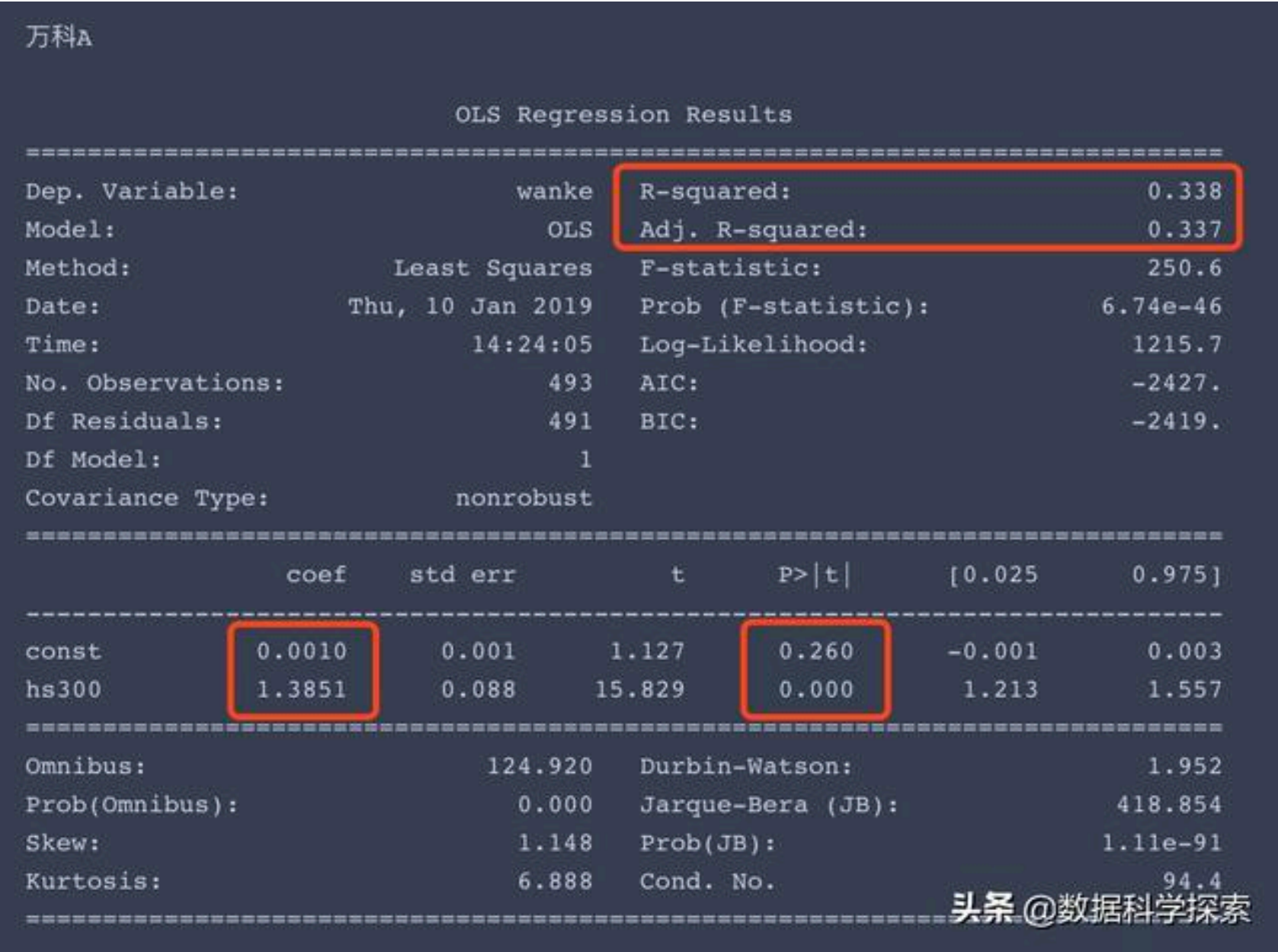
各只股票与沪深三百之间都有很明显的线性关系，那么接下来我们就开始求解了。我们这次使用statsmodels来求解，在这里我们使用sm.add\_constant()方法增加一个常数项，用于求解 $\alpha$ 。

```
import statsmodels.api as sm
stock_names = {
    'wanke': '万科A',
    'pingan': '中国平安',
    'maotai': '贵州茅台',
    'wanhua': '万华化学',
    'keda': '科大讯飞'
}
for stock in ['wanke', 'pingan', 'maotai', 'wanhua', 'keda']:
    model = sm.OLS(df_rp[stock], sm.add_constant(df_rp['hs300']))
    result = model.fit()
```

```
print(stock_names[stock] + '
')
print(result.summary())
print('
')
')
```

万科A

万科A的截距项不显著，即阿尔法为0； $\beta=1.39$ ，即假如大盘涨了10%，万科A预期涨13.9%。R方0.34，拟合效果一般。



中国平安

中国平安的截距项为0.0014，即除了大盘波动带来的收益，其自身价值额外产生了0.14%的收益； $\beta=1.3$ ，即假如大盘涨了10%，中国平安预期涨13%。R方0.56，还不错。



## OLS Regression Results

```

=====
Dep. Variable:          pingan      R-squared:          0.559
Model:                  OLS         Adj. R-squared:     0.558
Method:                 Least Squares  F-statistic:        621.9
Date:                   Thu, 10 Jan 2019  Prob (F-statistic): 2.71e-89
Time:                   14:24:05      Log-Likelihood:     1471.1
No. Observations:      493          AIC:               -2938.
Df Residuals:          491          BIC:               -2930.
Df Model:              1
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	0.0014	0.001	2.492	0.013	0.000	0.002
hs300	1.2999	0.052	24.939	0.000	1.198	1.402

```

=====
Omnibus:                65.332      Durbin-Watson:       1.934
Prob(Omnibus):          0.000      Jarque-Bera (JB):    125.284
Skew:                   0.768      Prob(JB):            6.24e-28
Kurtosis:               4.934      Cond. No.            94.4
=====

```

头条 @数据科学探索

## 贵州茅台

贵州茅台的截距项为0.0016，即除了大盘波动带来的收益，其自身价值额外产生了0.16%的收益； $\beta=1.17$ ，即假如大盘涨了10%，贵州茅台预期涨11.7%。R方0.39，拟合效果一般。

OLS Regression Results

Dep. Variable:	maotai	R-squared:	0.391			
Model:	OLS	Adj. R-squared:	0.390			
Method:	Least Squares	F-statistic:	315.1			
Date:	Thu, 10 Jan 2019	Prob (F-statistic):	8.00e-55			
Time:	14:24:05	Log-Likelihood:	1354.9			
No. Observations:	493	AIC:	-2706.			
Df Residuals:	491	BIC:	-2697.			
Df Model:	1					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]
-----						
const	0.0016	0.001	2.315	0.021	0.000	0.003
hs300	1.1712	0.066	17.751	0.000	1.042	1.301
=====						
Omnibus:	40.493	Durbin-Watson:	1.955			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	164.463			
Skew:	0.194	Prob(JB):	1.94e-36			
Kurtosis:	5.803	Cond. No.	94.4			
=====						

头条@数据科学探索

万华化学

万华化学的截距项为0.0017，即除了大盘波动带来的收益，其自身价值额外产生了0.17%的收益； $\beta=1.47$ ，即假如大盘涨了10%，万华化学预期涨14.7%。R方0.41，拟合效果一般。



## OLS Regression Results

```
=====
Dep. Variable:          wanhua      R-squared:          0.407
Model:                  OLS         Adj. R-squared:     0.405
Method:                 Least Squares  F-statistic:       336.5
Date:                   Thu, 10 Jan 2019  Prob (F-statistic): 1.25e-57
Time:                   14:24:05      Log-Likelihood:    1256.5
No. Observations:      493          AIC:               -2509.
Df Residuals:          491          BIC:               -2501.
Df Model:               1
Covariance Type:       nonrobust
=====
```

```
=====
              coef      std err          t      P>|t|      [0.025      0.975]
-----
const      0.0017      0.001         1.984      0.048      1.67e-05      0.003
hs300      1.4779      0.081        18.345      0.000      1.320      1.636
=====
```

```
Omnibus:          23.297      Durbin-Watson:       1.874
Prob(Omnibus):    0.000      Jarque-Bera (JB):    51.139
Skew:             0.228      Prob(JB):           7.86e-12
Kurtosis:         4.511      Cond. No.           94.4
=====
```

头条 @数据科学探索

## 科大讯飞

科大讯飞的截距项不显著，即 $\alpha$ 为0； $\beta=1.32$ ，即假如大盘涨了10%，贵州茅台预期涨13.2%。R方0.21，拟合效果较差。

## OLS Regression Results

```

=====
Dep. Variable:          keda      R-squared:          0.216
Model:                OLS      Adj. R-squared:       0.214
Method:              Least Squares  F-statistic:        135.3
Date:                Thu, 10 Jan 2019  Prob (F-statistic):    8.64e-28
Time:                14:24:05    Log-Likelihood:      1087.5
No. Observations:    493        AIC:                 -2171.
Df Residuals:        491        BIC:                 -2163.
Df Model:            1
Covariance Type:      nonrobust
=====

```

```

=====
              coef      std err          t      P>|t|      [0.025      0.975]
-----
const          0.0016      0.001      1.305      0.193     -0.001      0.004
hs300          1.3201      0.113     11.632      0.000      1.097      1.543
=====

```

```

Omnibus:          39.227    Durbin-Watson:          1.933
Prob(Omnibus):    0.000    Jarque-Bera (JB):       110.308
Skew:            0.353    Prob(JB):               1.11e-24
Kurtosis:        5.207    Cond. No.               94.4
=====

```

头条 @数据科学探索

可以看到，中国平安的拟合效果最好；这几只股票的阿尔法系数都不太高或者为0，即收益基本上来自于跟随大盘的波动；万华化学的 $\alpha$ 系数和 $\beta$ 系数均为最高，在牛市中不失为一个很好的选择。

这里的结果跟我们选择的周期以及基准指数相关性非常大，比如说沪深300事实上是一些优质公司的集合，上边列出的公司也属于优质的公司，过去两年中白马蓝筹基本上是主线，所以沪深300指数并不能代表整个市场的收益情况，这也是为什么我们的 $\alpha$ 系数为什么这么小，优质的公司比优质公司的均值，再加上这几个公司基本上是大部头，自然差不了太多。建议大家用中证500、中证1000、中证流通等指数再来看一下，也许会得到不同的结果哦。

CAPM模型也存在一定的问题，比如它假设了市场未来一段时间和历史中的波动情况是一致的，因此用历史数据计算得到的 $\alpha$ 和 $\beta$ 不一定能很好地预测未来的收益情况。之后我们还会有更多种选股策略提供给大家。