

Q1. CAPM 模型中的 alpha 值，经常被投资者用来作为投资决策的依据，一般来讲投资者会选择持有 alpha 较大的股票，现在，假设市场投资组合的收益率为 10%，无风险利率为 4%，A 公司的 β 值为 1.2，预期收益率为 9%，而 B 公司的 β 值为 1.3，预期收益率为 12%，那么投资者应该选择那只股票？

A 公司的 alpha 值：

$$9 = \alpha_A + 4 + 1.2 \times (10 - 4)$$

$$\alpha_A = -2.2$$

B 公司的 alpha 值：

$$12 = \alpha_B + 4 + 1.3 \times (10 - 4)$$

$$\alpha_B = 6.7$$

$$\alpha_B > \alpha_A$$

投资者应该选 B 公司的 股票。

Q2. 获取中国农业银行 2014 年的股票数据，并建立 CAPM 模型，市场组合收益率用本次课程的数据，无风险利率为 3.6%。

下载农业银行的数据代码为：

```
import pandas_datareader.data as web
nyyh = web.DataReader('601288.SS','yahoo',
dt.datetime(2014,1,1),dt.datetime(2014,12,31))
```

```
In [3]: #爬取数据
nyyh = web.DataReader('601288.SS','yahoo', datetime.datetime(2014,1,1),datetime.datetime(2014,12,31))
nyyh.index = pd.to_datetime(nyyh.index)

#删除交易量等于 0 的数据
nyyh = nyyh[nyyh.Volume !=0]
nyyh['return'] = (nyyh['Adj Close'] - nyyh['Adj Close'].shift(1))/nyyh['Adj Close'].shift(1)

#保留收益率变量
nyyh = nyyh['return']
nyyh.dropna(inplace=True)
```

2、获得市场组合收益数据：

```
In [4]: indexcd = pd.read_csv("TRD_Index.csv", index_col = 'Trddt')
mktcd = indexcd[indexcd.Indexcd ==902]
mktret = pd.Series(mktcd.Retindex.values,index = pd.to_datetime(mktcd.index))
mktret.name= 'market'
mktret = mktret['2014-01-01':'2014']
```

3、数据合并

```
In [5]: Ret = pd.merge(pd.DataFrame(mktret),pd.DataFrame(nyyh),left_index=True,right_index=True, how = 'inner')
```

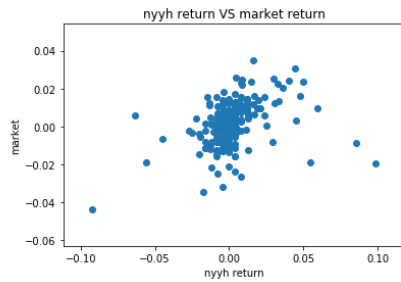
4、计算无风险收益率

```
In [6]: rf = 1.036**(1/365) -1
Ret['risk_premium'] = Ret['market'] - rf
```

5、画图

```
In [7]: plt.scatter(Ret['return'],Ret['market'])
plt.xlabel('nyyh return'); plt.ylabel('market')
plt.title('nyyh return VS market return')
```

```
Out[7]: Text(0.5,1,'nyyh return VS market return')
```



6、拟合曲线，找到 beta

```
In [8]: import statsmodels.api as sm

Ret['constant'] = 1 #增加截距项
X = Ret[['constant', 'risk_premium']]
Y = Ret['return']
model= sm.OLS(Y,X)
result =model.fit()
print(result.summary())
```

```

OLS Regression Results
=====
Dep. Variable:          return    R-squared:                0.118
Model:                  OLS      Adj. R-squared:            0.114
Method:                 Least Squares   F-statistic:             32.32
Date:                  Sun, 28 Oct 2018   Prob (F-statistic):      3.75e-08
Time:                  20:12:14         Log-Likelihood:          658.47
No. Observations:      244             AIC:                    -1313.
Df Residuals:          242             BIC:                    -1306.
Df Model:               1
Covariance Type:       nonrobust
=====
               coef    std err          t      P>|t|      [0.025    0.975]
-----
constant      0.0010    0.001      0.962    0.337    -0.001     0.003
risk_premium   0.5341    0.094     5.685    0.000     0.349     0.719
=====
Omnibus:                 129.964   Durbin-Watson:           1.955
Prob(Omnibus):            0.000   Jarque-Bera (JB):        2107.676
Skew:                     1.681   Prob(JB):                 0.00
Kurtosis:                 17.000   Cond. No.                 89.7
=====
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

/home/testuser/anaconda3/lib/python3.6/site-packages/statsmodels/compat/pandas.py:56: FutureWarning: The pandas.core.datetools module is deprecated and will be removed in a future version. Please use the pandas.tseries module instead.
from pandas.core import datetools

结论:

斜率为 0.5341, p值小于0.05, 判定为显著
截距为 0.001, p值小于0.05, 判定为不显著

农业银行的 capm 模型为:
 $r-r_f=0.001+r_f+0.5341 \times (r_m-r_f)$

Q3

假设某股票的三因子模型为:

$$R_i - R_f = 0.01 + 1.2(R_M - R_f) + 0.5SMB + 0.1HML$$

(1) 该股票的异常收益率为多少?

(2) 当 R_f 为 0.5% 且 R_M 、 R_{SMB} 、 R_{HML} 分别为 2%、2.4%、1.8% 时, 该股票的预期收益率为多少?

Q3.

假设某股票的三因子模型为：

$$R_i - R_f = 0.01 + 1.2(R_M - R_f) + 0.5SMB + 0.1HML$$

(1) 该股票的异常收益率为多少？

(2) 当 R_f 为 0.5% 且 R_M 、 R_{SMB} 、 R_{HML} 分别为 2%、2.4%、1.8% 时，该股票的预期收益率为多少？

(1) 股票异常收益为 0.01

(2) $R_i - 0.5\% = 0.01 + 1.2 \times (2\% - 0.5\%) + 0.5 \times 2.4\% + 0.1 \times 1.8\%$
 $R_i = 3.69\%$

Q4.

读取 problem21.txt 文件中中远航运 2014 年股价数据以及 ThreeFactors.txt 文件中的 2014 年三因子数据，按照相关步骤建立三因子模型。

1、

```
In [9]: #读取数据，计算收益率
zyhy=pd.read_table('problem21.txt', sep='\t', index_col='Date')
zyhy['return'] = (zyhy['zyhy'] - zyhy['zyhy'].shift(1))/zyhy['zyhy'].shift(1)
ret = zyhy['return']
zyhy['return'].dropna(inplace=True)
ThreeFactors=pd.read_table('ThreeFactors.txt', sep='\t', index_col='TradingDate')

#将索引转化为时间格式
ThreeFactors.index=pd.to_datetime(ThreeFactors.index)
ThrFac=ThreeFactors['2014-01-02:'] #截取 2014 年 1 月 2 号以后的数据
ThrFac=ThrFac.iloc[:,[2,4,6]] #提取对应的 3 个因子

#合并股票收益率和 3 因子的相关数据
zyThrFac=pd.merge(pd.DataFrame(zyhy['return']),
                  pd.DataFrame(ThrFac),
                  left_index=True,
                  right_index=True)
```

2、作图

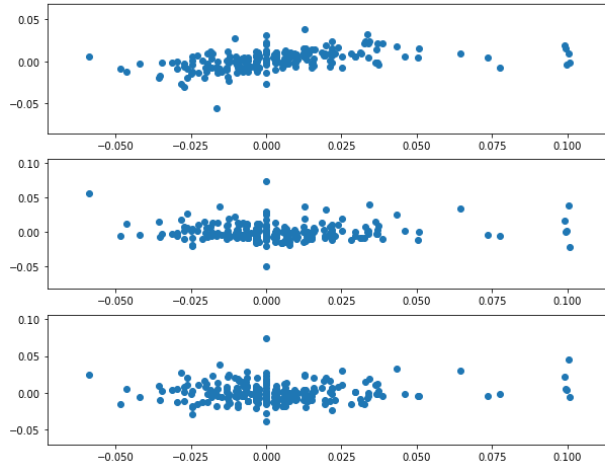
```
In [10]: plt.figure(figsize=(10,8))

plt.subplot(3,1,1)
plt.scatter(zyThrFac['return'], zyThrFac.RiskPremium2)

plt.subplot(3,1,2)
plt.scatter(zyThrFac['return'], zyThrFac.SMB2)

plt.subplot(3,1,3)
plt.scatter(zyThrFac['return'], zyThrFac.HML2)

plt.show()
```



```
In [11]: #回归
import statsmodels.api as sm

regThrFac=sm.OLS(zyThrFac['return'], sm.add_constant(zyThrFac.iloc[:,1:4]))
result=regThrFac.fit()
result.summary()
```

Out[11]: OLS Regression Results

Dep. Variable:	return	R-squared:	0.223			
Model:	OLS	Adj. R-squared:	0.213			
Method:	Least Squares	F-statistic:	23.01			
Date:	Sun, 28 Oct 2018	Prob (F-statistic):	3.94e-13			
Time:	20:12:14	Log-Likelihood:	598.75			
No. Observations:	245	AIC:	-1190.			
Df Residuals:	241	BIC:	-1175.			
Df Model:	3					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	0.0021	0.001	1.507	0.133	-0.001	0.005
RiskPremium2	0.9948	0.126	7.893	0.000	0.746	1.243
SMB2	-0.7332	0.223	-3.288	0.001	-1.172	-0.294
HML2	0.7493	0.211	3.559	0.000	0.335	1.164
Omnibus:	105.816	Durbin-Watson:	1.766			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	440.407			
Skew:	1.763	Prob(JB):	2.33e-96			
Kurtosis:	8.542	Cond. No.	220.			

结论：

const 的 p 值为 0.133，大于 0.05，判定为不显著
RiskPremium2 的 p 值为 0.000，小于 0.05，判定为显著
SMB2 的 p 值为 0.001，小于 0.05，判定为显著
HML2 的 p 值为 0.000，小于 0.05，判定为显著