**A题获取数据方法【Python版】**

**若您使用Python进行研究**

请了解点宽网**数据字典说明文档和Python API说明文档。数据字典**里面有BP因子数据说明，这部分将是进行机器学习优化股票多因子模型的特征数据。**Python API**说明文档里面有获取数据和策略编写的各种函数方法，**本赛题将使用这些函数方法进行提取数据和策略编写。**这些函数方法您使用Python环境进行使用即可。

如下为A题获取数据的方法的简单案例：

打开AT和Python IDE环境。编写如下Python代码，并在Python IDE环境运行即可。

atrader库下载安装方法：

|  |
| --- |
| pip install atrader |

【注】A题数据获取方法为：使用函数API获取数据

具体函数API介绍请到点宽网->AT相关->Python API说明文档查看：

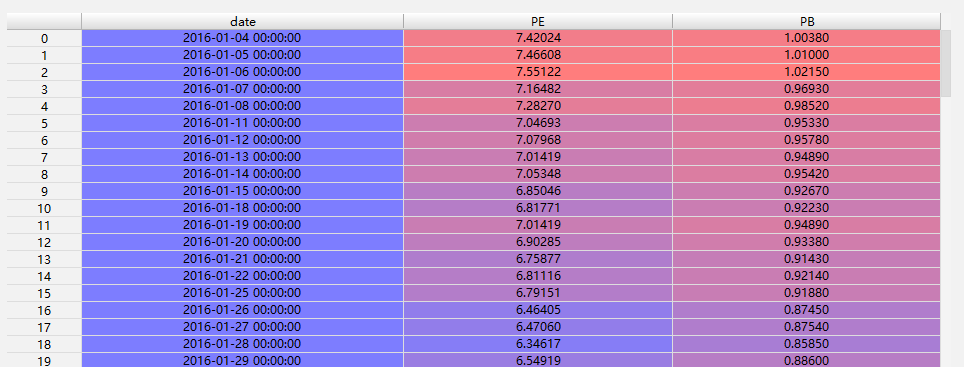
<https://www.digquant.com.cn/documents/17>

**（一）模型研究阶段获取数据**

**（1）方法一：利用get\_factor\_by\_code - 获取BP因子数据（单个股票）**

|  |
| --- |
| #导入atrader库：  import atrader as at  #获取BP因子数据（单个股票）：  factor\_data=at.get\_factor\_by\_code(factor\_list=['PE', 'PB'], target='SZSE.000001', begin\_date='2016-01-01', end\_date='2016-07-02')  #打印输出结果：  print(factor\_data) |

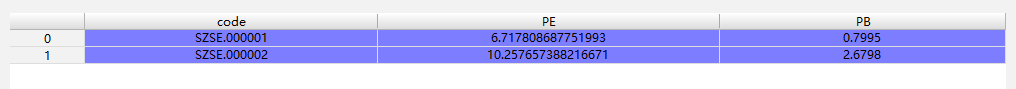
结果如下：



**（2）方法二：利用get\_factor\_by\_day - 获取BP因子数据（单天）**

|  |
| --- |
| #导入atrader库：  import atrader as at  #获取BP因子数据（单个股票）：  factor\_data=at.get\_factor\_by\_day(factor\_list=['PE', 'PB'], target\_list=['SZSE.000001','SZSE.000002'], date='2016-06-01')  #打印输出结果：  print(factor\_data) |

结果如下：



**（3）方法三：利用get\_factor\_by\_factor - 获取BP因子数据（单个因子）**

|  |
| --- |
| #导入atrader库：  import atrader as at  #获取BP因子数据（单个股票）：  factor\_data=at.get\_factor\_by\_factor(factor='PE', target\_list=['SZSE.000001','SZSE.000002'], begin\_date='2016-01-01', end\_date='2016-07-02')  #打印输出结果：  print(factor\_data) |

结果如下：



**策略内使用方法在下一页↓**

**（二）策略内获取数据**

|  |
| --- |
| # -\*- coding: utf-8 -\*-  """  策略内获取数据方法  """  from atrader import \*  #初始化函数：  def init(context: Context):  #注册行情数据：  reg\_kdata('day', 1)  # 注册因子数据  reg\_factor(['PE', 'PB', 'MA10'])  #参数设置：  context.Len=10  def on\_data(context: Context):  #获取K线数据：  KData = get\_reg\_kdata(reg\_idx=context.reg\_kdata[0], length=context.Len, fill\_up=True, df=True)  print(KData)  #获取因子数据：  FData = get\_reg\_factor(reg\_idx=context.reg\_factor[0],length=context.Len, df=True) # 获取因子数据  print(FData)  if \_\_name\_\_ == '\_\_main\_\_':  #A股股票：  Acode=get\_code\_list('sse\_a,szse\_a')  target\_list=list(Acode['code'])  run\_backtest(strategy\_name='策略内获取数据方法', file\_path='策略内获取数据方法.py', target\_list=target\_list, frequency='day', fre\_num=1, begin\_date='2016-01-01', end\_date='2016-06-30') |

**策略内获取数据及策略框架如下图所示：**



**策略内数据：**

这里就不打印出来了，大家可以调试运行查看，调试的时候请点击多几次调试的【继续】按钮，查看获取的数据存储格式。原因在于您只能知道数据存储格式，才能获取到你需要的数据进行策略逻辑计算。

