## 第四章 Python 数据分析模块—Pandas

## 本部分主要介绍Python中的数据分析模块：Pandas的使用。数据分析需要一个“容器”装载。而大部分的结构化的数据都可以装载入一个类似于“表”的结构当中。而那些“非结构化”的数据也可以通过特征工程等装载入这个“容器”中。Python中的Pandas模块就是为了构建这种通用构建数据分析“容器”，并对数据进行清洗，合并，整理等的一系列函数的组合。学习Pandas包打开了数据科学处理数据的大门。

## 第一节 Pandas 简介

* Series
* DataFrame

1.Series:类似一维数组的对象。有两个重要属性：values 和 index

* Series 的创建
* 索引
* 对齐

import pandas as pd

x = pd.Series([1,3,5,7,9])

x.values

x.index

2. DataFrame：Pandas 最重要的数据结构，用途非常广泛

* DataFrame 的创建
* data view
* 索引

(1)dates1=pd.date\_range('20140101',periods=7)#

dates1

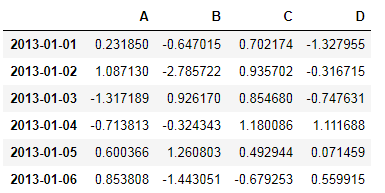
DatetimeIndex(['2014-01-01', '2014-01-02', '2014-01-03', '2014-01-04',

'2014-01-05', '2014-01-06', '2014-01-07'],

dtype='datetime64[ns]', freq='D')

（2）import numpy as np

df = pd.DataFrame(np.random.randn(6,4), index=dates, columns=list('ABCD'))



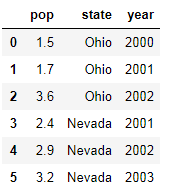
data = {'state': ['Ohio', 'Ohio', 'Ohio', 'Nevada', 'Nevada', 'Nevada'],

'year': [2000, 2001, 2002, 2001, 2002, 2003],

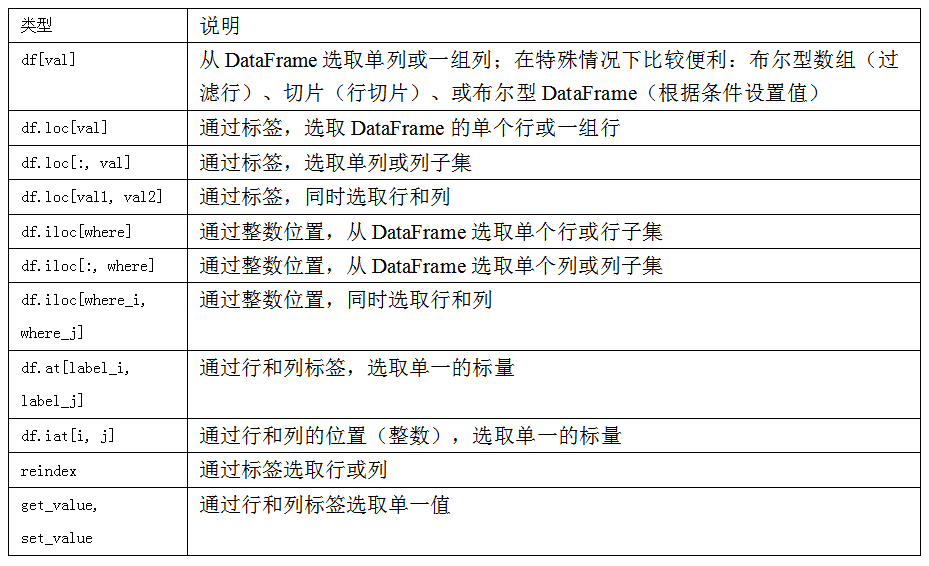
'pop': [1.5, 1.7, 3.6, 2.4, 2.9, 3.2]}

（3）frame = pd.DataFrame(data)

frame



（4）数据筛选



（5）Boolean indexing

df1.index = ['1','2','3','4','5','6','7']#改变索引名称

（6）一些运算

df1.apply(np.mean)#求均值

df1.mean()#求均值

（7）几种常用的操作

* apply
* merge
* grouping

numpy 的真对元素的函数课作用在df上

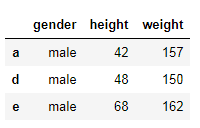
df1.apply(sum)#求和

（8）**Grouping**

* **Splitting** the data into groups based on some criteria
* **Applying** a function to each group independently
* **Combining** the results into a data structure

分组grouped = df.groupby('gender')

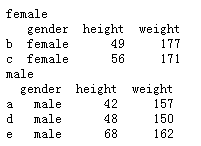
grouped.get\_group('male')



分组展示for name,group in grouped:

print(name)

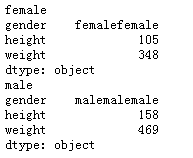
print(group)



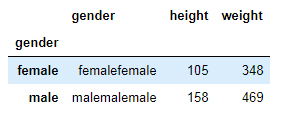
for name,group in grouped:

print(name)

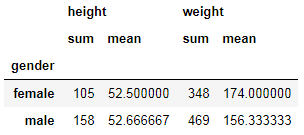
print(group.sum())



求和grouped.apply(sum)



聚合grouped.aggregate([np.sum,np.mean])



时间序列

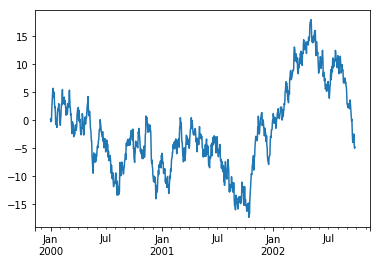
rng = pd.date\_range('1/1/2012', periods=180, freq='S')

ts = pd.Series(np.random.randn(1000),

index=pd.date\_range('1/1/2000', periods=1000))

%matplotlib inline

ts.plot()

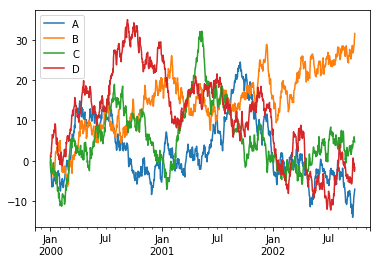


df = pd.DataFrame(np.random.randn(1000, 4), index=ts.index,

columns=['A', 'B', 'C', 'D'])

df = df.cumsum()

df.plot()



**非常重要：**

数据的存储和读取

* df.to\_csv()
* pd.read\_csv()
* df.to\_excel()
* pd.read\_excel()

爬虫技术1：

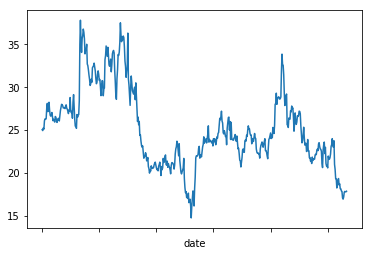
导入财经类数据（股票为例）

import tushare as ts

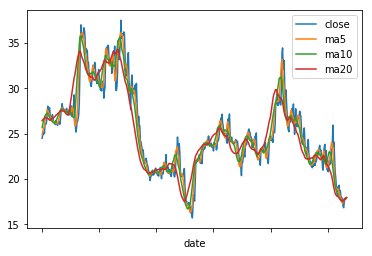
sk = ts.get\_hist\_data('600848')

%matplotlib inline

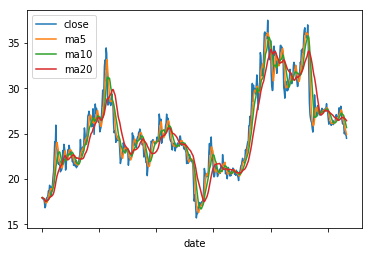
sk['open'].plot()



sk[['close','ma5','ma10','ma20']].plot()



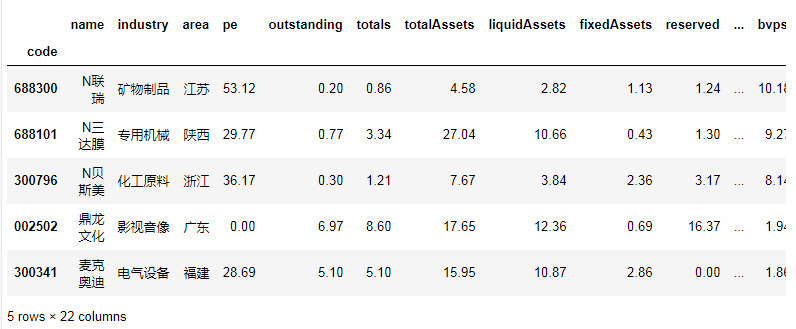
sk[['close','ma5','ma10','ma20']].sort\_index().plot()



sk.to\_csv('sk600848.csv')#到处CSV格式数据表格

info = ts.get\_stock\_basics()

info.head()



**作业**

* 研究一些股票，找出相关系数大于0.8的股票,若没有，计算最大的相关系数
* 假设初始资金100万，设计投资策略，并比较策略和大盘的收益
* 画出资金每日增长／减少曲线[一条是投资策略的，一条是按大盘走势的]
* 一些提示

x = ts.get\_hist\_data('688018')#获取股票数据集

y = ts.get\_hist\_data('600122')

x.close.corr(y.close)#相关系数

sk.to\_csv('600122.csv')#导出数据为CSV格式