Pandas 教程

1. 课程介绍

课程内容和目标

什么是 Pandas

Pandas 是 Python 里分析结构化数据的工具集

基础是 numpy：高性能矩阵运算
图形库 matplotlib: 提供数据可视化

Pandas 能做什么

结构化数据分析

-》数据挖掘

例子

1. 学生成绩分析
2. 股票数据分析

课程内容

使用 Python 进行数据分析

- ipython 介绍
- Pandas 快速入门
- Pandas 核心数据结构和基础运算
    - DataFrame, Series
- Pandas 高级内容
    - 索引和数据选择

    - 分组统计
    - 时间序列
    - 数据IO
- 数据可视化
- 实例
    - 股票数据分析
    - 时间事件日志

课程目标

会使用 Pandas 进行简单的数据分析

ipython

开发环境搭建

https://www.python.org/downloads/ (https://www.python.org/downloads/)
python 2.7.10

http://jupyter.org/ (http://jupyter.org/)
pip install jupyter
pip install numpy
pip install matplotlib

Windows需要额外安装 pyreadline:
pip install pyreadline

技巧

python 命令行与 ipython命令行的区别

显示的数据可读性更强

import numpy as np
from numpy.random.randn

data = {i: randn(i} for i in range(6)}

命令补全

Shell 命令

```
* 大部分直接可工作 cd, ls, pwd
* 少部分可在前面加 ! 号 !rename
```

内省：类或变量加问号

魔术命令

```
* %run 命令
* %timeit 命令
* %quickref 显示快速参考文档
* %magic 显示魔术命令列表和文档
```

ipython notebook

在web上进行探索性编程，内联图片显示

启动

ipython notebook

特点

```
* web 上的 ipython
* 显示内联图片
* 导出数据和分析过程
```

2. Pandas 快速入门

10 Minutes to pandas

http://pandas.pydata.org/pandas-docs/stable/10min.html
(http://pandas.pydata.org/pandas-docs/stable/10min.html)

2.1 numpy 简介

高性能科学计算和数据分析的基础包，是所有高级数据分析工具的构建基础。

面向数组的思维模式。

ndarray

ndarray

```
* 多维数组
* 多维数组的运算
```

创建

```
* 一维数组
* 二维数组
```

属性

```
* shape
* dtype
```

数组索引

使用数组来索引

使用布尔值索引

数学运算

矩阵运算

其他常用函数

2.2 快速入门一

10 Minutes to pandas

http://pandas.pydata.org/pandas-docs/stable/10min.html
(http://pandas.pydata.org/pandas-docs/stable/10min.html)

最权威的快速入门文档，我们的视频质量肯定没有这个文档好。

这个视频的价值：提供低门槛，轻松的学习环境

创建 pandas 对象

查看数据

选择数据

2.3 快速入门二

处理丢失数据

数据运算

数据合并

数据分组

2.4 快速入门三

数据整形

数据透视

时间序列

数据可视化

数据载入与保存

2.5 数据分析实例：分析 MovieLens 电影数据

grouplens.org/datasets/movielens

3. Pandas 基础

3.1 核心数据结构

Series

创建

Series 是一维带标签的数组，数组里可以放任意的数据（整数，浮点数，字符串，Python Object）。其基本的创建函数是：

s = pd.Series(data, index=index)

其中 index 是一个列表，用来作为数据的标签。data 可以是不同的数据类型：

- Python 字典
- ndarray 对象
- 一个标量值，如 5

特性

Series 对象的性质

- 类 ndarray 对象
- 类 dict 对象
- 标签对齐操作

DataFrame

创建

DataFrame 是二维带行标签和列标签的数组。可以把 DataFrame 想你成一个 Excel 表格或一个 SQL 数据库的表格，还可以相像成是一个 Series 对象字典。它是 Pandas 里最常用的数据结构。
创建 DataFrame 的基本格式是：

df = pd.DataFrame(data, index=index, columns=columns)

其中 index 是行标签，columns 是列标签，data 可以是下面的数据：

- 由一维 numpy 数组，list，Series 构成的字典
- 二维 numpy 数组
- 一个 Series
- 另外的 DataFrame 对象

特性

- 列选择/增加/删除
- 使用 assign() 方法来插入新列
- 索引和选择
  - 选择一列 -> df[col] -> Series
  - 根据行标签选择一行 -> df.loc[label] -> Series
  - 根据行位置选择一行 -> df.iloc[label] -> Series
  - 选择多行 -> df[5:10] -> DataFrame
  - 根据布尔向量选择多行 -> df[bool_vector] -> DataFrame
- 数据对齐
- 使用 numpy 函数

Panel

Panel 是三维带标签的数组。实际上，Pandas 的名称由来就是由 Panel 演进的，即 pan(el)-da(ta)-s。Panel 比较少用，但依然是最重要的基础数据结构之一。

items: 坐标轴 0，索引对应的元素是一个 DataFrame
major_axis: 坐标轴 1, DataFrame 里的行标签
minor_axis: 坐标轴 2, DataFrame 里的列标签

3.2 基础运算

重新索引

丢弃部分数据

映射函数

apply

applymap

排序和排名

数据唯一性及成员资格

3.3 索引

行索引

列索引

索引类

重复索引

多层索引

创建

索引交换

按照索引层次进行统计

索引与列的转换

4. Pandas 高级内容

4.1 分组运算

分组计算

拆分 -》应用 -》合并

对 Series 分组

对 DataFrame 分组

分组中的元素个数统计

对分组进行迭代

分组转化为字典

按列分组

其他分组方法

通过字典进行分组

通过函数分组

通过索引级别进行分组

4.2 聚合统计

数据聚合

内置聚合函数

自定义聚合函数 agg

一次性应用多个聚合函数

不同的列应用不同聚合函数

重置索引

分组运算和转换

分组数据变换 transform

自定义数据处理 apply

数据过滤 filter

4.3 数据IO

- 索引：将一个列或多个列读取出来构成 DataFrame，其中涉及是否从文件中读取索引以及列名
- 类型推断和数据转换：包括用户自定义的转换以及缺失值标记
- 日期解析
- 迭代：针对大文件进行逐块迭代。这个是Pandas和Python原生的csv库的最大区别
- 不规整数据问题：跳过一些行，或注释等等

索引及列名

缺失值处理

逐块读取数据

保存数据到磁盘

二进制数据

二进制的优点是容量小，读取速度快。缺点是可能在不同版本间不兼容。比如 Pandas 版本升级后，早期版本保存的二进制数据可能无法正确地读出来。
pickle

其他格式简介

其他格式

- HDF5: HDF5是个C语言实现的库，可以高效地读取磁盘上的二进制存储的科学数据。
- Excel文件: pd.read_excel/pd.ExcelFile/pd.ExcelWriter
- JSON: 通过 json 模块转换为字典，再转换为 DataFrame
- SQL 数据库：通过 pd.io.sql 模块来从数据库读取数据
- NoSQL (MongoDB) 数据库：需要结合相应的数据库模块，如 pymongo 。再通过游标把数据读出来，转换为 DataFrame

4.4 时间序列

时间日期

- 时间戳 tiimestamp：固定的时刻 -> pd.Timestamp
- 固定时期 period：比如 2016年3月份，再如2015年销售额 -> pd.Period
- 时间间隔 interval：由起始时间和结束时间来表示，固定时期是时间间隔的一个特殊

时间日期在 Pandas 里的作用

- 分析金融数据，如股票交易数据
- 分析服务器日志

python 里的 datetime

Pandas 里的时间序列

日期范围

生成日期范围

时间频率

时期及算术运算

时期序列

时期的频率转换

asfreq

- A-DEC: 以 12 月份作为结束的年时期
- A-NOV: 以 11 月份作为结束的年时期
- Q-DEC: 以 12 月份作为结束的季度时期

季度时间频率

Timestamp 和 Period 相互转换

重采样

- 高频率 -> 低频率 -> 降采样：5 分钟股票交易数据转换为日交易数据
- 低频率 -> 高频率 -> 升采样
- 其他重采样：每周三 (W-WED) 转换为每周五 (W-FRI)

OHLC 重采样

通过 groupby 重采样

升采样和插值

时期重采样

性能

时间日期解析

5 数据可视化

Pandas 的数据可视化使用 matplotlib 为基础组件。更基础的信息可参阅 matplotlib 相关内容。本节主要介绍 Pandas 里提供的比 matplotlib 更便捷的数据可视化操作。

线型图

柱状图

直方图

概率密度图

散布图

饼图

高级绘图函数

1. Pandas 高级绘图函数放在 pandas.tools.plotting 包里。
2. 定制化的绘图需要学习 matplotlib 这个包的用法

6 实例

6.1 实例：股票数据分析

https://github.com/kamidox/stock-analysis (https://github.com/kamidox/stock-analysis)

股票数据获取

股票波动幅度分析

波动幅度周期选择

波动幅度计算

年化收益率

最大年化收益率

当前年化收益率

平均年化收益率

6.2 实例：时间事件日志

https://github.com/kamidox/utils (https://github.com/kamidox/utils)

时间事件日志简介

要点：

- 使用 dida365.com 来作为 GTD 工具
- 使用特殊格式记录事件类别和花费的时间，如： “[探索发现] 体验 iMac 开发环境 [3h]”
- 导出数据
- 分析数据

数据读取

导出数据

读入数据

数据洗清

数据选择

数据解析

数据分析

时间总览

平均每天投资在自己身上的时间是多少？-> 全部时间 / 总天数

精力分配

每项工作占用的时间配比

专注力

长时间专注一件事情的能力

动态精力分配

从时间轴的维度看各项工作的精力分配

6.3 课程小结

课程回顾

1. 介绍了 ipython : 数据分析和科学计算领域的 IDE
2. Pandas 快速入门：快速了解 pandas 在数据处理方面的基础接口。以 MovieLens 数据分析结束这部分内容。
3. Pandas 基础：核心数据结构 ； 基础运算（映射，排序）； 索引（行索引，列索引，多层索引，重复索引）；
4. 分组和聚合运算
5. 数据 IO ： 从文件里导入数据及数据导出
6. 时间序列
7. 数据可视化
8. 实例：股票数据分析
9. 实例：时间事件日志

课程代码：https://github.com/kamidox/pandas_tutor (https://github.com/kamidox/pandas_tutor)

思维导图：
http://naotu.baidu.com/file/5eba96c2d922e30b7a4bf6b74c638dd0?token=440f2f0ff8c8b88f (http://naotu.baidu.com/file/5eba96c2d922e30b7a4bf6b74c638dd0?token=440f2f0ff8c8b88f)
密码：706V

课程展望

从更大的维度来看 Pandas：

- 数据收集：爬虫技术
- 数据清洗：Pandas
- 数据挖掘：Pandas
- 数据建模：Scikit 等
- 数据应用：Ternado, Django etc.

数据思维：

推荐关注知乎用户：何明科 -> 数据冰山
https://www.zhihu.com/people/he-ming-ke (https://www.zhihu.com/people/he-ming-ke)

用 excel 都可以做出逼格满满，价值连城的数据分析。在高手眼里，一片树叶都可以是武器，且其威力丝毫不比刀剑差。