Python数据分析——第6周

法律声明

【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

http://edu.dataguru.cn

关注炼数成金企业微信

- 提供全面的数据价值资讯，涵盖商业智能与数据分析、大数据、企业信息化、数字化技术等，各种高性价比课程信息，赶紧掏出您的手机关注吧！

本周内容

- 数据整理与预处理
  - 数据清洗
  - 合并数据集
  - 数据转换
  - 重塑和轴向旋转
  - 字符串操作
  - 示例

数据清洗

◆ 缺失值处理

- 删除记录
- 数据插补——拉格朗日插值法、牛顿插值法
- 不处理

插补方法	方法描述
均值/中位数/众数插补	根据属性值的类型，用该属性取值的平均数/中位数/众数进行插补
使用固定值	将缺失的属性值用一个常量替换。如广州一个工厂普通外来务工人员的“基本工资”属性的空缺值可以用2015年广州市普通外来务工人员工资标准1895元/月，该方法就是使用固定值
最近临插补	在记录中找到与缺失样本最接近的样本的该属性值插补
回归方法	对带有缺失值的变量，根据已有数据和与其有关的其他变量（因变量）的数据建立拟合模型来预测缺失的属性值
插值法	插值法是利用已知点建立合适的插值函数$f(x)$，未知值由对应点x_i求出的函数值$f(x_i)$近似代替

◆ 异常值处理

数据插补

◆ 拉格朗日插值法

根据数学知识可知，对于平面上已知的n个点（无两点在一条直线上）可以找到一个n-1次多项式$y = a_0+a_1x+a_2x^2+\ldots+a_{n-1}x^{n-1}$，使此多项式曲线过这n个点。

1）求已知的过n个点的n-1次多项式：

$$y = a_0+a_1x+a_2x^2+\ldots+a_{n-1}x^{n-1} \quad (4\text{-}1)$$

将n个点的坐标（x_1，y_1），（x_2，y_2）...（x_n，y_n）代入多项式函数，得

$$\begin{aligned} y_1 &= a_0 + a_1x_1 + a_2x_1^2 + \cdots + a_{n-1}x_1^{n-1} \\ y_2 &= a_0 + a_1x_1 + a_2x_2^2 + \cdots + a_{n-1}x_2^{n-1} \\ &\cdots\cdots \\ y_n &= a_0 + a_1x_n + a_2x_n^2 + \cdots + a_{n-1}x_n^{n-1} \end{aligned}$$

解出拉格朗日插值多项式为：

$$\begin{aligned} L(x) = {} & y_1\frac{(x-x_2)(x-x_3)\cdots(x-x_n)}{(x_1-x_2)(x_1-x_3)\cdots(x_1-x_n)} \\ & + y_2\frac{(x-x_1)(x-x_3)\cdots(x-x_n)}{(x_2-x_1)(x_2-x_3)\cdots(x_2-x_n)} \\ & + \cdots\cdots \\ & + y_n\frac{(x-x_1)(x-x_2)\cdots(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)\cdots(x_n-x_{n-1})} \end{aligned} \quad (4\text{-}2)$$

$$= \sum_{i=0}^{n} y_i \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$$

2）将缺失的函数值对应的点x代入插值多项式得到缺失值的近似值L（x）。

数据插补

◆ 牛顿插值法

1）求已知的n个点对（x_1，y_1），（x_2，y_2）…（x_n，y_n）的所有阶差商公式

$$f[x_1,x]=\frac{f[x]-f[x_1]}{x-x_1}=\frac{f(x)-f(x_1)}{x-x_1} \quad (4\text{-}3)$$

$$f[x_2,x_1,x]=\frac{f[x_1,x]-f[x_2,x_1]}{x-x_2} \quad (4\text{-}4)$$

$$f[x_3,x_2,x_1,x]=\frac{f[x_2,x_1,x]-f[x_3,x_2,x_1]}{x-x_3} \quad (4\text{-}5)$$

……

$$f[x_n,x_{n-1},\cdots,x_1,x]=\frac{f[x_{n-1},\cdots,x_1,x]-f[x_n,x_{n-1},\cdots,x_1]}{x-x_n} \quad (4\text{-}6)$$

2）联立以上差商公式建立如下插值多项式f（x）

$$\begin{aligned}f(x)=&f(x_1)+(x-x_1)f[x_2,x_1]+(x-x_1)(x-x_2)f[x_3,x_2,x_1]+\\&(x-x_1)(x-x_2)(x-x_3)f[x_4,x_3,x_2,x_1]+\cdots+\\&(x-x_1)(x-x_2)\cdots(x-x_{n-1})f[x_n,x_{n-1},\cdots x_2,x_1]+\\&(x-x_1)(x-x_2)\cdots(x-x_n)f[x_n,x_{n-1},\cdots x_1,x]\\=&P(x)+R(x)\end{aligned} \quad (4\text{-}7)$$

$$\begin{aligned}P(x)=&f(x_1)+(x-x_1)f[x_2,x_1]+(x-x_1)(x-x_2)f[x_3,x_2,x_1]+\\&(x-x_1)(x-x_2)(x-x_3)f[x_4,x_3,x_2,x_1]+\cdots+\\&(x-x_1)(x-x_2)\cdots(x-x_{n-1})f[x_n,x_{n-1},\cdots,x_2,x_1]\end{aligned} \quad (4\text{-}8)$$

$$R(x)=(x-x_1)(x-x_2)\cdots(x-x_n)f[x_n,x_{n-1},\cdots x_1,x] \quad (4\text{-}9)$$

P（x）是牛顿插值逼近函数，R（x）是误差函数。

3）将缺失的函数值对应的点x代入插值多项式得到缺失值的近似值f（x）。

合并数据

- Pandas对象
  - Merge方法：根据一个或多个键将不同dataframe中的行合并
  - Concat方法：沿一条轴将对多个对象堆叠起来

- 数据库风格的DataFrame合并
  - Merge
  - Merge参数

参数	说明
left	参与合并的左侧DataFrame
right	参与合并的右侧DataFrame
how	“inner”、“outer”、“left”、“right”其中之一。默认为“inner”
on	用于连接的列名。必须存在于左右两个DataFrame对象中。如果未指定，且其他连接键也未指定，则以left和right列名的交集作为连接键
left_on	左侧DataFrame中用作连接键的列
right_on	右侧DataFrame中用作连接键的列
left_index	将左侧的行索引用作其连接键
right_index	类似于left_index
sort	根据连接键对合并后的数据进行排序，默认为True。有时在处理大数据集时，禁用该选项可获得更好的性能
suffixes	字符串值元组，用于追加到重叠列名的末尾，默认为('_x', '_y')。例如，如果左右两个DataFrame对象都有“data”，则结果中就会出现“data_x”和“data_y”
copy	设置为False，可以在某些特殊情况下避免将数据复制到结果数据结构中。默认总是复制

合并数据

- 索引上的合并

- 轴向连接
  - Numpy数组——concatenation
  - Pandas对象——concat
  - Concat的参数

参数	说明
objs	参与连接的pandas对象的列表或字典。唯一必需的参数
axis	指明连接的轴向，默认为0
join	“inner”、“outer”其中之一，默认为“outer”。指明其他轴向上的索引是按交集（inner）还是并集（outer）进行合并
join_axes	指明用于其他n-1条轴的索引，不执行并集/交集运算
keys	与连接对象有关的值，用于形成连接轴向上的层次化索引。可以是任意值的列表或数组、元组数组、数组列表（如果将levels设置成多级数组的话）
levels	指定用作层次化索引各级别上的索引，如果设置了keys的话[译注3]
names	用于创建分层级别的名称，如果设置了keys和（或）levels的话
verify_integrity	检查结果对象新轴上的重复情况，如果发现则引发异常。默认（False）允许重复
ignore_index	不保留连接轴上的索引，产生一组新索引range(total_length)

合并数据

- 合并重叠数据
  - Numpy——where
  - Series——combine_first
  - DataFrame_combin_first

重塑和轴向旋转

- 重塑层次化索引
  - Stack：将数据的列“旋转”为行
  - Unstack：将数据的行“旋转”为列

- 长格式与宽格式数据的转换
  - 长格式数据
  - 宽格式数据

数据转换

- 移除重复数据
- 利用函数或映射进行数据转换
- 替换值
- 重命名轴索引
- 离散化和面元划分
- 检测和过滤异常值
- 排列和随机采样
- 计算指标与哑变量

字符串操作

- 字符串对象方法
  - Split
  - Strip
  - ……
- Python内置的字符串方法

方法	说明
count	返回子串在字符串中的出现次数（非重叠）
endswith、startswith	如果字符串以某个后缀结尾（以某个前缀开头），则返回True
join	将字符串用作连接其他字符串序列的分隔符
index	如果在字符串中找到子串，则返回子串第一个字符所在的位置。如果没有找到，则引发ValueError。
find	如果在字符串中找到子串，则返回第一个发现的子串的第一个字符所在的位置。如果没有找到，则返回-1
rfind	如果在字符串中找到子串，则返回最后一个发现的子串的第一个字符所在的位置。如果没有找到，则返回-1
replace	用另一个字符串替换指定子串
strip、rstrip、lstrip	去除空白符（包括换行符）。相当于对各个元素执行x.strip()（以及rstrip、lstrip）。译注10
split	通过指定的分隔符将字符串拆分为一组子串
lower、upper	分别将字母字符转换为小写或大写
ljust、rjust	用空格（或其他字符）填充字符串的空白侧以返回符合最低宽度的字符串

字符串操作

正则表达式

- Re模块
  - 模式匹配
  - 替换
  - 拆分

正则表达式方法

方法	说明
findall、finditer	返回字符串中所有的非重叠匹配模式。findall返回的是由所有模式组成的列表，而finditer则通过一个迭代器逐个返回
match	从字符串起始位置匹配模式，还可以对模式各部分进行分组。如果匹配到模式，则返回一个匹配项对象，否则返回None
search	扫描整个字符串以匹配模式。如果找到则返回一个匹配项对象。跟match不同，其匹配项可以位于字符串的任意位置，而不仅仅是起始处
split	根据找到的模式将字符串拆分为数段
sub、subn	将字符串中所有的（sub）或前n个（subn）模式替换为指定表达式[译注12]。在替换字符串中可以通过\1、\2等符号表示各分组项

字符串操作

- Pandas中矢量化字符串方法

方法	说明
cat	实现元素级的字符串连接操作，可指定分隔符
contains	返回表示各字符串是否含有指定模式的布尔型数组
count	模式的出现次数
endswith、startswith	相当于对各个元素执行x.endswith(pattern)或x.startswith(pattern)
findall	计算各字符串的模式列表
get	获取各元素的第i个字符
join	根据指定的分隔符将Series中各元素的字符串连接起来
len	计算各字符串的长度
lower、upper	转换大小写。相当于对各个元素执行x.lower()或x.upper()
match	根据指定的正则表达式对各个元素执行re.match
pad	在字符串的左边、右边或左右两边添加空白符
center	相当于pad(side='both')
repeat	重复值。例如，s.str.repeat(3)相当于对各个字符串执行x * 3
replace	用指定字符串替换找到的模式
slice	对Series中的各个字符串进行子串截取
split	根据分隔符或正则表达式对字符串进行拆分
strip、rstrip、lstrip	去除空白符，包括换行符。相当于对各个元素执行x.strip()、x.rstrip()、x.lstrip()

示例：USDA食品数据库

- 美国农业部（USDA）的一份关于食物营养信息的数据库
- 该数据的JSON版

```
{
  "id": 21441,
  "description": "KENTUCKY FRIED CHICKEN, Fried Chicken, EXTRA CRISPY,
Wing, meat and skin with breading",
  "tags": ["KFC"],
  "manufacturer": "Kentucky Fried Chicken",
  "group": "Fast Foods",
  "portions": [
    {
      "amount": 1,
      "unit": "wing, with skin",
      "grams": 68.0
    },

    ...
  ],
  "nutrients": [
    {
      "value": 20.8,
      "units": "g",
      "description": "Protein",
      "group": "Composition"
    },

    ...
  ]
}
```

炼数成金逆向收费式网络课程

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。

- 关于逆向收费式网络的详情，请看我们的培训网站 http://edu.dataguru.cn

Thanks

FAQ时间