科学计算

王肖燕

主要内容

一．Numpy简介

二．Pandas简介

三．MatPlotlib介绍

四．实例讲解

Numpy

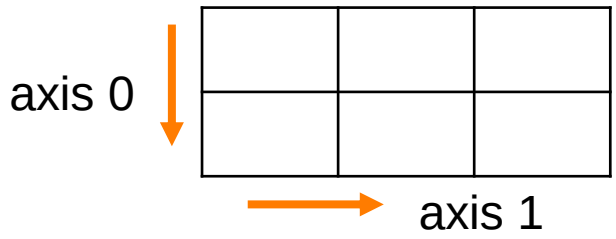
- 是Python数组计算、矩阵运算和科学计算的核心库
- 提供了一个高性能的数组对象，轻松创建n维数组，丰富的函数和方法让数组运算实现方便
- 应用于多领域并可与C++/Fortran语言无缝结合

Numpy简介

- Numpy的数据结构是n维的数组对象
  - ndarray
  - 列表，效率降低
- Numpy属于Python的第三方工具，使用前在代码中导入
  - import numpy as np
  - data1=[1,2,3,4,5]
  - print(np.array(data1))
  - Print(np.eye(3))

相关概念

- 一维数组
- 二维数组
  - 包括行和列，以称矩阵
- 三维数组
  - 矩阵列表，常见的多维数组
- 轴
  - axis: 指定某个轴，则沿着这个轴做相关操作



创建数组1

- 列表转换成数组
  - data2=[[1,2,3],[4,5,6]]
  - array2=np.array(data2)
- 数据类型
  - array2.dtype
- 转换数据格式
  - array2.astype(str)

创建数组2

- 复制
  - Arr1=np.array([4,5,6])
  - Arr2=np.array(Arr1,copy=True)
  - Arr2是Arr1的副本
- 维度
  - L1=[6,7,8,9]
  - Arr3=np.array(L1,ndmin=2)
  - Arr3中的数组是[[6,7,8,9]]

创建数组3

- 以0填充的数组
  - Arr=np.zeros(3)
- 以1填充的数组
  - Arr=np.ones(3)
- 以指定值填充的数组
  - Arr=np.full((4,4),666)

创建数组4

- Arange函数
  - Arr=np.arange(1,20,2)
- 等差
  - Arr=np.linspace(35,43,9)
- 等比
  - Arr=np.logspace(0,9,10,base=2)

创建数组5

- rand函数
  - Arr=np.random.rand(6)
  - Arr=np.random.rand(2,3)
- randn函数，满足正态分布
  - Arr=np.random.randn(6)
  - Arr=np.random.randn(2,5)
- randint函数
  - Arr=np.random.randint(1,5,10)

创建数组6

- zeros_like函数
  - Arr=np.zeros_like([[1,2,3][4,5,6]])
- ones_like函数
  - Arr=np.ones_like([[1,2,3][4,5,6]])
- full_like函数
  - Arr=np.full_like([[1,2,3],[4,5,6]],1.6,dtype=float)

数组运算1

- 计算方便，不需要大量的循环即可批量计算
- 算术运算
  - array1+1，及其它算术运算符
  - array1*array2，及其它算术运算符
- 比较运算
  - Arr1=np.array([3,4])
  - Arr2=np.array([5,3])
  - print(Arr1>Arr2)，及其它关系运算

数组运算2

- Array数组内的元素访问（索引）和列表相同
- 访问
  - 一维：arr[0], arr[-1]
  - 二维：arr[0,0], arr[0,0]
- 切片
  - array1[2], array1[-2:], array1[-2:]
  - arr=np.array([[1,2,3],[4,5,6],[7,8,9]])
  - arr[:2,1:]与arr[:2][1:]

数组运算3

- 数组改变形状
  - 一维重塑
    - Arr1=np.arange(8)
    - Arr2=Arr1.reshape(2,4)
  - 二维重塑
    - Arr1=np.array([[1,2,3],[4,5,6]])
    - Arr2=Arr1.reshape(3,2)
- 转置
  - Arr.T 或 Arr.transpose()

数组操作

- 增加
  - 水平：np.hstack((Arr1, Arr2))
  - 垂直：np.vstack((Arr1, Arr2))
- 删除
  - 行：np.delete(Arr,1,axis=0), np.delete(Arr,(1,2),0)
  - 列：np.delete(Arr,1,axis=1)
- 查询
  - Arr=np.arange(10)
  - np.where(Arr>4,1,0)
  - np.where(Arr>4)

数组函数

- abs()
- around()
- ceil()
- floor()
- modf()小数整数分隔
- maximum(), fmax()
- minimum(), fmin()
- sum()
- cumsum()累计求和
- cumprod()累积
- np.mean()
- arr.min(),max()
- argmin(), argmax()
- average()
- ptp()最大值最小值差

举例

- 定价问题：某商品的销售价格表为[34.5, 36, 37.8, 39, 39.8, 33.6]，对应的销售价格下，销售数量分别为[900, 580, 230, 150, 120, 1800]，求定价在多少比较合理？
- 把王维的这首诗“远看山有色近听水无声春去花还在人来鸟不惊”存放在数组中，并转换为四行五列的矩阵。

Pandas

- 是基于NumPy 的一种工具，该工具是为了解决数据分析任务而创建的
- 纳入了大量库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具
- 提供了大量能使我们快速便捷地处理数据的函数和方法
- pandas是使Python成为强大而高效的数据分析环境的重要因素之一

Pandas提供的数据结构

- **Series**：带标签的一维数组，与Numpy中的一维array类似，二者与Python基本的数据结构List也很相近。能保存不同种数据类型：字符串、boolean值、数字等
- **DataFrame**：带标签且大小可变的二维表格结构
- **Panel** ：带标签且大小可变的三维数组

Series

- 序列，用于存储一行或一列的数据，以及与之相关的索引集合
- Series([数据1，数据2，…],index=[1,2,3])
- 序列允许存放多种数据类型，索引也可以省略
- 可通过位置或索引访问数据

生成序列

```
>>> import numpy as np
>>> import pandas as pd
>>> x = pd.Series([1,3,5,np.nan])
>>> x
0    1.0
1    3.0
2    5.0
3    NaN
dtype: float64
```

DataFrame

- 存储多行多列的数据集合
- Series的容器

```
df=pd.DataFrame({'age':pd.Series([18,19,22]),
                 'name':pd.Series(['bob','alice','anna'])})
```

```
In [4]: df
Out[4]:
   age   name
0   18    bob
1   19  alice
2   22   anna
```

数据导入

- 文本文件
  - 导入txt文件：read_table(file, names=[列名1,列名2,...],sep="",...)
- 电子表格
  - 导入Excel文件：read_excel(file, sheetname,header=0)

导入文本

```
import numpy as np
import pandas as pd
df1=pd.read_table(r"rz.txt", encoding="utf-8")
```

```
In [20]: df1
Out[20]:
          学号        班级   姓名 性别  英语  体育  军训  数分  高代  解几
0  2308024241  23080242   成龙  男  76  78  77  40  23  60
1  2308024244  23080242   周怡  女  66  91  75  47  47  44
2  2308024251  23080242   张波  男  85  81  75  45  45  60
3  2308024249  23080242   朱浩  男  65  50  80  72  62  71
4  2308024219  23080242   封印  女  73  88  92  61  47  46
5  2308024201  23080242   迟培  男  60  50  89  71  76  71
6  2308024347  23080243   李华  女  67  61  84  61  65  78
7  2308024307  23080243   陈田  男  76  79  86  69  40  69
```

```
In [21]: df1.head(2)
Out[21]:
          学号        班级  姓名 性别  英语  体育  军训  数分  高代  解几
0  2308024241  23080242  成龙  男  76  78  77  40  23  60
1  2308024244  23080242  周怡  女  66  91  75  47  47  44
```

```
In [22]: df1.tail(2)
Out[22]:
          学号        班级   姓名 性别  英语  体育  军训  数分  高代  解几
19  2308024422  23080244  李晓亮  男  85  60  85  72  72  83
20  2308024201  23080242   迟培  男  60  50  89  71  76  71
```

导入Excel

```
import numpy as np
import pandas as pd
df2=pd.read_excel(r"data.xls",'Sheet3')
```

```
In [34]: df2
Out[34]:
```

	学号	班级	姓名	性别	英语	体育	军训	数分	高代	解几
0	2308024241	23080242	成龙	男	76	78	77	40	23	60
1	2308024244	23080242	周怡	女	66	91	75	47	47	44
2	2308024251	23080242	张波	男	85	81	75	45	45	60
3	2308024249	23080242	朱浩	男	65	50	80	72	62	71
4	2308024219	23080242	封印	女	73	88	92	61	47	46
5	2308024201	23080242	迟培	男	60	50	89	71	76	71
6	2308024347	23080243	李华	女	67	61	84	61	65	78

导出Excel

```
import numpy as np
import pandas as pd
df=pd.DataFrame({'age':pd.Series([18,19,22]),
                 'name':pd.Series(['bob','alice','anna'])})
df.to_excel(r"r1.xlsx")
df.to_excel(r"r2.xlsx",index=False)
```

	A	B	C
1		age	name
2	0	18	bob
3	1	19	alice
4	2	22	anna

	A	B
1	age	name
2	18	bob
3	19	alice
4	22	anna

数据清洗

- 重复值
  - 数据中存在重复数据
  - 一般做删除处理
- 缺失值
  - 识别
  - 处理

重复值

- df.duplicated()
  - 判断每一行数据是否重复，True为重复
- df.drop_duplicates()
  - 去除全部的重复数据
- df.drop_duplicates([列名])
  - 去除指定列的重复数据

缺失值

- 查看
  - 缺失值一般以NaN表示
  - df.info()查看
  - df.isnull()判断是否存在缺失值
- 处理
  - 删除
    - df.dropna()删除所有包含缺失值的数据行
    - df.[df[列名].notnull()]删除某一行中数据存在空的行
  - 填充
    - df.fillna(0)，可以只填充某一列

索引设置

- 重置
  - df.index=[‘aa’,’bb’,’cc’]
  - df.reindex([‘a’,’b’,’c’,’d’])
- 设置某列为索引
  - df.set_index([‘c4’])
- 设置为从0开始的连续索引
  - df.reset_index()
  - df.reset_index(drop=True)

数据查看1

```
>>> df.head()   #默认显示前5行
>>> df.head(3)    #查看前3行
>>> df.tail(2)      #查看最后2行
>>> df.index
>>> df.columns
>>> df.values
>>> df.describe()  #平均值、标准差、最小值、最大值 等信息
```

数据查看2

- 一行
  - df.loc[索引名]
  - df.iloc[索引值]
- 多行
  - df.loc[索引名列表]， df.loc[索引名切片]
  - df.iloc[索引值列表]，df.iloc[索引值切片]
- 指定列
  - df[列名列表]
  - df.loc[:,[列名列表]]， df.loc[:,列名切片]
  - df.iloc[:,[列索引列表]]，df.iloc[:,列索引切片]

数据查看3

- 某个元素
  - df.iat[行索引值，列索引值]
  - df.at[行索引名，列索引名]
  - df.iloc[行索引值，列索引值]
  - df.loc[行索引名，列索引名]
- 按条件取
  - df.loc[条件]
  - df[条件]

增加数据

- 列
  - df[列名]=数据
  - df.loc[:,列名]=数据
  - df.insert(插入位置，列名，数据）
- 行
  - df.loc[行索引]=数据
  - df1.append(df2)

修改数据

- 列名
  - df.columns=列名列表
- 行名
  - df.index=行名列表
- 数据
  - 引用到行，修改整行
  - 引用到列，修改整列
  - 引用数据，修改数据

删除数据

- 行
  - df.drop(行名列表)，df.drop(index=行名列表)
- 列
  - df.drop(列名列表，axis=1)，df.drop(columns=行名列表)
- 按条件
  - df.drop(index=满足条件的索引)

排序

- 依据标签进行

```
>>> df.sort_index(axis=0, ascending=False)
>>> df.sort_index(axis=0, ascending=True)
>>> df.sort_index(axis=1, ascending=False)
```

- 依据值进行排序

```
>>> df.sort_values(by='A') #依照某列的数据排序
```

随机排列数据

s=np.random.permutation(len(df.index))

df.take(s) #按行随机排序二维数组中的数据

```
In [38]: s
Out[38]: array([1, 2, 0])

In [39]: df2.take(s)
Out[39]:
          学号        班级  姓名 性别  英语  体育  军训  数分  高代  解几
1  2308024244  23080242  周怡  女  66  91  75  47  47  44
2  2308024251  23080242  张波  男  85  81  75  45  45  60
0  2308024241  23080242  成龙  男  76  78  77  40  23  60
```

数据选择1

>>> df[‘高代’]　#选择列

```
In [42]: df2['高代']
Out[42]:
0     23
1     47
2     45
3     62
4     47
5     76
6     65
```

>>> “李小强” in df[‘姓名’]

False

>>> “成龙” in df[‘姓名’].values

True

数据选择2

- loc
  - 以columns和index为参数
  - 只有一个参数时，指index
  - df.loc[‘a’]
- iloc
  - 以columns和index位置索引为参数
  - 只有一个参数时，指index索引
  - df.iloc[0]

数据选择3

- 一行
  - df.loc[‘姓名’]
  - df.iloc[0]
- 多行
  - df.loc[[‘姓名’,’Python’]]
  - df.iloc[[0,2]]
  - df.loc[‘姓名’:’Python’]
  - df.iloc[0:2]

数据分析应用

- 利用前面的Numpy和Pandas等常用分析工具并结合常用的统计量
  - 进行数据描述
  - 展现数据特征
  - 内在结构

基本统计分析

- 描述性统计分析
  - 一般统计某个变量的最小值、第一个四分位值、中值、第三个四分位值以及最大值
- describe函数
  - Pandas中
  - 返回均值、标准差、最大值、最小值、分位数等

分组分析

- 根据分组字段将分析对象划分成不同的部分，以对比分析各组之间差异性
- 常用的统计指标
  - 计数、求和和平均值

分布分析

- 根据分析的目的，将定量数据进行等距或不等距的分组，研究各组分布规律的一种分析方法

交叉分析

- 分析两个或两个以上分组变量之间的关系
- 以交叉表形式进行变量间关系的对比分析
- 常见格式：

  df.pivot_table(values,index[,columns,aggfunc])

相关分析

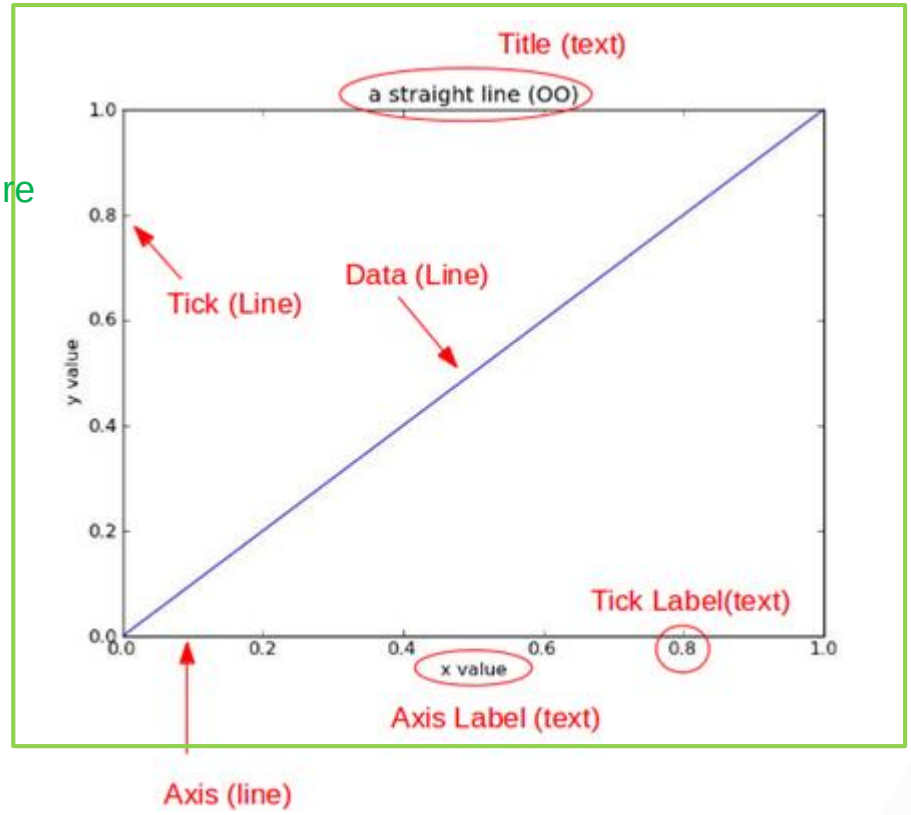
- 判断两个变量是否具有线性相关关系最直观的方法是绘制散点图
  - 看变量之间是否符合某个变化规律
- 计算相关系数
  - 两个连续变量之间的关系：Pearson相关系数
  - 不服从正态分布的变量：Spearman秩相关系数
- 相关系数与相关程度
  - 低度相关：0~0.3
  - 中度相关：0.3~0.8
  - 高度相关：0.8~1

matplotlib库

- 依赖于numpy模块和tkinter模块
- 可以绘制多种形式的图形，包括线图、直方图、饼状图、散点图、误差线图等等，图形质量可满足出版要求，是数据可视化的重要工具

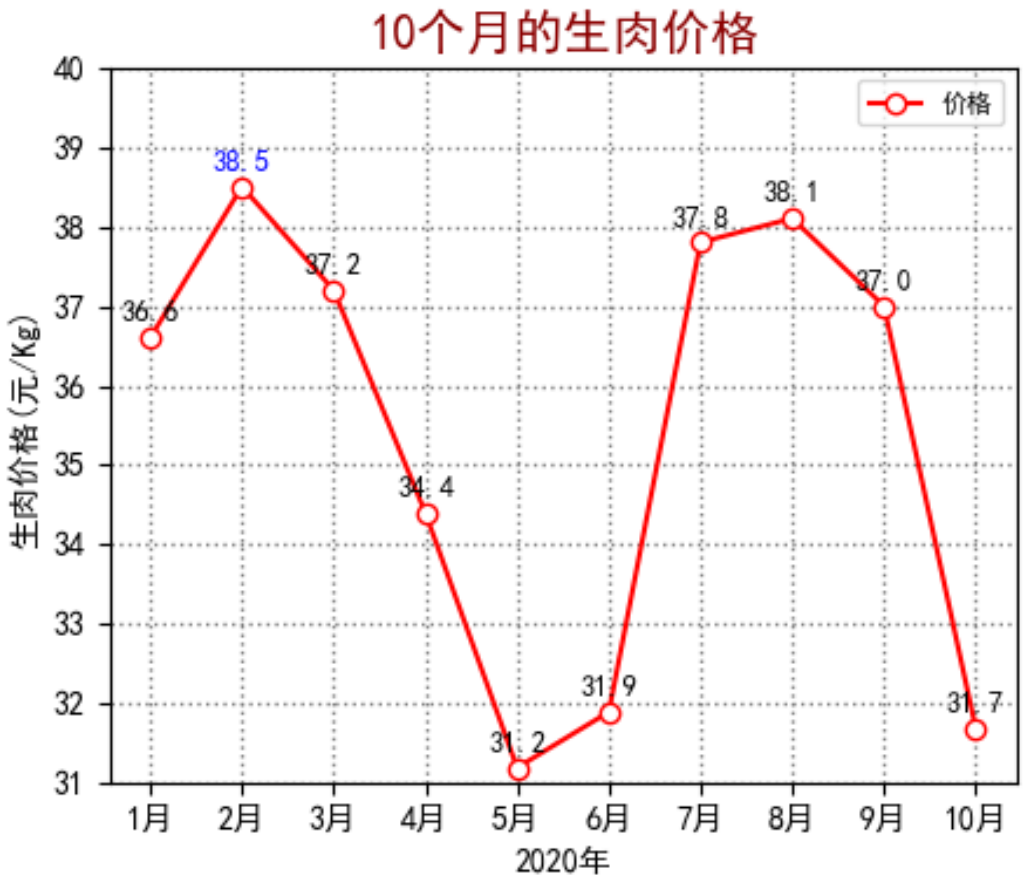
matplotlib库

figure



画布、绘图区、数据、图表标题、坐标轴及标题、刻度、文本标签等

拆线图



- color='r'
- linestyle='-'
- marker='o'
- mfc='w'
- figsize=(5,4)
- xticks(locs,[labels])
- ylim(31,39)
- grid(color,linestyle,linewidth)
- plt.text(x,y,s,fontsize,color)
- Plt.legend(('价格'),fontsize,loc)

Marker & linestyle

'.'	point marker
','	pixel marker
'o'	circle marker
'v'	triangle_down marker
'^'	triangle_up marker
'<'	triangle_left marker
'>'	triangle_right marker
'1'	tri_down marker
'2'	tri_up marker
'3'	tri_left marker
'4'	tri_right marker

Marker

's'	square marker
'p'	pentagon marker
'*'	star marker
'h'	hexagon1 marker
'H'	hexagon2 marker
'+'	plus marker
'x'	x marker
'D'	diamond marker
'd'	thin_diamond marker
'|'	vline marker
'_'	hline marker

linestyle

'-'	solid line style
'--'	dashed line style
'-.'	dash-dot line style
':'	dotted line style

color

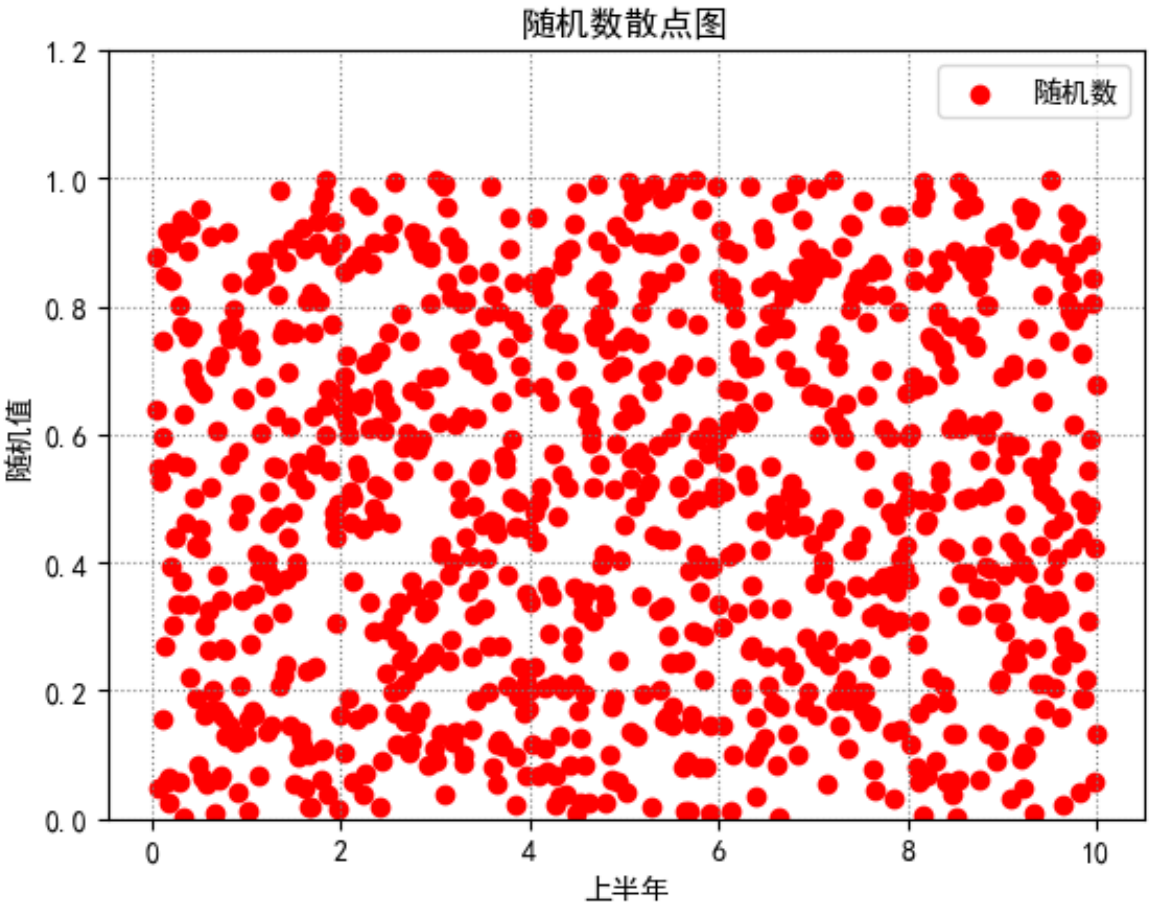
black
k
dimgray
dimgrey
grey
gray
darkgrey
darkgray
silver
lightgray
lightgrey
gainsboro
whitesmoke
white
w
snow
rosybrown
lightcoral
indianred
brown
firebrick
maroon
darkred
red
r
mistyrose
salmon
tomato
darksalmon
coral
orangered
lightsalmon
sienna
seashell
chocolate
saddlebrown
sandybrown
peachpuff
peru
linen

bisque
darkorange
burlywood
antiquewhite
tan
navajowhite
blanchedalmond
papayawhip
moccasin
orange
wheat
oldlace
floralwhite
darkgoldenrod
goldenrod
cornsilk
gold
lemonchiffon
khaki
palegoldenrod
darkkhaki
ivory
beige
lightyellow
lightgoldenrodyellow
olive
y
yellow
olivedrab
yellowgreen
darkolivegreen
greenyellow
chartreuse
lawngreen
sage
lightsage
darksage
honeydew
darkseagreen
palegreen

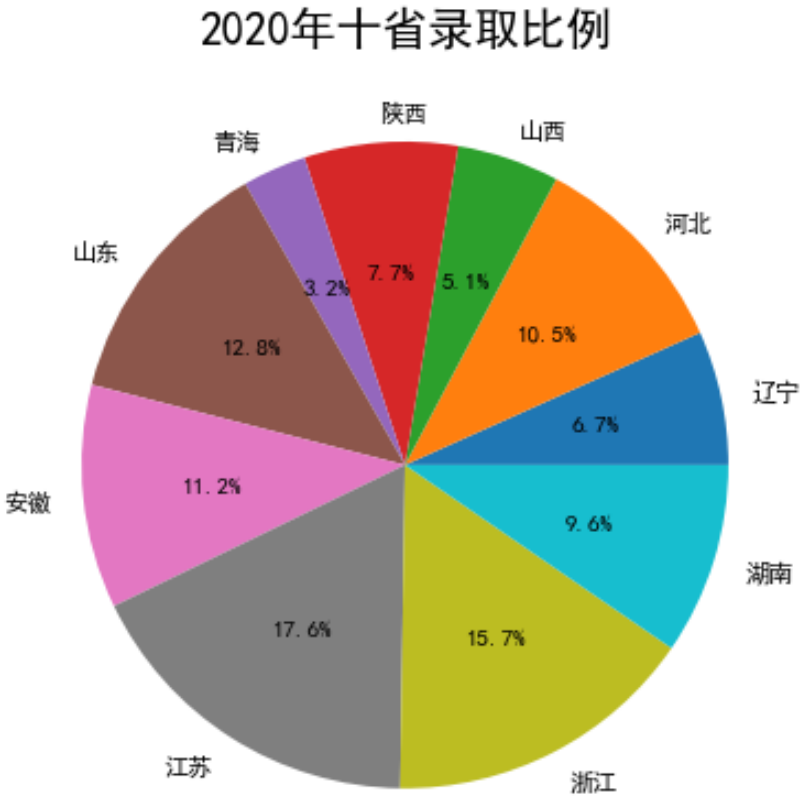
lightgreen
forestgreen
limegreen
darkgreen
green
g
lime
seagreen
mediumseagreen
springgreen
mintcream
mediumspringgreen
mediumaquamarine
aquamarine
turquoise
lightseagreen
mediumturquoise
azure
lightcyan
paleturquoise
darkslategray
darkslategrey
teal
darkcyan
c
cyan
aqua
darkturquoise
cadetblue
powderblue
lightblue
deepskyblue
skyblue
lightskyblue
steelblue
aliceblue
dodgerblue
lightslategrey
lightslategray
slategray

slategrey
lightsteelblue
cornflowerblue
royalblue
ghostwhite
lavender
midnightblue
navy
darkblue
mediumblue
blue
b
slateblue
darkslateblue
mediumslateblue
mediumpurple
blueviolet
indigo
darkorchid
darkviolet
mediumorchid
thistle
plum
violet
purple
darkmagenta
m
fuchsia
magenta
orchid
mediumvioletred
deeppink
hotpink
lavenderblush
palevioletred
crimson
pink
lightpink

散点图



饼状图



- plt.pie()
- sizes
- labels
- autopct='%.1f%%'
- textprops
- labeldistance

柱状图

Thank You!