numpy matplotlib pandas复习

import numpy as np

import pandas as pd

from matplotlib import pyplot as plt

没有强制要求让写模块的引入.就不写.

试题类型:

1.选择题 15\*2 ----30 (知识点各占5个)

2.多选题 5\*2 -----10 (np 2 plt 1 pd 2)

3.判断题 5\*2 -----10 (np 2 plt 1 pd 2)

4.简答题 5\*6 -----30 (np 2 plt 1 pd 2)

5.分析题 2\*10 -----20 (pd)

考试时间(雨课堂):

--- 2周

复习:

numpy:

1.创建数组 np.array()

2.区间数组 np.arange(10)

---start:0 end:10(不包含10)

--- np.arange(1,10,3):

--步长:

1 2 3 4 5 6 7 8 9

3.等差,等比数列

--- 包含start 和end

--- 具体生成多少样本数由num

--默认是50

4.全0 全1

--- zeros ones

--np.zeros((2,3))

-- np.zeros(2,3)----X

创建一个10\*10数组,操作之后数据的边框全为1,内部全为0

5.数组的属性:

np.ndim ---- 几维数组

np.shape ---- (2,3) (4,2,3)

np.size ---- 总个数 len():一维是相同,之上结果不同

np.itemsize -- 32/8 == int(4) 64/8 float(8)

6.索引和切片

[start:end:step]

[::-1]

ar1 = np.arange(10)

ar1[:5] ar1[5:]

②.二维

ar1 = np.arange(20).reshape(4,5)

ar1[1][2]

arr[行,列]

arr[:,1] arr[...,1]

arr[...,1:]

y = x[[0,1,2],[0,1,0]]

# y分别获取x中的(0,0)、(1,1) 和(2,0)的数据

③. [条件]

7.统计函数

mean--平均值

median--中位数

std -- 标准查找

var -- 方差

sum()

max

min

8.随机数:

np.random.rand --- >(0,1)均匀分布

np.random.randn --->(0,1)均匀分布

np.random.randint---(60,101,size=(10,10))

9,函数:

resize和reshape的区别:

reshape:将源数据转化新的维度

转化为数据的size == 源数据的size

resize: 转化为数据的size 等于或者不等于 源数据的size

numpy.unique()--->唯一

sort() 排序

二.绘图:

1.基本属性

初始化设置:

中文字题:plt.rcParams["font.sans-serif"]

轴上面的负号:plt.rcParams['axes.unicode\_minus'] = False

设置大小

设置分辨率

linestyle-->线条的样式

linewidth ->线条的宽度

2.函数:

xlabel() ylabel() --->x轴和y轴名称

xticks() yticks--->设置x轴和y轴的刻度

xlim() ylim() -- >设置x轴和y轴的取值范围

show() --> 显示图表.需要注意,如果需要保存,必须先保存,在使用show显示

legend()--->图例

---需要指定图例位置:

loc=" " 有5个单词组合确定位置

subplot()和 subplots()区别:

3.绘制不同的图形:

plt.plot()---折线图

plt.bar()---柱状图

plt.barh()--- 横向的柱状图

plt.pie() -- 绘制饼状图

plt.scatter()---绘制散点图

plt.hist()---绘制直方图

柱状图和直方图的区别?

给2组数据,绘制柱状图?

三.pandas

1. Series和DataFrame:

Series -- 一维的

DataFrame -- 二维

和numpy--->标签

Series:

--- 创建:pd.Series

-- 切片和索引:

和numpy操作基本一致

如果是存在标签:

②.方法和属性

index

values

unique()--->去重数据

nunique()-->返回唯一值的个数

DataFrame:

①.创建:

np.DataFrame(列表)

-- 列表中每一个元素是一行数据

--- 列和行标签默认(RangeIndex)

np.DataFrame(字典)

-- k:v格式(k对应列标签,v对应是列值)

-- 行标签:(RangeIndex)

②.行列操作:

列:

df["列名"]

df["新列名"] == 一组数据

df["列名"] --->Series

df[["列名"]] -- >DataFrame

df[["列名1","列名2"]]

行:

df.loc[]

df.iloc[]

如果想使用列索引,必须借助iloc

df.iloc[:,2:]

③.属性和方法:

T --->转置(行-->列,列->行)

columns---> 列标签

head()-----> 查看前几条数据,默认是5

tail() ---> 查看后几条数据,默认是5

info()--->显示数据概括信息

sort\_index()--->按照行标签排序

sort\_values()--->按照列值排序

时间:

①.返回固定频率的时间戳

date\_range()---方法

: start --包含开始

: end --包含结束

: periods --->要生成的周期数

: freq --->默认是天

pd.date\_range(start='1/1/2021', end='1/08/2021'):

---->['2021-01-01', '2021-01-02', '2021-01-03', '2021-01-04', '2021-01-05', '2021-01-06', '2021-01-07', '2021-01- 08']

pd.date\_range(start='2010', end='2011')

-->返回的是2010全年的和2011-01-01

pd.date\_range(start='1/1/2018', periods=8)

pd.date\_range(end='1/8/2018', periods=8)

分组.聚合:

groupby().min()

groupby("company").agg(["min","max"])

# 将company作为行标签 . 所有其他数值型列的最大值和最小值

缺失值:

fillna(值)

要求:

将"age"字典填充为"age"字段的平均值/中位数

age\_mean = df['age'].mean()

df['age'].fillna(age\_mean)

文件读取:

read\_csv

read\_excel

合并操作:

merge(左数据,右数据,on=关联字段,how="left/right/inner")

user: id username user\_score: id user\_id score

关联字段:user(id) user\_score(user\_id)

a.将user数据中id列标签改成user\_id

merge(user,user\_score,on="user\_id")

pandas函数:

ser\_index

reset\_index()

....

字符串操作:

lower() 将的字符串转换为小写。

len() 得出字符串的长度。

strip() 去除字符串两边的空格（包含换行符,看起来像空格\t）。

split() 用指定的分割符分割字符串。

contains(pattern) 如果子字符串包含在元素中，则为每个元素返回一个

布尔值 True，否则为 False

startswith(pattern) 如果 Series 中的元素以指定的字符串开头，则返回

True。

绘图:

df.plot.bar()

实例:

复习下将的pandas操作实例和实验作业

编程题：

绘制柱状图

Pandas操作题

分析题：numpy索引和切片操作，随机数，操作函数和统计函数

泰坦尼克灾难数据分析

淘宝电商数据分析