# 1、去掉数据表中的列以及增加列

for i in range(1,8):

**data\_2g['Order\_'+ str(i)] = data\_2g.apply(lambda x: (x['RNCID\_'+ str(i)], x['CellID\_'+ str(i)]), axis=1)**

data\_2g.drop('RNCID\_'+ str(i), axis=1, inplace=True)

data\_2g.drop('CellID\_'+ str(i), axis=1, inplace=True)

# data\_2g.to\_csv('E:/test.csv')

# 按照主基站分组

data\_group = data\_2g.groupby(['Order\_1', 'Lon\_1', 'Lat\_1']).groups

# 存储分组后的数据，key为（'RNCID\_1', 'CellID\_1'）

data\_groupByBase = {}

for id in data\_group:

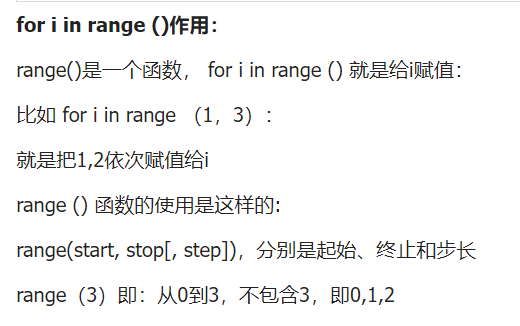
data\_list = data\_2g[(data\_2g['Order\_1'][0] == id[0][0]) & (data\_2g['Order\_1'][1] == id[0][1])].reset\_index(drop=True)

data\_groupByBase[id] = data\_list

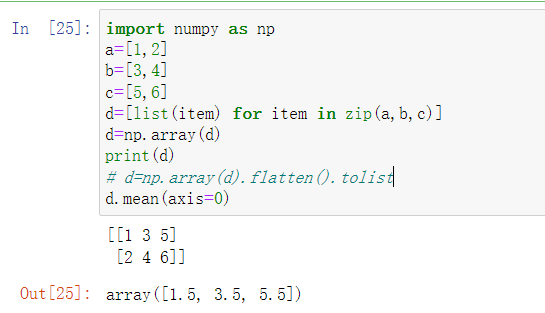
print(len(data\_groupByBase))

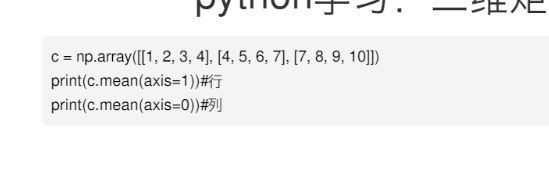
# 2、range

左闭右开



# 3、求平均值

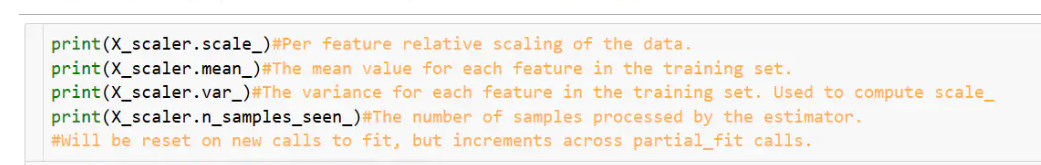




# 4、判断类型



# 5、kmeans



import pandas as pd

from sklearn.cluster import KMeans

import matplotlib.pyplot as plt

SSE=[]

#存放每次结果的误差平方和

for k in range(1, 9):

estimator=KMeans(n\_clusters=k)

#构造聚类器

estimator.fit(xxx)

SSE.append(estimator.inertia\_)

X=range(1, 9)

plt.xlabel(‘k’)

plt.ylabel(‘SSE’)

plt.plot(X, SSE, ‘o-‘)

plt.show()

# 6、数据排序

sort\_values

by：按哪几列排序

ascending=True 表示升序排列，ascending=True表示降序排列

na\_position=True表示排序的时候，把空值放到前列，这样可以比较清晰的看到哪些地方有空值

