**Python数据处理编程**

**实验报告**

****

|  |  |
| --- | --- |
| **姓名** |  |
| **学号** |  |
| **班级** |  |
| **专业** |  |
| **年级** |  |

目录

[一、 问题描述](#_Toc16459_WPSOffice_Level1) [1](#_Toc16459_WPSOffice_Level1)

[1.1 实验目的](#_Toc16459_WPSOffice_Level2) [1](#_Toc16459_WPSOffice_Level2)

[1.2 实验开发环境和工具](#_Toc18983_WPSOffice_Level2) [1](#_Toc18983_WPSOffice_Level2)

[1.3 实验内容](#_Toc26830_WPSOffice_Level2) [1](#_Toc26830_WPSOffice_Level2)

[二、 需求和规格说明](#_Toc18983_WPSOffice_Level1) [1](#_Toc18983_WPSOffice_Level1)

[三、 概要设计](#_Toc26830_WPSOffice_Level1) [5](#_Toc26830_WPSOffice_Level1)

[四、详细设计](#_Toc23828_WPSOffice_Level1) [5](#_Toc23828_WPSOffice_Level1)

[4.1设计思想](#_Toc18863_WPSOffice_Level2) [5](#_Toc18863_WPSOffice_Level2)

[五、结果分析：](#_Toc16578_WPSOffice_Level1) [12](#_Toc16578_WPSOffice_Level1)

[5.1实现注释程序清单](#_Toc29867_WPSOffice_Level2) [12](#_Toc29867_WPSOffice_Level2)

[5.2结果分析](#_Toc3801_WPSOffice_Level2) [13](#_Toc3801_WPSOffice_Level2)

[六、用户手册](#_Toc166_WPSOffice_Level1) [26](#_Toc166_WPSOffice_Level1)

[七、调试及总结](#_Toc15069_WPSOffice_Level1) [26](#_Toc15069_WPSOffice_Level1)

## **实验四 分析纽约市高中数据**

1. **问题描述** 
   1. **实验目的**

通过本项目进一步熟悉通过Pandas库中的函数清洗、分析、可视化数据。

* 1. **实验开发环境和工具**

可以在Windows或Linux操作系统上搭建开发环境，使用Anaconda科学计算包安装Python 3，里面会包含大多数科研常用的库。.ipynb文件推荐使用Anaconda内置的web版Jupyter Notebook。

* 1. **实验内容**

在之前的三个任务[1-3]中，我们探索了纽约公立高中SAT分数和人口统计学因素之间的关系。简单介绍一下背景知识，SAT(学术能力测验)是美国高中毕业生每年都会参加的一项考试。SAT有三个部分，每个部分的最高分是800分。大学用SAT的成绩来决定录取哪些学生，高的SAT平均分通常意味着一个好学校。

纽约市已经公布了[高中学生SAT分数](https://data.cityofnewyork.us/Education/2012-SAT-Results/f9bf-2cp4)的数据，以及其它的人口统计数据。在之前的三个任务中，我们将下列数据集合并为一个单独的干净的pandas数据帧对象。

·[按学校分类的SAT成绩](https://data.cityofnewyork.us/Education/SAT-Results/f9bf-2cp4)---纽约市每一所高中的SAT成绩。

·[上课出勤信息](https://data.cityofnewyork.us/Education/2010-2011-School-Attendance-and-Enrollment-Statist/7z8d-msnt)---纽约市各学校的出勤信息。

·[班级规模](https://data.cityofnewyork.us/Education/2010-2011-Class-Size-School-level-detail/urz7-pzb3)--每个学校的班级规模信息。

·[AP测试结果](https://data.cityofnewyork.us/Education/AP-College-Board-2010-School-Level-Results/itfs-ms3e)---每所高中的大学预修(AP)考试成绩(通过某一科目的选修AP考试可以获得该科目的大学学分)。

·[毕业的结果](https://data.cityofnewyork.us/Education/2005-2010-Graduation-Outcomes-School-Level/vh2h-md7a)---学生毕业的百分比，以及其它结果信息。

·[人口统计信息](https://data.cityofnewyork.us/Education/2006-2012-School-Demographics-and-Accountability-S/ihfw-zy9j)---每个学校的人口统计信息。

·[学校调查](https://data.cityofnewyork.us/Education/2011-NYC-School-Survey/mnz3-dyi8)问卷---对每个学校的家长、教师和学生进行调查问卷。

纽约市移民人口众多，非常多样化，所以比较人口统计学因素（如种族，收入和性别）与SAT分数的关系是一种很好的方式来确定SAT是否是一个公平的测试。例如，如果某些种族群体在SAT考试中一直表现得更好，我们就可以得到一些证据表明SAT是不公平的。

在最后一次任务中，我们开始进行一些分析。我们将在这次任务中扩展这种分析。我们已经包含了读取所有数据、合并数据并在notebook中创建关联的代码。数据帧 combined 包含了我们将在分析中使用的所有数据。

1. **需求和规格说明**

**1.了解项目和数据集：**

具体编程任务如下：

**·**在Jupyter notebook中设置matplotlib使得其可以工作**。**

**·**在**combined**中有几个字段来自于对[家长、教师和学生的问卷调查](https://data.cityofnewyork.us/Education/NYC-School-Survey-2011/mnz3-dyi8)。

制作一个条形图，显示这些字段和sat\_score之间的相关性。

·可在配套的notebook中的**survey\_fields** 变量中找到这些字段列表。

**·**请从纽约市的网站上[下载](https://data.cityofnewyork.us/Education/NYC-School-Survey-2011/mnz3-dyi8)的**zip** 文件，从分析其中的数据字典，看看能否发现令人惊讶的关联。

**·**把结果写在一个标记单元格（Markdown cell）中**。**

**2.探索学校安全系数和SAT分数：**

在上一个任务中，你可能已经注意到：**saf\_t\_11**和saf\_s\_11这两个衡量教师和学生如何感受学校安全性的指标与sat\_score高度相关。在这个任务中，我们将进一步研究这一关系，并试图找出哪些学校的安全分数较低。

具体编程任务如下：

**·**调查安全分数

·将数据帧combined中的saf\_s\_11列与sat\_score绘制成散点图。

·把你关于安全系数和SAT成绩的结论写在标记单元格中。

**·**标出安全分数

·计算每个地区的平均安全得分

·制作一张按地区显示安全分数的地图。

·把按地理区域的安全性的结论写在标记单元格中。你可能想了解一下[纽约市的行政区](https://www.nycgo.com/boroughs-neighborhoods)。

**3.探索种族和SAT分数：**

有一些列表示每个种族在给定学校的百分比：

**·**white\_per

**·**asian\_per

**·**black\_per

**·**hispanic\_per

通过绘制这些列与**sat\_score**之间的相关性，我们可以确定在SAT成绩中是否存在种族差异。

具体编程任务如下：

**·**调查SAT分数的种族差异。

**·**绘制一个柱状图（bar plot），显示上述列与**sat\_score**之间的相关性

**·**在标记单元格中记录你的发现，并注意观察是否有出人意料的相关性。

**·**探索那些SAT分数低，西班牙语成绩高（hispanic\_per）的学校。

**·**绘制一个hispanic\_per与**sat\_score**的散点图

**·**散点图显示了什么?在标记单元格中记录任何有趣的发现**。**

**·**研究一下西班牙语（hispanic\_per）平均水平超过95%的学校**。**

**·**在数据中查找到相关学校的名称。

**·**使用维基百科和谷歌来研究学校的名字。

**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

**·**研究一下西班牙语（hispanic\_per）平均分值低于10%并且SAT平均分值超过1800分的学校。

**·**在数据中查找到相关学校的名称。

**·**使用维基百科和谷歌来研究学校的名字。

**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

**4.探索性别和SAT分数：**

数据中有两列表示学校中男女学生的比例：

**·**male\_per

**·**female\_per

我们可以绘制出上述每个百分比和sat\_score之间的相关性

具体编程任务如下：

**·**分析SAT分数的性别差异

**·绘制**一个柱状图，显示上面的列与sat\_score之间的相关性。

**·**在标记单元格中记录你的发现，并注意观察是否有出人意料的相关性

**·**探索那些有SAT高分和女性平均成绩（female\_per）高的学校。

**·**绘制一个female\_per与sat\_score的散点图。

**·**观察散点图所显示的信息，在标记单元格中记录任何有趣的观察结果**。**

**·**研究一下女性平均成绩（female\_per）超过60%且SAT平均成绩超过1700分的学校**。**

**·**在数据中查找相关学校的名称。

**·**使用维基百科和谷歌来研究学校的名字。

**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

**5.探索AP分数和SAT分数：**

在美国，高中学生参加AP考试来获得大学学分。AP考试分成很多不同的专业主题。参加AP考试的学生数量与学校的SAT成绩高度相关，这是有道理的。让我们来探讨一下这种关系。因为total\_enrollment与sat\_score高度相关，所以我们不想对结果产生偏差。相反，我们将查看每所学校至少参加过一次AP考试的学生百分比。

具体编程任务如下：

**·**计算每个学校参加AP考试的学生百分比。

**·**将AP Test Takers列除以**total\_enrollment** 列。

**·**列名“AP Test Takers”的末尾有一个空格——别忘了加进去。

**·**将结果分配给**ap\_per** 列。

**·**调查AP分数和SAT分数之间的关系

**·**绘制**ap\_per**与**sat\_score**的散点图

**·**观察散点图显示的信息，并在标记单元中记录任何有趣的观察结果**。**

**6.进一步探索：**

在这个指导项目中，我们对人口统计数据和SAT分数之间的关系做了相当多的调查。不过，还有相当多的分析工作要做。下面是一些可能的后续步骤：

·确定班级规模和SAT分数之间是否存在相关性

·找出哪个社区有最好的学校

·如果我们将这些信息与包含财产值的数据集结合起来，我们就能找到拥有好学校且最便宜的社区。

·调研家长，老师和学生对调查问卷回答之间的差异。

·根据sat\_score和其他属性为学校分配分数。

**三、概要设计**

**1.了解项目和数据集**

**2.探索学校安全系数和SAT分数：**

在上一个任务中，已经注意到：**saf\_t\_11**和saf\_s\_11这两个衡量教师和学生如何感受学校安全性的指标与sat\_score高度相关。在这个任务中，我们将进一步研究这一关系，并试图找出哪些学校的安全分数较低。

**3.探索种族和SAT分数：**

有一些列表示每个种族在给定学校的百分比：

**·**white\_per

**·**asian\_per

**·**black\_per

**·**hispanic\_per

通过绘制这些列与**sat\_score**之间的相关性，我们可以确定在SAT成绩中是否存在种族差异。

**4.探索性别和SAT分数：**

数据中有两列表示学校中男女学生的比例：

**·**male\_per

**·**female\_per

绘制出上述每个百分比和sat\_score之间的相关性

**5.探索AP分数和SAT分数：**

在美国，高中学生参加AP考试来获得大学学分。AP考试分成很多不同的专业主题。参加AP考试的学生数量与学校的SAT成绩高度相关，这是有道理的。让我们来探讨一下这种关系。因为total\_enrollment与sat\_score高度相关，所以我们不想对结果产生偏差。相反，我们将查看每所学校至少参加过一次AP考试的学生百分比。

**6.进一步探索：**

在这个指导项目中，我们对人口统计数据和SAT分数之间的关系做了相当多的调查。不过，还有相当多的分析工作要做。下面是一些可能的后续步骤：

·确定班级规模和SAT分数之间是否存在相关性

·找出哪个社区有最好的学校

·如果我们将这些信息与包含财产值的数据集结合起来，我们就能找到拥有好学校且最便宜的社区。

·调研家长，老师和学生对调查问卷回答之间的差异。

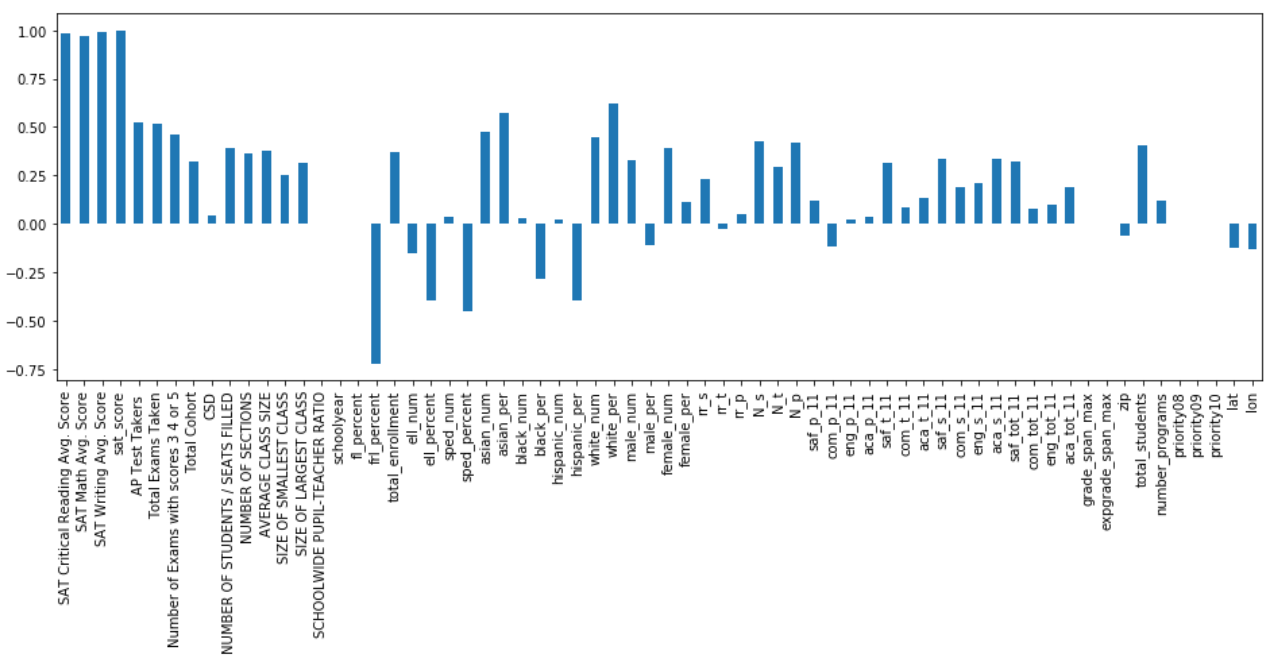
·根据sat\_score和其他属性为学校分配分数。

**四、详细设计**

**4.1设计思想**

1. **了解项目和数据集**

制作一个条形图，显示这些字段和sat\_score之间的相关性。



**2.探索学校安全系数和SAT分数：**

**·**调查安全分数

·将数据帧combined中的saf\_s\_11列与sat\_score绘制成散点图。

·把你关于安全系数和SAT成绩的结论写在标记单元格中。

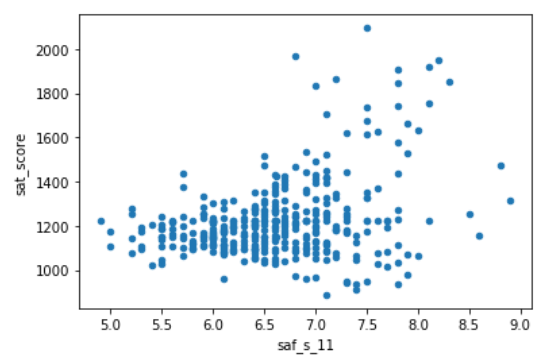
**·**标出安全分数

·计算每个地区的平均安全得分

·制作一张按地区显示安全分数的地图。

·把按地理区域的安全性的结论写在标记单元格中。你可能想了解一下[纽约市的行政区](https://www.nycgo.com/boroughs-neighborhoods)。

combined.plot.scatter(x='saf\_s\_11', y='sat\_score')



安全系数saf\_s\_11和SAT成绩，由之前计算可以知道相关系数为0.337639呈正相关，由散点图也可以看出sat\_score随saf\_s\_11的增长呈缓慢增长趋势。

safe = combined.groupby("saf\_s\_11").agg(numpy.mean)

safe.reset\_index(inplace = True)

low\_enrollment = combined[combined["total\_enrollment"] < 1000]

low\_enrollment = low\_enrollment[low\_enrollment["sat\_score"] < 1000]

首先要安装pyproj和basemap包，具体安装过程及其中出现的问题见调试过程

from mpl\_toolkits.basemap import Basemap

m = Basemap(

projection='merc',

llcrnrlat=40.496044,

urcrnrlat=40.915256,

llcrnrlon=-74.255735,

urcrnrlon=-73.700272,

resolution='i'

)

m.drawmapboundary(fill\_color='#85A6D9')

m.drawcoastlines(color='#6D5F47', linewidth=.4)

m.drawrivers(color='#6D5F47', linewidth=.4)

longitudes = combined["lon"].tolist()

latitudes = combined["lat"].tolist()

m.scatter(longitudes, latitudes, s=50, zorder=2, latlon=True, c=combined["ell\_percent"], cmap="summer")

plt.show()



利用basemap包制作了一张按地区显示安全分数的地图。

在这个过程中，我们学会了：

如何创建学校和区级地图。

如何找到相关性，且相关性意味着什么。

为什么我们应该绘制数据，而不是仅仅依赖相关性r值。

**3.探索种族和SAT分数：**

**·**调查SAT分数的种族差异。

**·**绘制一个柱状图（bar plot），显示上述列与**sat\_score**之间的相关性

**·**在标记单元格中记录你的发现，并注意观察是否有出人意料的相关性。

**·**探索那些SAT分数低，西班牙语成绩高（hispanic\_per）的学校。

**·**绘制一个hispanic\_per与**sat\_score**的散点图

**·**散点图显示了什么?在标记单元格中记录任何有趣的发现**。**

**·**研究一下西班牙语（hispanic\_per）平均水平超过95%的学校**。**

**·**在数据中查找到相关学校的名称。

**·**使用维基百科和谷歌来研究学校的名字。

**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

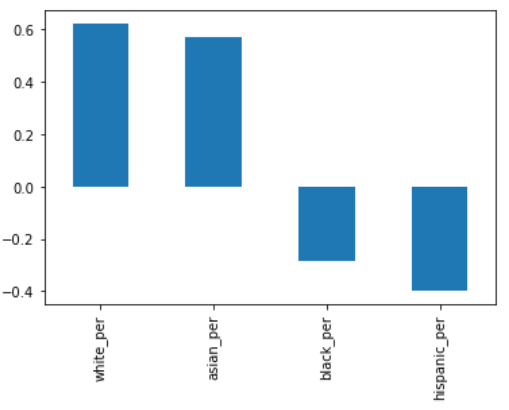
**·**研究一下西班牙语（hispanic\_per）平均分值低于10%并且SAT平均分值超过1800分的学校。

**·**在数据中查找到相关学校的名称。

**·**使用维基百科和谷歌来研究学校的名字。

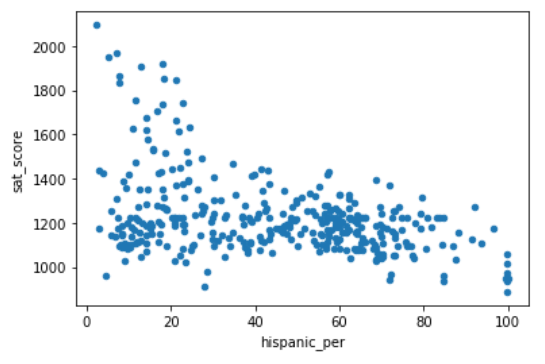
**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

correlations[["white\_per","asian\_per","black\_per","hispanic\_per"]].plot(kind='bar')



由该图可以直观的看出["white\_per","asian\_per","black\_per","hispanic\_per"]四列与“sat—\_score”之间的相关性。

combined.plot.scatter(x='hispanic\_per', y='sat\_score')

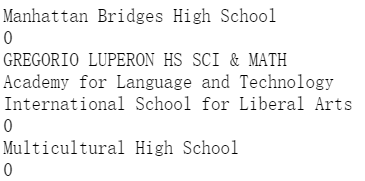


根据hispanic\_per与**sat\_score**的散点图的分布，可以看出两者之间呈负相关，证实了上述柱状图该数据的准确性。

for i in range(363):

if combined["hispanic\_per"][i]>95:

print(combined["SchoolName"][i])

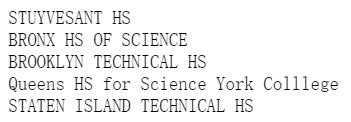


曼哈顿桥中学Manhattan Bridges High School 朱莉娅马斯特曼实验室与示范学校Julia R. Masterman Laboratory and Demonstration School .美国排名前十的高中。 Academy for Language and Technology International School for Liberal Arts 语言与技术学院 国际文理学院 multicultural high school 多元文化高中

for i in range(363):

if combined["hispanic\_per"][i]<10 and combined["sat\_score"][i]>1800:

print(combined["SchoolName"][i])



STUYVESANT HS

布朗克斯科学

布鲁克林技术HS

皇后学院科学约克学院

史坦顿岛技术专员

**4.探索性别和SAT分数：**

**·**分析SAT分数的性别差异

**·绘制**一个柱状图，显示上面的列与sat\_score之间的相关性。

**·**在标记单元格中记录你的发现，并注意观察是否有出人意料的相关性

**·**探索那些有SAT高分和女性平均成绩（female\_per）高的学校。

**·**绘制一个female\_per与sat\_score的散点图。

**·**观察散点图所显示的信息，在标记单元格中记录任何有趣的观察结果**。**

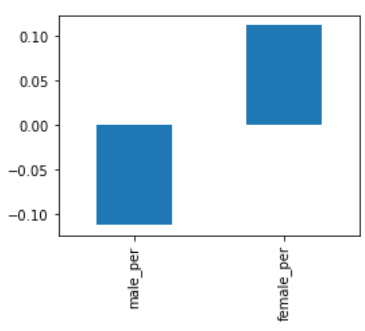
**·**研究一下女性平均成绩（female\_per）超过60%且SAT平均成绩超过1700分的学校**。**

**·**在数据中查找相关学校的名称。

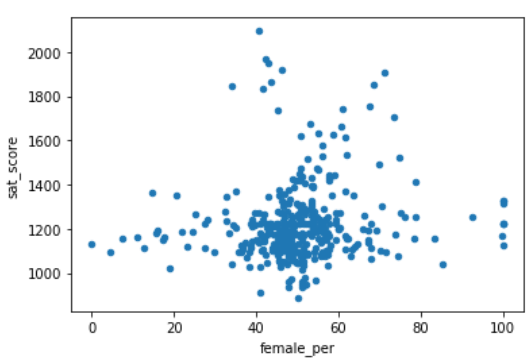
**·**使用维基百科和谷歌来研究学校的名字。

**·**这些学校有什么有趣的地方吗?在标记单元格中记录你的发现。

correlations[["male\_per","female\_per"]].plot(kind='bar',figsize=(4,3))



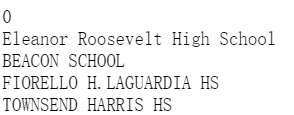
combined.plot.scatter(x='female\_per', y='sat\_score')



for i in range(363):

if combined["female\_per"][i]>60 and combined["sat\_score"][i]>1700:

print(combined["SchoolName"][i])



埃莉诺·罗斯福高中

信标学校

FIORELLO H.LAGUARDIA HS

汤森·哈里斯

**5.探索AP分数和SAT分数：**

**·**计算每个学校参加AP考试的学生百分比。

**·**将AP Test Takers列除以**total\_enrollment** 列。

**·**列名“AP Test Takers”的末尾有一个空格——别忘了加进去。

**·**将结果分配给**ap\_per** 列。

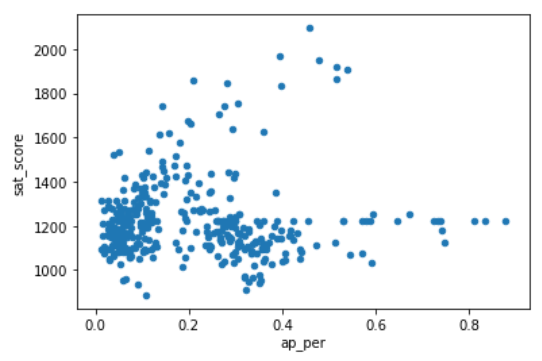
**·**调查AP分数和SAT分数之间的关系

**·**绘制**ap\_per**与**sat\_score**的散点图

**·**观察散点图显示的信息，并在标记单元中记录任何有趣的观察结果**。**

combined["ap\_per"] = combined['AP Test Takers ']/combined['total\_enrollment']

combined.plot.scatter(x='ap\_per', y='sat\_score')



由散点图观察可知**ap\_per**与**sat\_score相关性较小。**

**6.进一步探索：**

在这个指导项目中，我们对人口统计数据和SAT分数之间的关系做了相当多的调查。不过，还有相当多的分析工作要做。下面是一些可能的后续步骤：

·确定班级规模和SAT分数之间是否存在相关性

·找出哪个社区有最好的学校

·如果我们将这些信息与包含财产值的数据集结合起来，我们就能找到拥有好学校且最便宜的社区。

·调研家长，老师和学生对调查问卷回答之间的差异。

·根据sat\_score和其他属性为学校分配分数。

根据相关信息可知，[班级规模](https://data.cityofnewyork.us/Education/2010-2011-Class-Size-School-level-detail/urz7-pzb3)为每个学校的班级规模信息。[学校调查](https://data.cityofnewyork.us/Education/2011-NYC-School-Survey/mnz3-dyi8)问卷为对每个学校的家长、教师和学生进行调查问卷。

**五、结果分析：**

**5.1实现注释程序清单**

|  |
| --- |
| **#读入数据**  import pandas  import numpy  import re  data\_files = [  "ap\_2010.csv",  "class\_size.csv",  "demographics.csv",  "graduation.csv",  "hs\_directory.csv",  "sat\_results.csv"  ]  data = {}  for f in data\_files:  d = pandas.read\_csv("schools/{0}".format(f))  data[f.replace(".csv", "")] = d **#读入调查问卷** all\_survey = pandas.read\_csv("schools/survey\_all.txt", delimiter="\t", encoding='windows-1252')  d75\_survey = pandas.read\_csv("schools/survey\_d75.txt", delimiter="\t", encoding='windows-1252')  survey = pandas.concat([all\_survey, d75\_survey], axis=0)  survey["DBN"] = survey["dbn"]  survey\_fields = [  "DBN",  "rr\_s",  "rr\_t",  "rr\_p",  "N\_s",  "N\_t",  "N\_p",  "saf\_p\_11",  "com\_p\_11",  "eng\_p\_11",  "aca\_p\_11",  "saf\_t\_11",  "com\_t\_11",  #"eng\_t\_10",  "aca\_t\_11",  "saf\_s\_11",  "com\_s\_11",  "eng\_s\_11",  "aca\_s\_11",  "saf\_tot\_11",  "com\_tot\_11",  "eng\_tot\_11",  "aca\_tot\_11",  ]  survey = survey[survey\_fields]  data["survey"] = survey #**添加DBN列** data["hs\_directory"]["DBN"] = data["hs\_directory"]["dbn"]  def pad\_csd(num):  string\_representation = str(num)  if len(string\_representation) > 1:  return string\_representation  else:  return "0" + string\_representation    data["class\_size"]["padded\_csd"] = data["class\_size"]["CSD"].apply(pad\_csd)  data["class\_size"]["DBN"] = data["class\_size"]["padded\_csd"] + data["class\_size"]["SCHOOL CODE"] #**将列内容转换为数值类型** cols = ['SAT Math Avg. Score', 'SAT Critical Reading Avg. Score', 'SAT Writing Avg. Score']  for c in cols:  data["sat\_results"][c] = pandas.to\_numeric(data["sat\_results"][c], errors="coerce")  data['sat\_results']['sat\_score'] = data['sat\_results'][cols[0]] + data['sat\_results'][cols[1]] + data['sat\_results'][cols[2]]  def find\_lat(loc):  coords = re.findall("\(.+, .+\)", loc)  lat = coords[0].split(",")[0].replace("(", "")  return lat  def find\_lon(loc):  coords = re.findall("\(.+, .+\)", loc)  lon = coords[0].split(",")[1].replace(")", "").strip()  return lon  data["hs\_directory"]["lat"] = data["hs\_directory"]["Location 1"].apply(find\_lat)  data["hs\_directory"]["lon"] = data["hs\_directory"]["Location 1"].apply(find\_lon)  data["hs\_directory"]["lat"] = pandas.to\_numeric(data["hs\_directory"]["lat"], errors="coerce")  data["hs\_directory"]["lon"] = pandas.to\_numeric(data["hs\_directory"]["lon"], errors="coerce") #**压缩数据集** class\_size = data["class\_size"]  class\_size = class\_size[class\_size["GRADE "] == "09-12"]  class\_size = class\_size[class\_size["PROGRAM TYPE"] == "GEN ED"]  class\_size = class\_size.groupby("DBN").agg(numpy.mean)  class\_size.reset\_index(inplace=True)  data["class\_size"] = class\_size  data["demographics"] = data["demographics"][data["demographics"]["schoolyear"] == 20112012]  data["graduation"] = data["graduation"][data["graduation"]["Cohort"] == "2006"]  data["graduation"] = data["graduation"][data["graduation"]["Demographic"] == "Total Cohort"] #**将AP分数转换为数值类型** cols = ['AP Test Takers ', 'Total Exams Taken', 'Number of Exams with scores 3 4 or 5']  for col in cols:  data["ap\_2010"][col] = pandas.to\_numeric(data["ap\_2010"][col], errors="coerce" #**将AP分数转换为数值类型** cols = ['AP Test Takers ', 'Total Exams Taken', 'Number of Exams with scores 3 4 or 5']  for col in cols:  data["ap\_2010"][col] = pandas.to\_numeric(data["ap\_2010"][col], errors="coerce") #**合并数据集** combined = data["sat\_results"]  combined = combined.merge(data["ap\_2010"], on="DBN", how="left")  combined = combined.merge(data["graduation"], on="DBN", how="left")  to\_merge = ["class\_size", "demographics", "survey", "hs\_directory"]  for m in to\_merge:  combined = combined.merge(data[m], on="DBN", how="inner")  combined = combined.fillna(combined.mean())  combined = combined.fillna(0) #**为映射添加学区列** def get\_first\_two\_chars(dbn):  return dbn[0:2]  combined["school\_dist"] = combined["DBN"].apply(get\_first\_two\_chars) #**计算相关性** correlations = combined.corr()  correlations = correlations["sat\_score"]  print(correlations)  %matplotlib inline  import matplotlib  import matplotlib.pyplot as plt  correlations.plot(kind='bar',figsize=(16,5))  combined.plot.scatter(x='saf\_s\_11', y='sat\_score')  safe = combined.groupby("saf\_s\_11").agg(numpy.mean)  safe.reset\_index(inplace = True)  low\_enrollment = combined[combined["total\_enrollment"] < 1000]  low\_enrollment = low\_enrollment[low\_enrollment["sat\_score"] < 1000]  from mpl\_toolkits.basemap import Basemap  m = Basemap(  projection='merc',  llcrnrlat=40.496044,  urcrnrlat=40.915256,  llcrnrlon=-74.255735,  urcrnrlon=-73.700272,  resolution='i'  )  m.drawmapboundary(fill\_color='#85A6D9')  m.drawcoastlines(color='#6D5F47', linewidth=.4)  m.drawrivers(color='#6D5F47', linewidth=.4)  longitudes = combined["lon"].tolist()  latitudes = combined["lat"].tolist()  m.scatter(longitudes, latitudes, s=50, zorder=2, latlon=True, c=combined["ell\_percent"], cmap="summer")  plt.show()  correlations[["white\_per","asian\_per","black\_per","hispanic\_per"]].plot(kind='bar')  combined.plot.scatter(x='hispanic\_per', y='sat\_score')  for i in range(363):  if combined["hispanic\_per"][i]>95:  print(combined["SchoolName"][i])  for i in range(363):  if combined["hispanic\_per"][i]<10 and combined["sat\_score"][i]>1800:  print(combined["SchoolName"][i])  correlations[["male\_per","female\_per"]].plot(kind='bar',figsize=(4,3))  combined.plot.scatter(x='female\_per', y='sat\_score')  for i in range(363):  if combined["female\_per"][i]>60 and combined["sat\_score"][i]>1700:  print(combined["SchoolName"][i])  combined["ap\_per"] = combined['AP Test Takers ']/combined['total\_enrollment']  combined.plot.scatter(x='ap\_per', y='sat\_score') |

1. **用户手册**

若用户需要对类似问题进行分析时，只需按照步骤分析修改图图片显示要求，python需要安装matplotlib、pyproj、basemap、pandas、numpy、re包、源代码选在jupyter notebook上运行。

**七、调试及总结**

1.关于basemap包安装过程中遇到的问题：

环境：Windows10

基于 Python3.6 版本的 Anaconda

在 windows10 下，安装步骤如下：

（1）首先，从下面的网站下载 basemap 和 pyproj 的whl文件

https://www.lfd.uci.edu/~gohlke/pythonlibs/

对应 python3.6 版，windows 系统下的文件如下：

Win+r cmd 输入控制命令python查看Python版本3.6选择cp36

basemap-2.4.2-cp36-cp36m-win\_amd64.whl

pyproj-1.9.5.1-cp36-cp36m-win\_amd64.whl

（2）安装 pyproj 的 whl 文件， 如下：

pip install 路径+文件名

（3）安装basemap 的 whl 文件，如下：

（4）测试下 basemap 是否安装成功

1.在控制窗口python环境下输入

from mpl\_toolkits.basemap import Basemap

观察是否报错，如果没有报错则在jupyter notebook中输入该语句测试。

报错输入import sys；sys.path查看python包应该安装的路径。并用pip list查看basemap是否安装成功。

1. 若从控制命令窗口可以导入而jupyter notebook中显示没有该模块，则同样运用import sys；sys.path查看安装路径，我们如果发现两者路径不同，则需要在原python文件site\_packages中添加txt文件并写入jupyter notebook文件site\_packages路径，将后缀名改为pth即可。