

BÀI TẬP THỰC HÀNH MÔN THỐNG KÊ MÁY TÍNH VÀ ỨNG DỤNG

- ❖ Bài tập được thiết kế theo từng lab, mỗi lab là 3 tiết có sự hướng dẫn của GV.
- ❖ Cuối mỗi buổi thực hành, sinh viên nộp lại phần bài tập mình đã thực hiện cho GV hướng dẫn.
- ❖ Những câu hỏi mở rộng/khó giúp sinh viên trau dồi thêm kiến thức của môn học. Sinh viên phải có trách nhiệm nghiên cứu, tìm câu trả lời nếu chưa thực hiện xong trong giờ thực hành.

NỘI DUNG

LAB 1: LÀM QUEN VỚI PYTHON	3
LAB 2: THỐNG KÊ MÔ TẢ	19
LAB 3: THỐNG KÊ MÔ TẢ	35
LAB 4: ƯỚC LƯỢNG	41
LAB 5: KIỂM ĐỊNH.....	51
LAB 6: HỒI QUY TUYẾN TÍNH ĐƠN BIẾN	54
LAB 7: HỒI QUY ĐA BIẾN.....	62
PHỤ LỤC: CÁC DATASET DÙNG TRONG BÀI TẬP THỰC HÀNH.....	63

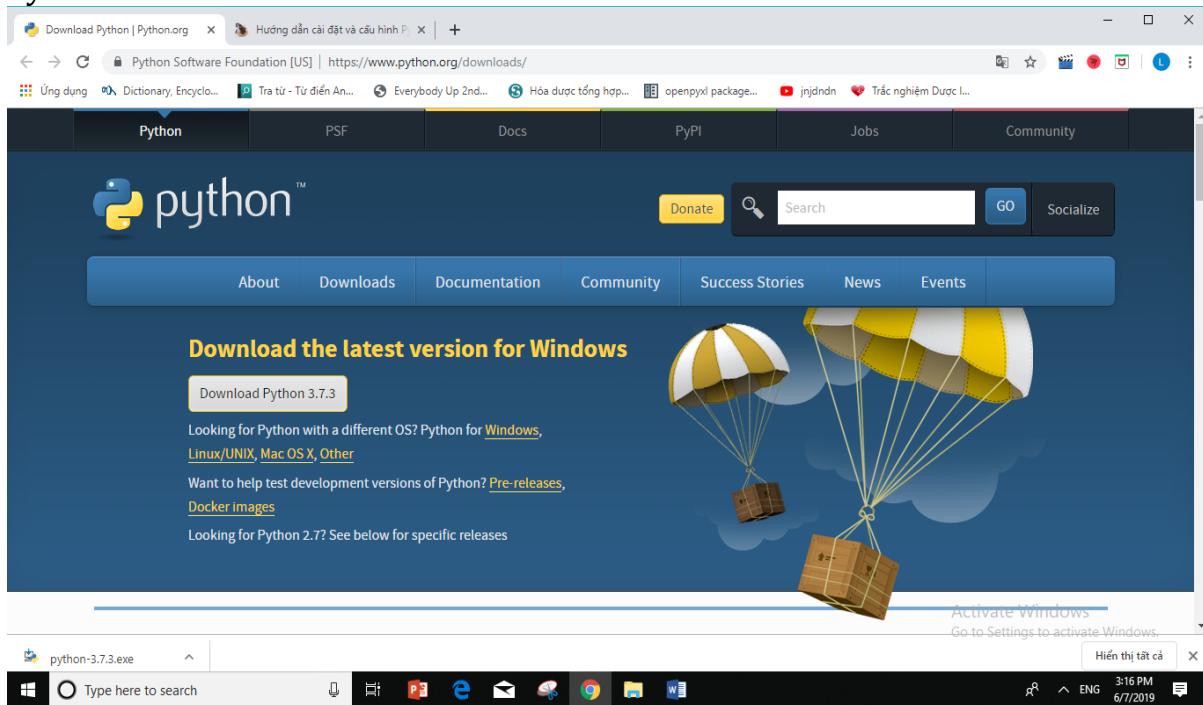
LAB 1: LÀM QUEN VỚI PYTHON

Nội dung:

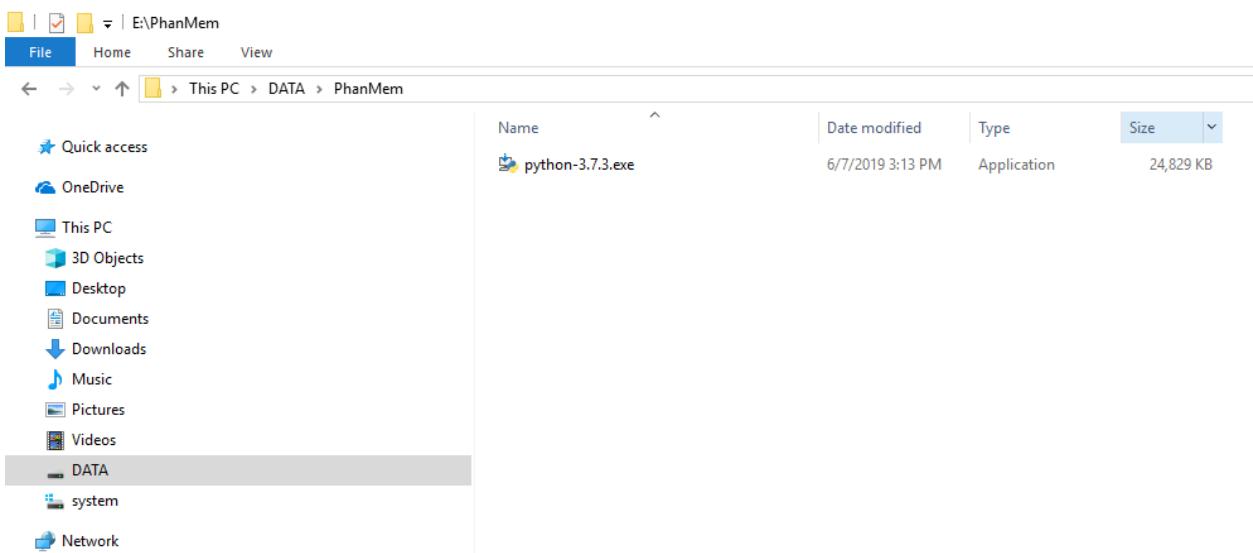
1. Download Python
2. Cài đặt Python
3. Làm quen với Python
4. Các IDE cho Python
5. Các package quan trọng sử dụng trong thống kê
6. Bài tập

1. Download Python

Để download Python, bạn truy cập địa chỉ: <https://www.python.org/downloads/>
Nhấn vào nút **Download Python 3.7.3** để download phiên bản mới nhất của Python.



Sau khi download xong bạn có 1 file **python-3.7.3.exe**.

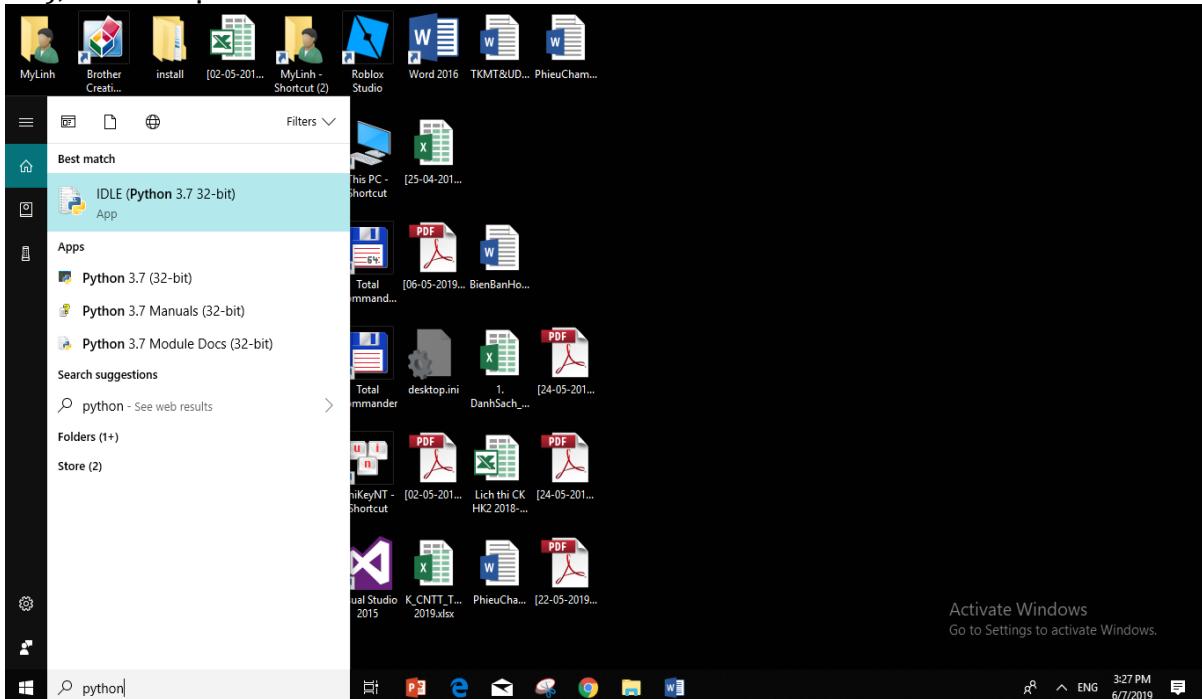


2. Cài đặt Python

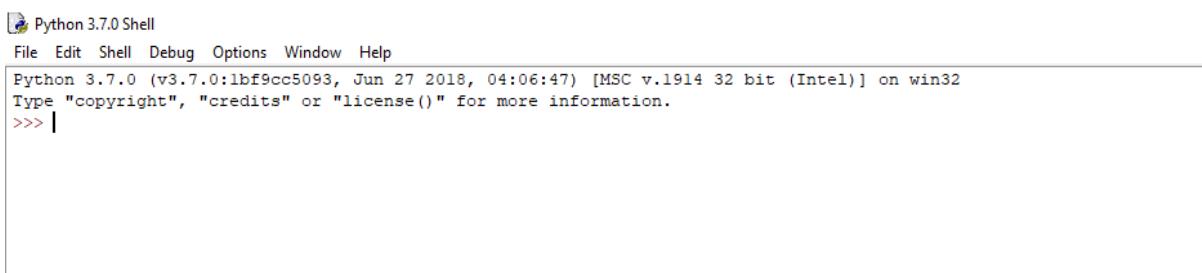
Thực thi file bạn download được ở bước trên để bắt đầu cài đặt. Chọn "**Customize Installation**" để bạn có thể tùy chọn ví trí **Python** sẽ được cài đặt. Thực hiện theo các bước để hoàn thành việc cài đặt.

3. Làm quen với Python

Vào mục tìm kiếm của Window gõ chữ "Python", sẽ xuất hiện IDLE (Python 3.7 32-bit), nhấn chọn vào IDLE trên.

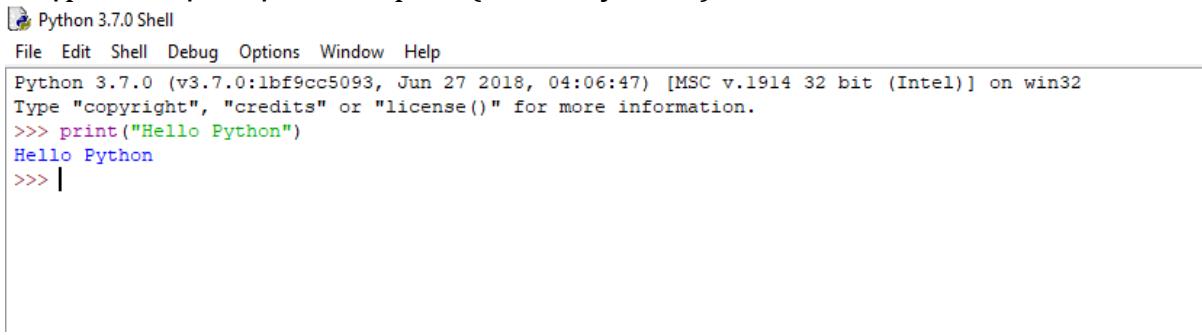


Chương trình "Python Shell" đã được thực thi, nó là một chương trình giúp bạn viết mã Python. Dưới đây là hình ảnh của Python Shell:



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Nhập vào một đoạn code: `print("Hello Python")` và nhấn Enter.



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("Hello Python")
Hello Python
>>> |
```

Sau khi bạn cài đặt xong Python, ta có thêm một công cụ Python Shell, đây là một IDE (Integrated Development Environment) giúp bạn viết mã Python. Nếu bạn không muốn sử dụng Python Shell bạn có thể sử dụng một IDE khác.

4. Các IDE cho Python

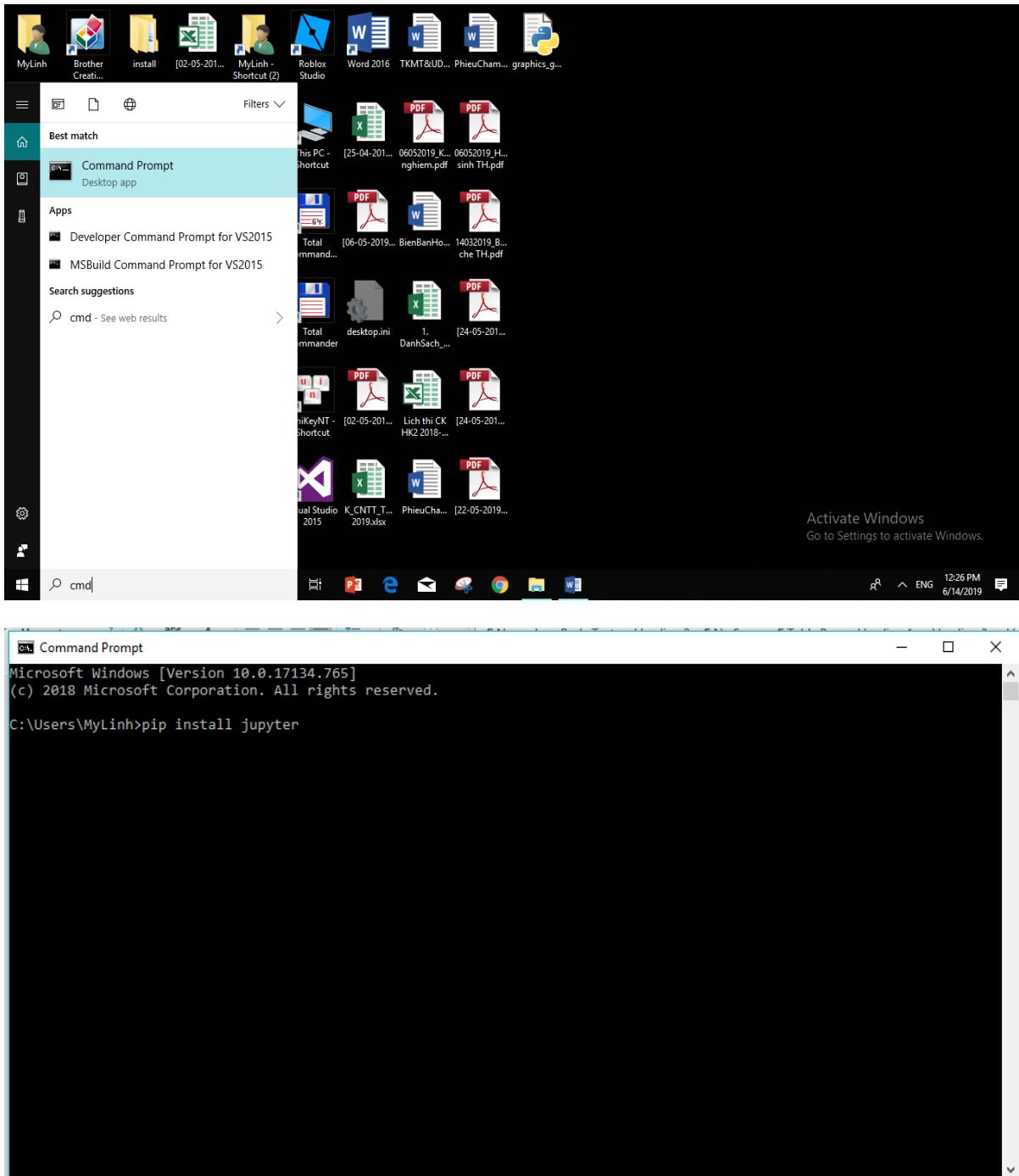
Một số IDE giúp bạn lập trình Python:

- PyCharm
- Anaconda
- Jupiter Notebook
-

Hướng dẫn cài đặt Jupiter Notebook:

Sau khi cài đặt xong Python 3.7, vào **Command Promt** gõ lệnh: **pip install jupyter**

Nếu chương trình không nhận biết được lệnh trên thì gõ lệnh **py -m pip install jupyter**



Quá trình cài đặt diễn ra bình thường nếu không có dòng nào màu đỏ.
Hướng dẫn sử dụng jupyter notebook:

- Khởi động Jupyter Notebook:** Ở command prompt, nhập vào câu lệnh dưới đây, server sẽ được khởi động, và có thể xác nhận việc hiển thị giao diện của Jupyter Notebook ở browser.

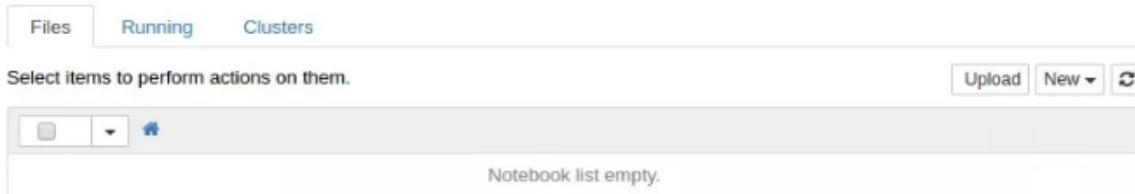
```
jupyter notebook
```

Nếu chương trình không nhận biết được lệnh trên thì gõ lệnh **py -m jupyter notebook**

Mặc định thì Jupyter Notebook sẽ sử dụng cổng 8888, tuy nhiên cũng có thể chỉ định cổng khác bằng tham số --port. Xem ví dụ dưới:

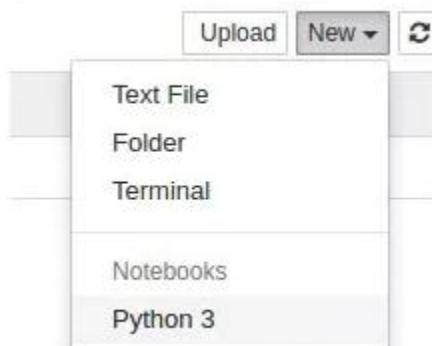
```
jupyter notebook --port 9000
```

Sau khi khởi động, màn hình dưới đây sẽ hiển thị. Ở màn hình này, danh sách các file trong thư mục hiện tại sẽ được hiển thị.



Home page của Jupyter Notebook

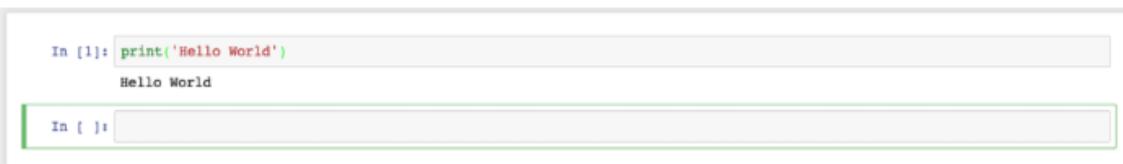
- Cách mở một Notebook mới:** Click vào button 「New」 ở góc bên phải, rồi lựa chọn 「Python 3」 để có thể mở một Notebook mới.



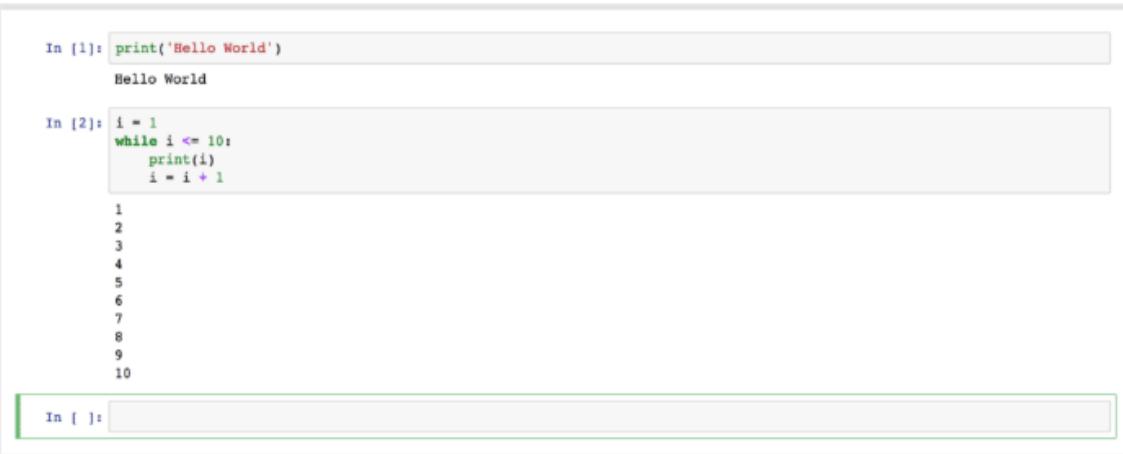
- Làm việc với Notebook:** Một notebook bao gồm nhiều cell (ô). Khi tạo mới một notebook, bạn luôn được tạo sẵn một cell rỗng đầu tiên.



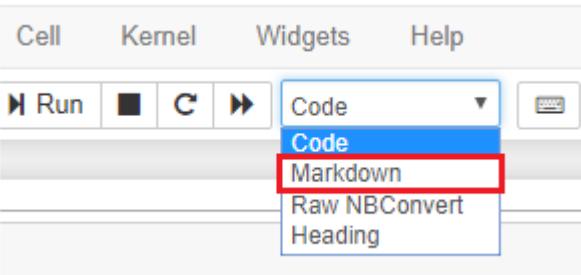
Cell trên có kiểu là “Code”, điều đó có nghĩa là bạn có thể gõ code Python vào cell này. Để thực thi code, bạn có thể nhấn nút Run cell hoặc nhấn phím Ctrl + Enter.



Kết quả được hiển thị tại ô bên dưới. Một cell rỗng sẽ được tạo sau khi bạn thực thi code. Hãy gõ tiếp một đoạn code Python dưới đây để thử nghiệm:



Bạn có thể chuyển loại cell từ **Code** thành **Markdown** để viết những đoạn văn bản giải thích code của bạn. Để chuyển đổi, bạn click vào ComboBox **Code** và chọn **Markdown** như hình:



Sau khi chuyển, hãy nhập ngay một đoạn **Markdown** sau để thử nghiệm

The screenshot shows a Jupyter Notebook interface. At the top, there's a red header bar with the title. Below it is a green-bordered cell containing Markdown text:

```
# This is a headline
## Sub headline
**Text**
More Text
```

Below this is another cell with Python code:

```
In [1]: print('Hello World')
Hello World
```

Then comes a cell with a while loop:

```
In [2]: i = 1
while i <= 10:
    print(i)
    i = i + 1
```

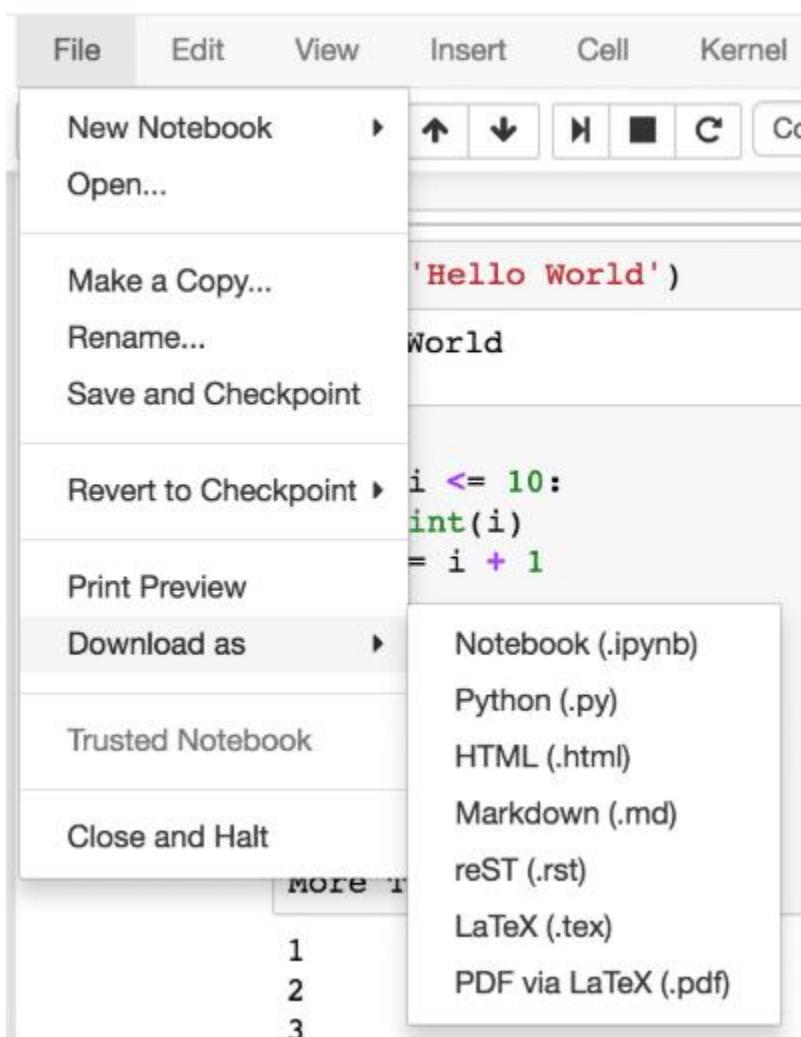
The output of this cell is the numbers 1 through 10, each on a new line.

At the bottom of the notebook, there's a text area with the following content:

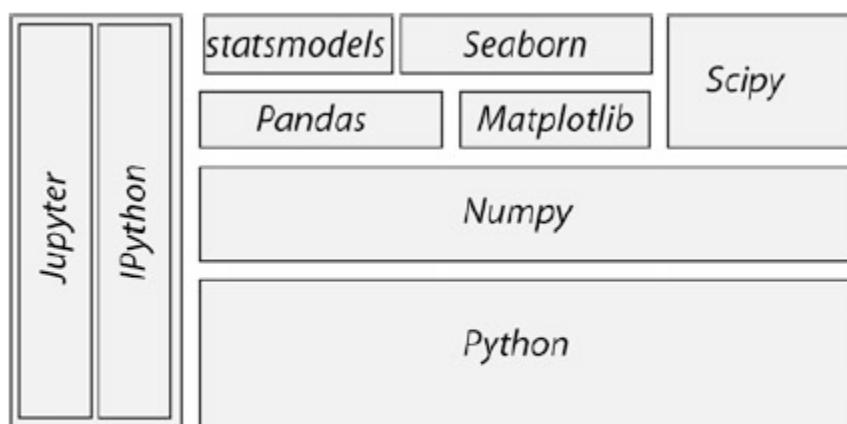
This is a headline
Sub headline
Text
More Text

Nếu bạn muốn chỉnh sửa đoạn **Markdown** vừa thực thi thì chỉ việc click vào kết quả vừa xuất hiện và bạn sẽ được chuyển sang chế độ chỉnh sửa.

4. **Checkpoint:** Một trong những chức năng cực hay của Jupyter Notebook là Checkpoints. Bằng cách tạo các Checkpoints lưu trạng thái hiện tại của notebook, Jupyter Notebook cho phép bạn có thể quay lại thời điểm tạo Checkpoints để kiểm tra hoặc hoàn tác trước đó.
Để tạo Checkpoint, chọn **File -> Save and Checkpoint**. Nếu bạn muốn xem lại các Checkpoints trước đó thì chọn **File -> Revert to Checkpoint**.
5. **Chức năng Export notebook:** Jupyter Noteboook cho phép bạn export notebook của bạn ra một vài loại file như: PDF, HTML, Python(.py),... Để làm được điều đó, bạn chọn **File -> Download as:**



5. Các package quan trọng sử dụng trong thống kê:



The structure of the most important *Python* packages for statistical applications

numpy: dùng cho các kiểu dữ liệu vector và array

scipy: dùng cho các thuật toán cơ bản trong thống kê

matplotlib: dùng để vẽ các dạng đồ thi

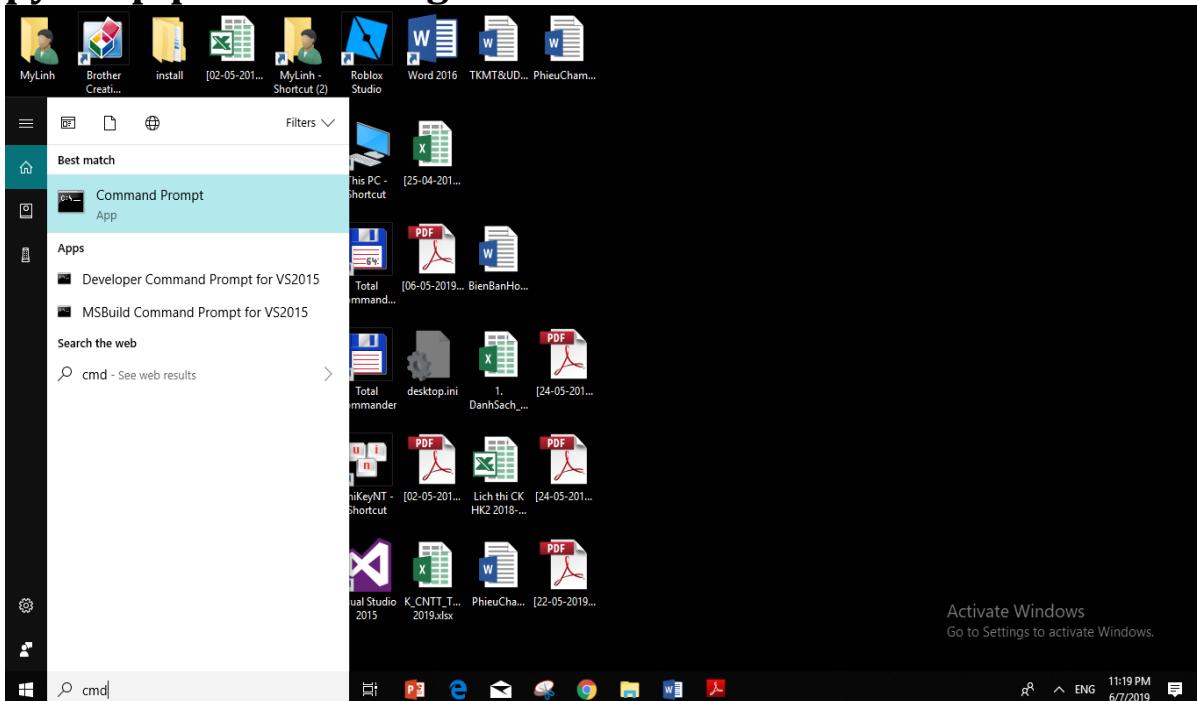
seaborn: dùng để vẽ các dạng đồ thi

pandas: dùng cho các Dataframe (giống 1 bảng gồm các dòng và các cột)

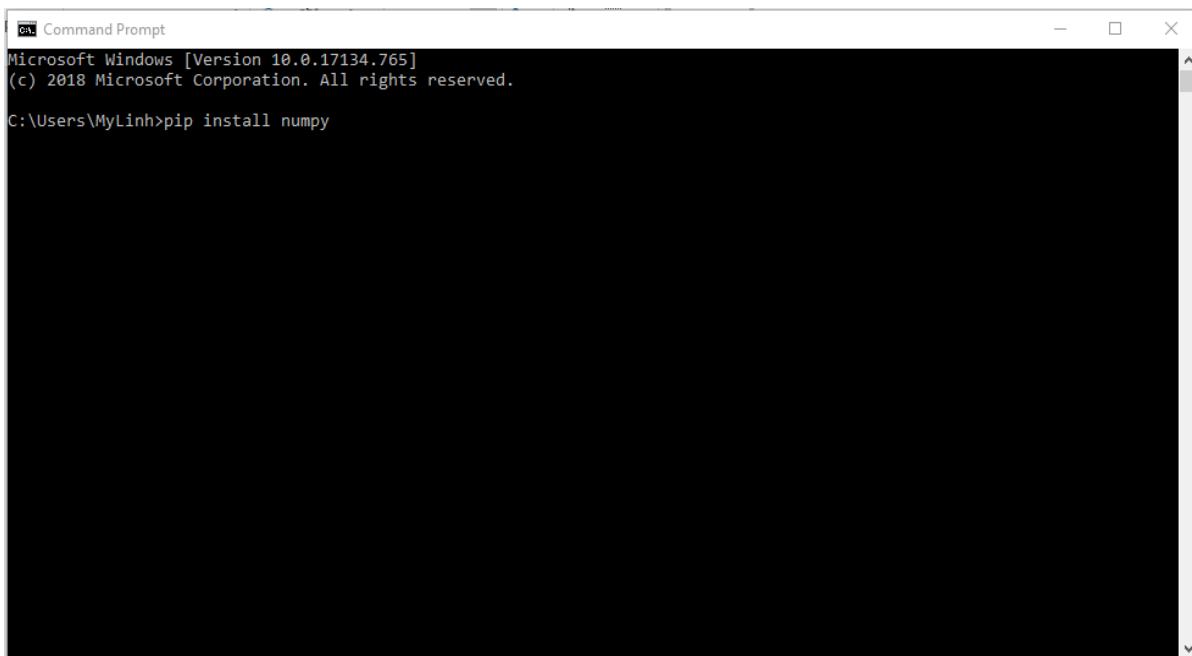
statsmodels: dùng để mô hình hóa thống kê và phân tích nâng cao ví dụ như phân tích hồi quy và phân tích phương sai.

Hướng dẫn cài đặt các package này: vào Command Prompt của Window gõ lệnh: **pip install <tên gói>**

Nếu chương trình không nhận biết được lệnh trên thì gõ lệnh **py -m pip install <tên gói>**



Ví dụ: pip install numpy



6. Bài tập:

- **Kiểu dữ liệu: Tuple, List, Array và DataFrames**

Tuple(): một tập hợp các kiểu dữ liệu khác nhau, không thể sửa đổi khi đã tạo.

Ví dụ:

```
In [1]: import numpy as np
```

```
In [2]: myTuple = ('abc', np.arange(0,3,0.2), 2.5)
```

```
In [3]: myTuple[2]
```

```
Out[3]: 2.5
```

List[]: các phần tử trong list có thể được cập nhật. Vì vậy, list thường được sử dụng cho các item cùng kiểu dữ liệu chẳng hạn kiểu dữ liệu số, chuỗi,...Chú ý: phép cộng list là “+”

Ví dụ:

```
In [4]: myList = ['abc', 'def', 'ghij']
```

```
In [5]: myList.append('klm')
```

```
In [6]: myList
```

```
Out[6]: ['abc', 'def', 'ghij', 'klm']
```

```
In [7]: myList2 = [1,2,3]
```

```
In [8]: myList3 = [4,5,6]
```

```
In [9]: myList2 + myList3
```

```
Out[9]: [1, 2, 3, 4, 5, 6]
```

Array []: vectors và matrices, dùng để thao tác với kiểu dữ liệu dạng số, được định nghĩa trong package numpy. Phép toán ‘+’, ‘.dot’ dùng để cộng, nhân các phần tử trong mảng lại với nhau.

Ví dụ:

```
In [10]: myArray2 = np.array(myList2)
```

```
In [11]: myArray3 = np.array(myList3)
```

```
In [12]: myArray2 + myArray3
```

```
Out[12]: array([5, 7, 9])
```

```
In [13]: myArray2.dot(myArray3)
```

```
Out[13]: 32
```

DataFrame: cấu trúc dữ liệu sử dụng cho dữ liệu thống kê, được định nghĩa trong package **pandas**.

DataFrame là cấu trúc dữ liệu 2 chiều, có gắn nhãn với các cột có thể giống hoặc khác kiểu dữ liệu, giống như một bảng dữ liệu gồm các dòng và các cột.

Ví dụ: tạo 1 DataFrame với 3 cột có tên là “Time,” “x,” và “y”:

```
import numpy as np
import pandas as pd

t = np.arange(0,10,0.1)
x = np.sin(t)
y = np.cos(t)

df = pd.DataFrame({'Time':t, 'x':x, 'y':y})
```

Trong pandas, các dòng được xử lý thông qua các chỉ số và cột thông qua tên của chúng.

Để lấy dữ liệu cột tên “Time”, bạn có hai cách sau:

```
df.Time  
df['Time']
```

Nếu bạn muốn lấy dữ liệu hai cột cùng một lúc, bạn thực hiện như sau:

```
data = df[['Time', 'y']]
```

Để hiển thị dòng 5 dòng đầu tiên hoặc 5 dòng cuối cùng của DataFrame, sử dụng:

```
data.head()  
data.tail()
```

Để lấy dữ liệu từ dòng 5 đến dòng 10, sử dụng:

```
data[4:10]
```

Để lấy dữ liệu đồng thời 2 cột “Time” và “y”, dòng 5 đến dòng 10, sử dụng:

```
df[['Time', 'y']][4:10]
```

Hoặc có thể sử dụng:

```
df.iloc[4:10, [0, 2]]
```

➤ **Đọc dữ liệu từ file text vào DataFrame:**

Bạn có thể dễ dàng đọc vào một file .csv bằng cách sử dụng hàm **read_csv** và được trả về 1 dataframe.

Bạn cũng có thể dùng hàm **read_csv** để đọc **1 file text** và cũng được trả về 1 dataframe.

Tuy nhiên, bạn cũng sẽ phải lưu ý một vài tham số của hàm **read_csv** như:

- encoding: chỉ định encoding của file đọc vào. Mặc định là utf-8.
- sep: thay đổi dấu ngăn cách giữa các cột. Mặc định là dấu phẩy (’,’)
- header: chỉ định file đọc vào có header (tiêu đề của các cột) hay không. Mặc định là infer.
- index_col: chỉ định chỉ số cột nào là cột chỉ số(số thứ tự). Mặc định là None.
- n_rows: chỉ định số bản ghi sẽ đọc vào. Mặc định là None – đọc toàn bộ.

Ví dụ:

Đọc dữ liệu từ file **babies.txt** vào **DataFrame**:

```
In [8]: import pandas as pd
import numpy as np
```

```
In [9]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\\babies.txt', sep='\s+')
print(data)
```

	bwt	smoke
0	120	0
1	113	0
2	128	1
3	123	0
4	108	1
5	136	0
6	138	0
7	132	0
8	120	0
9	143	1
10	140	0
11	144	1
12	141	1
13	110	1
14	114	0
15	115	0
16	92	1
17	115	1
18	144	0
19	119	1
20	105	0

Tạo 1 DataFrame tên là **df_data** gồm tất cả các dòng dữ liệu, các **cột được đặt tên là: bwt và smoke** (nếu file dữ liệu đã có header thì lệnh trên sẽ đặt lại tên header)

```
In [10]: import pandas as pd
import numpy as np
```

```
In [11]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\\babies.txt', sep='\s+')
df_data=pd.DataFrame(data,columns=['bwt','smoke'])
print(df_data)
```

	bwt	smoke
0	120	0
1	113	0
2	128	1
3	123	0
4	108	1
5	136	0
6	138	0
7	132	0
8	120	0
9	143	1
10	140	0
11	144	1
12	141	1
13	110	1
14	114	0
15	115	0
16	92	1
17	115	1
18	144	0
19	119	1

Tạo 1 DataFrame tên là **df_cohutthuoc** gồm các dòng dữ liệu có cột **smoke=1**

```
In [12]: import pandas as pd
import numpy as np
```

```
In [13]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\babies.txt', sep='\s+')
df_cohutthuoc= pd.DataFrame(data,columns=['bwt','smoke']).query('smoke==1')
print(df_cohutthuoc)
```

	bwt	smoke
2	128	1
4	108	1
9	143	1
11	144	1
12	141	1
13	110	1
16	92	1
17	115	1
19	119	1
21	115	1
25	103	1
27	114	1
29	114	1
37	134	1
38	122	1
42	138	1
44	87	1
45	143	1
49	145	1
51	108	1
56	121	1

Tạo 1 DataFrame tên là **df_khonghutthuoc** gồm các dòng dữ liệu có cột **smoke=0**

```
In [14]: import pandas as pd
import numpy as np
```

```
In [15]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\babies.txt', sep='\s+')
df_khonghutthuoc= pd.DataFrame(data,columns=['bwt','smoke']).query('smoke==0')
print(df_khonghutthuoc)
```

	bwt	smoke
0	120	0
1	113	0
3	123	0
5	136	0
6	138	0
7	132	0
8	120	0
10	140	0
14	114	0
15	115	0
18	144	0
20	105	0
22	137	0
23	122	0
24	131	0
26	146	0
28	125	0
30	122	0
31	93	0
32	130	0
33	110	0

Tạo một mảng tên là **arr_cohutthuoc** lấy dữ liệu từ cột **bwt** của DataFrame **df_cohutthuoc**

```
In [16]: import pandas as pd
import numpy as np
```

```
In [17]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\babies.txt', sep='\s+')
df_cohutthuoc= pd.DataFrame(data,columns=['bwt','smoke']).query('smoke==1')
arr_cohutthuoc=np.array(df_cohutthuoc["bwt"])
print(arr_cohutthuoc)
```

```
[128 108 143 144 141 110 92 115 119 115 103 114 114 134 122 138 87 143
 145 108 124 122 101 128 104 137 103 133 91 153 99 114 129 125 114 85
 87 120 107 119 103 91 95 141 100 115 94 101 112 128 93 100 105 160
 113 129 118 133 116 113 131 121 122 101 113 131 96 142 75 125 104 118
 98 150 119 101 113 97 115 121 117 110 130 140 111 154 122 144 114 111
 154 150 99 117 130 81 124 125 115 104 119 123 141 129 119 109 104 110
 98 136 121 91 85 106 109 98 101 71 124 93 101 100 104 117 117 117
 109 120 103 123 104 122 116 129 133 122 133 130 106 121 140 120 127 71
 107 129 145 102 129 135 104 126 127 98 131 99 115 102 143 87 130 123
 116 144 120 116 112 132 146 119 100 118 129 122 117 144 115 99 68 102
 109 102 99 128 101 109 117 88 95 119 127 107 126 98 96 93 101 130
 125 132 69 114 123 129 114 119 119 131 114 110 103 117 126 132 101 114
 108 123 113 93 130 111 97 107 105 133 161 115 127 128 117 119 91 116
 126 98 103 117 115 118 144 85 130 117 135 115 123 154 110 104 123 129
 108 103 127 107 106 152 136 123 93 109 120 129 125 96 138 114 127 113
 99 97 126 119 117 131 118 109 131 134 128 86 115 141 100 114 120 95
 90 131 103 144 86 136 92 129 105 125 115 111 143 116 110 87 132 105
 123 104 96 83 86 110 84 123 96 117 125 114 96 107 98 107 144 136
 125 100 88 77 93 109 145 92 111 134 100 134 145 126 119 103 116 102
 121 100 116 86 87 88 96 112 115 122 127 90 58 109 142 115 163 77
 124 102 94 112 119 97 115 144 99 106 121 123 139 105 146 122 138 120]
```

Tạo một mảng tên là **arr_khonghutthuoc** lấy dữ liệu từ cột **bwt** của DataFrame **df_khonghutthuoc**

```
In [18]: import pandas as pd
import numpy as np
```

```
In [19]: data=pd.read_csv('E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\babies.txt', sep='\s+')
df_khonghutthuoc= pd.DataFrame(data,columns=['bwt','smoke']).query('smoke==0')
arr_khonghutthuoc=np.array(df_khonghutthuoc["bwt"])
print(arr_khonghutthuoc)
```

```
[120 113 123 136 138 132 128 140 114 115 144 105 137 122 131 146 125 122
 93 130 119 113 134 107 128 129 110 111 155 110 122 115 102 143 146 124
145 106 75 107 124 97 142 130 156 120 127 121 120 149 129 139 138 138
131 128 134 114 92 135 125 128 105 119 116 133 155 126 129 137 125 134
118 131 121 131 118 152 121 117 112 109 132 117 128 117 134 127 122 147
120 144 136 102 126 126 115 127 119 123 105 134 144 111 125 135 134 129
121 138 136 120 134 112 132 136 96 124 113 137 133 107 136 130 90 123
137 101 142 124 151 109 131 127 117 150 85 128 105 107 119 134 117 115
 93 125 93 129 126 85 173 111 126 122 141 142 113 149 128 125 114 116
125 110 138 142 102 140 133 127 152 143 131 113 131 148 137 117 115 132
119 132 80 111 143 136 110 108 106 149 135 110 121 142 104 138 112 131
116 140 120 139 131 111 110 105 93 104 120 118 114 116 129 107 88 122
106 135 107 126 116 124 123 98 110 101 96 100 154 127 126 127 129 132
127 145 136 121 121 120 118 127 132 102 118 102 163 132 116 138 139 132
131 115 119 125 123 120 140 120 146 122 128 135 116 129 116 138 123 113
132 120 114 130 142 127 85 123 112 78 107 136 100 123 124 104 133 118
140 115 130 114 105 101 112 115 98 128 154 127 129 138 124 111 103 158
146 132 71 116 129 134 123 147 121 125 115 101 109 115 123 122 124 129
124 142 129 174 103 124 105 108 153 133 123 141 116 121 111 102 118
131 115 147 123 125 99 116 170 104 108 99 97 109 147 105 105 119 103
117 120 145 124 91 109 79 133 114 128 129 97 176 143 113 150 124 119]
```

➤ Đọc dữ liệu từ file excel vào DataFrame:

Để đọc dữ liệu từ file excel vào DataFrame, dùng hàm read_excel

Ví dụ:

```
In [3]: import pandas as pd
import numpy as np
data = pd.read_excel(r'E:\ThongKeMayTinh_UngDung\BaiTap_ThucHanh\DataSet\18_M&M.xls')
print(data)
```

	Red	Orange	Yellow	Brown	Blue	Green
0	0.751	0.735	0.883	0.696	0.881	0.925
1	0.841	0.895	0.769	0.876	0.863	0.914
2	0.856	0.865	0.859	0.855	0.775	0.881
3	0.799	0.864	0.784	0.806	0.854	0.865
4	0.966	0.852	0.824	0.840	0.810	0.865
5	0.859	0.866	0.858	0.868	0.858	1.015
6	0.857	0.859	0.848	0.859	0.818	0.876
7	0.942	0.838	0.851	0.982	0.868	0.809
8	0.873	0.863	NaN	NaN	0.803	0.865
9	0.809	0.888	NaN	NaN	0.932	0.848
10	0.890	0.925	NaN	NaN	0.842	0.940
11	0.878	0.793	NaN	NaN	0.832	0.833
12	0.905	0.977	NaN	NaN	0.807	0.845
13	NaN	0.850	NaN	NaN	0.841	0.852
14	NaN	0.830	NaN	NaN	0.932	0.778
15	NaN	0.856	NaN	NaN	0.833	0.814
16	NaN	0.842	NaN	NaN	0.881	0.791
17	NaN	0.778	NaN	NaN	0.818	0.810
18	NaN	0.786	NaN	NaN	0.864	0.881
19	NaN	0.853	NaN	NaN	0.825	NaN
20	NaN	0.864	NaN	NaN	0.855	NaN
21	NaN	0.877	NaN	NaN	0.842	NaN

LAB 2: THỐNG KÊ MÔ TẢ

Nội dung:

1. Xây dựng histogram
2. Xây dựng scatterplot
3. Xây dựng bar char và pie char
4. Tính các giá trị thống kê: trung bình (mean), trung vị (median), range (min, max), phương sai (varian), độ lệch chuẩn (standard deviation)
5. Xây dựng box plot
6. Kiểm tra dạng chuẩn

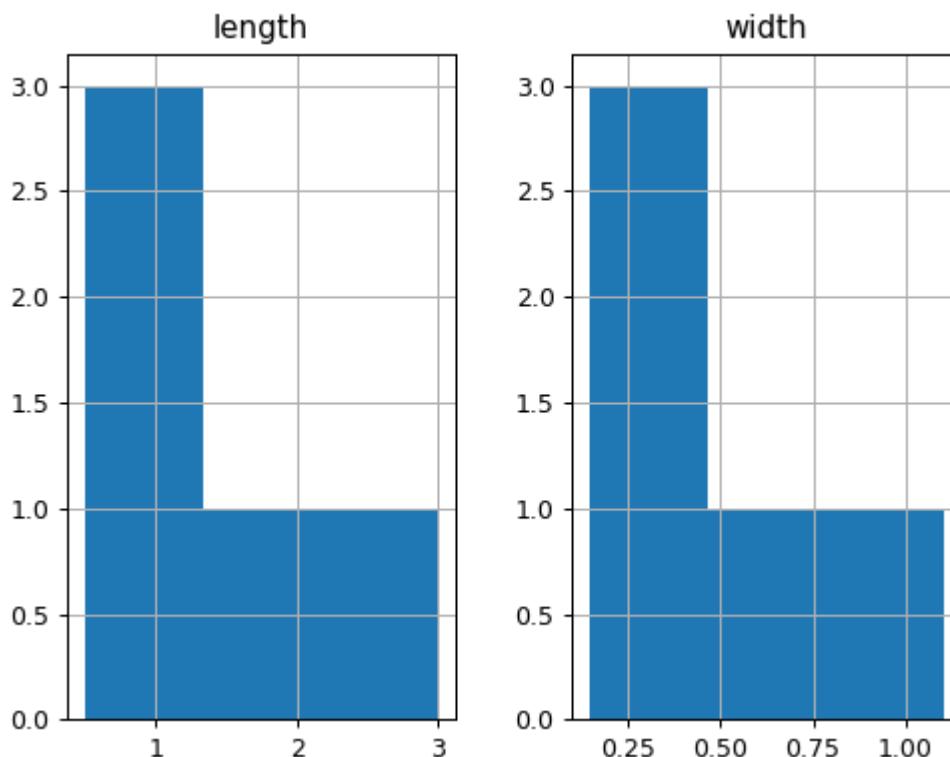
1. Xây dựng histogram:

Hướng dẫn:

Cách 1: Dùng DataFrame của package Pandas

Ví dụ:

```
>>> df = pd.DataFrame({
...     'length': [1.5, 0.5, 1.2, 0.9, 3],
...     'width': [0.7, 0.2, 0.15, 0.2, 1.1]
... }, index= ['pig', 'rabbit', 'duck', 'chicken', 'horse'])
>>> hist = df.hist(bins=3)
```



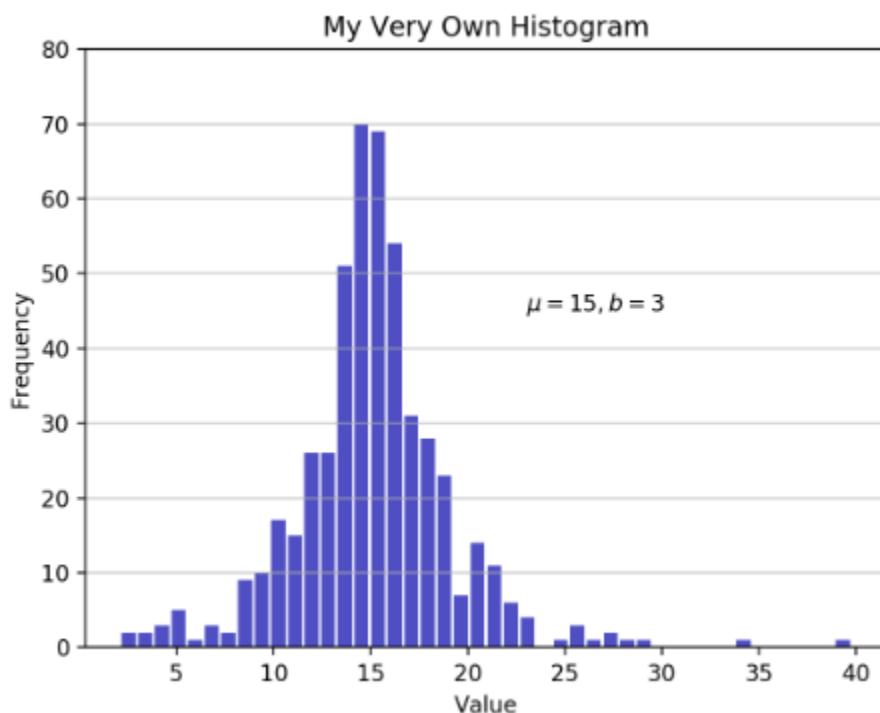
Cách 2: dùng hàm **matplotlib.pyplot.hist**

Ví dụ:

Python

```
import matplotlib.pyplot as plt

# An "interface" to matplotlib.axes.Axes.hist() method
n, bins, patches = plt.hist(x=d, bins='auto', color='#0504aa',
                             alpha=0.7, rwidth=0.85)
plt.grid(axis='y', alpha=0.75)
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.title('My Very Own Histogram')
plt.text(23, 45, r'$\mu=15, b=3$')
maxfreq = n.max()
# Set a clean upper y-axis limit.
plt.ylim(ymax=np.ceil(maxfreq / 10) * 10 if maxfreq % 10 else maxfreq + 10)
```

**Xây dựng histogram cho các bài tập sau:**

- Old Faithful: biểu diễn thời gian (tính bằng giây) phun trào Old Faithful từ Dataset 15.
- Chiều cao của phụ nữ: biểu diễn chiều cao của phụ nữ từ Dataset 1
- Trọng lượng của Diet Coke: biểu diễn trọng lượng (tính bằng pound) của Diet Coke từ Dataset 17.

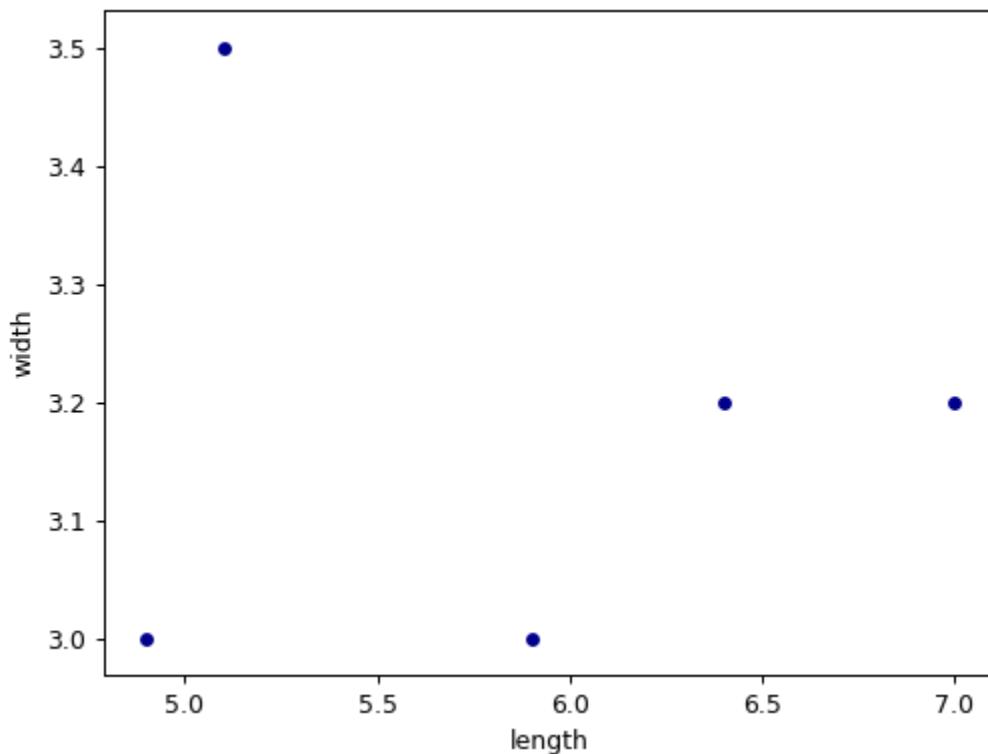
2. Xây dựng Scatterplot:

Hướng dẫn:

Cách 1: Dùng DataFrame của package Pandas

Ví dụ:

```
>>> df = pd.DataFrame([[5.1, 3.5, 0], [4.9, 3.0, 0], [7.0, 3.2, 1],
...                   [6.4, 3.2, 1], [5.9, 3.0, 2]],
...                   columns=['length', 'width', 'species'])
>>> ax1 = df.plot.scatter(x='length',
...                        y='width',
...                        c='DarkBlue')
```



Cách 2: Dùng hàm **matplotlib.pyplot.scatter**

Ví dụ:

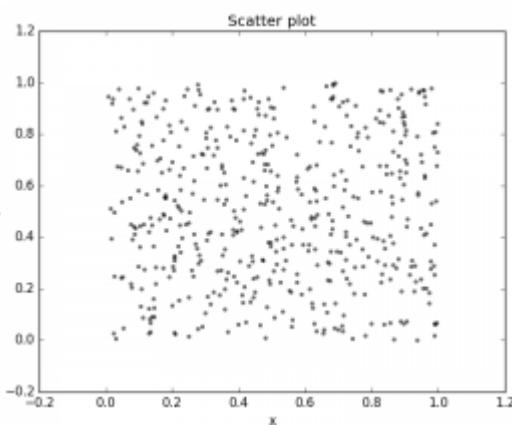
```

import numpy as np
import matplotlib.pyplot as plt

# Create data
N = 500
x = np.random.rand(N)
y = np.random.rand(N)
colors = (0,0,0)
area = np.pi*3

# Plot
plt.scatter(x, y, s=area, c=colors, alpha=0.5)
plt.title('Scatter plot pythonspot.com')
plt.xlabel('x')
plt.ylabel('y')
plt.show()

```



Xây dựng scatter plot cho các bài tập sau:

- **Nhựa/CO trong thuốc lá:** trong Dataset 4, biểu diễn **thuộc tính nhựa** trong thuốc lá cỡ king trên trục X và sử dụng carbon monoxide (**CO**) trong cùng loại thuốc lá cỡ king trên trục Y. Xác định mối quan hệ giữa **nhựa thuốc lá** và **CO** trong thuốc lá cỡ king.
- **Tiêu thụ năng lượng và nhiệt độ:** trong Dataset 12, sử dụng 22 giá trị nhiệt độ trung bình hàng ngày và sử dụng 22 giá trị lượng tiêu thụ năng lượng tương ứng (kWh). (Sử dụng nhiệt độ biểu diễn theo trục X). Dựa trên kết quả, có mối quan hệ giữa nhiệt độ trung bình hàng ngày và lượng năng lượng tiêu thụ hay không?

3. Xây dựng Bar char và Pie char:

Bar chart

Hướng dẫn:

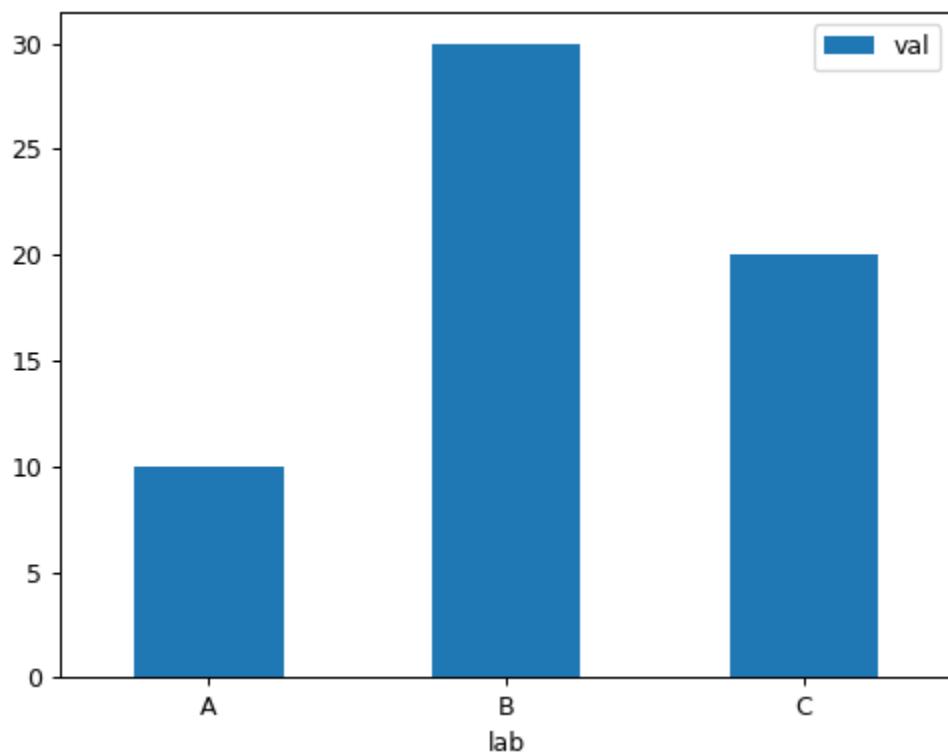
Cách 1: Dùng DataFrame của package Pandas

Ví dụ:

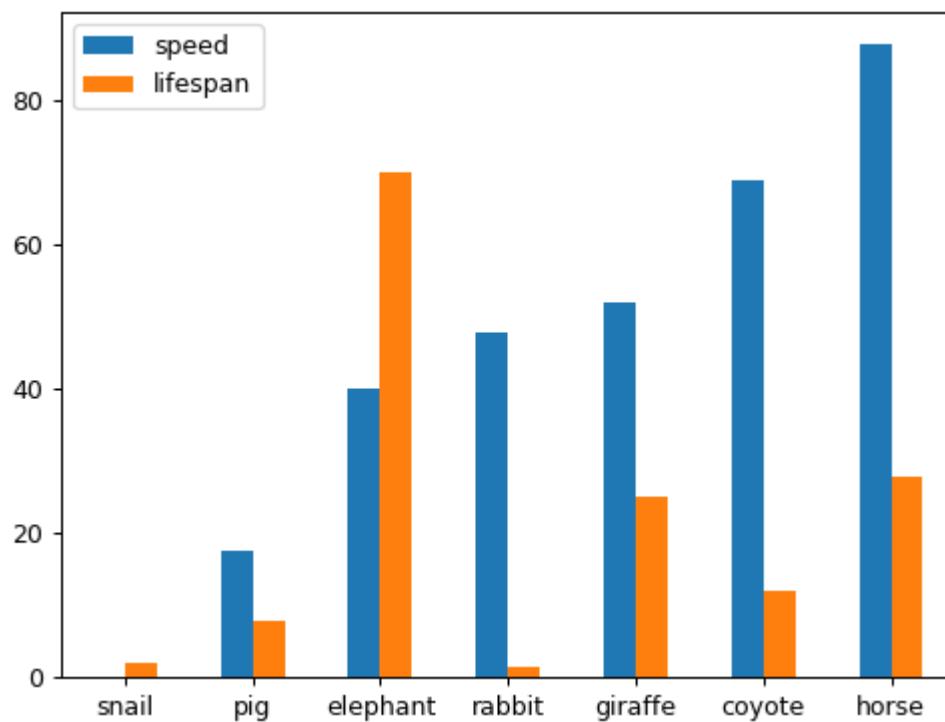
```

>>> df = pd.DataFrame({'lab':['A', 'B', 'C'], 'val':[10, 30, 20]})
>>> ax = df.plot.bar(x='lab', y='val', rot=0)

```



```
>>> speed = [0.1, 17.5, 40, 48, 52, 69, 88]
>>> lifespan = [2, 8, 70, 1.5, 25, 12, 28]
>>> index = ['snail', 'pig', 'elephant',
...           'rabbit', 'giraffe', 'coyote', 'horse']
>>> df = pd.DataFrame({'speed': speed,
...                      'lifespan': lifespan}, index=index)
>>> ax = df.plot.bar(rot=0)
```

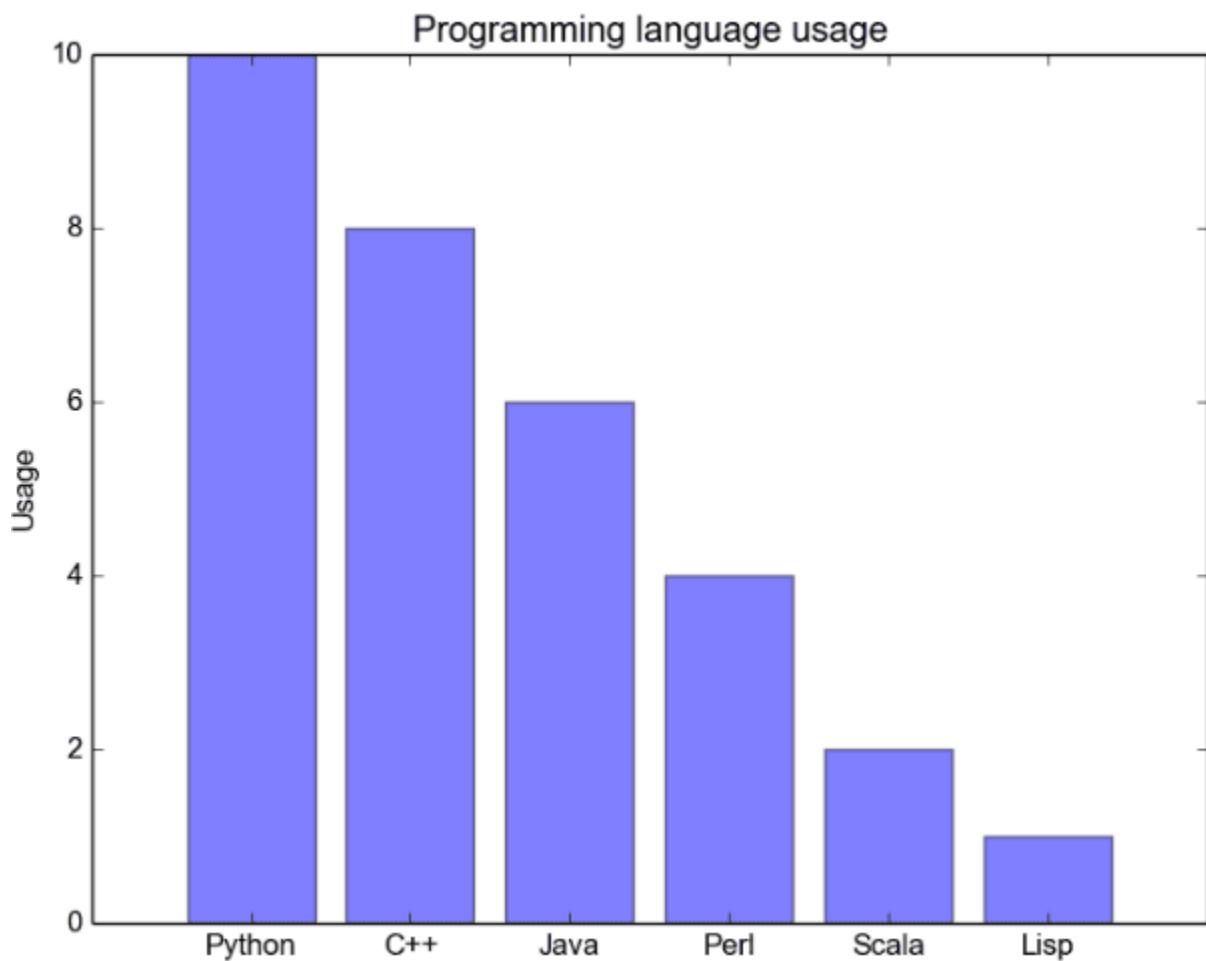
**Cách 2:** Dùng matplotlib.pyplot.bar**Ví dụ:**

```
import matplotlib.pyplot as plt; plt.rcParams()
import numpy as np
import matplotlib.pyplot as plt

objects = ('Python', 'C++', 'Java', 'Perl', 'Scala', 'Lisp')
y_pos = np.arange(len(objects))
performance = [10,8,6,4,2,1]

plt.bar(y_pos, performance, align='center', alpha=0.5)
plt.xticks(y_pos, objects)
plt.ylabel('Usage')
plt.title('Programming language usage')

plt.show()
```



```
import numpy as np
import matplotlib.pyplot as plt

# data to plot
n_groups = 4
means_frank = (90, 55, 40, 65)
means_guido = (85, 62, 54, 20)

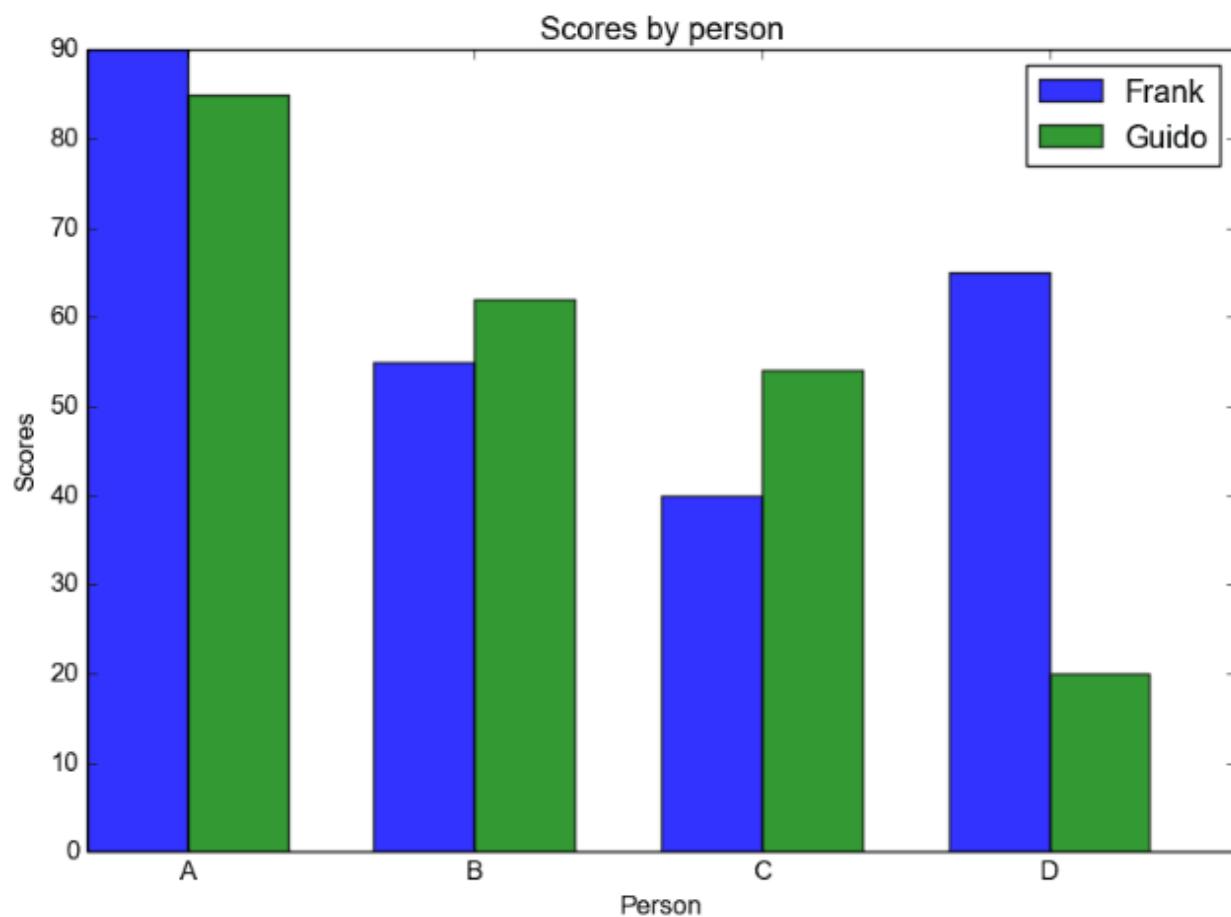
# create plot
fig, ax = plt.subplots()
index = np.arange(n_groups)
bar_width = 0.35
opacity = 0.8

rects1 = plt.bar(index, means_frank, bar_width,
                 alpha=opacity,
                 color='b',
                 label='Frank')

rects2 = plt.bar(index + bar_width, means_guido, bar_width,
                 alpha=opacity,
                 color='g',
                 label='Guido')

plt.xlabel('Person')
plt.ylabel('Scores')
plt.title('Scores by person')
plt.xticks(index + bar_width, ('A', 'B', 'C', 'D'))
plt.legend()

plt.tight_layout()
plt.show()
```



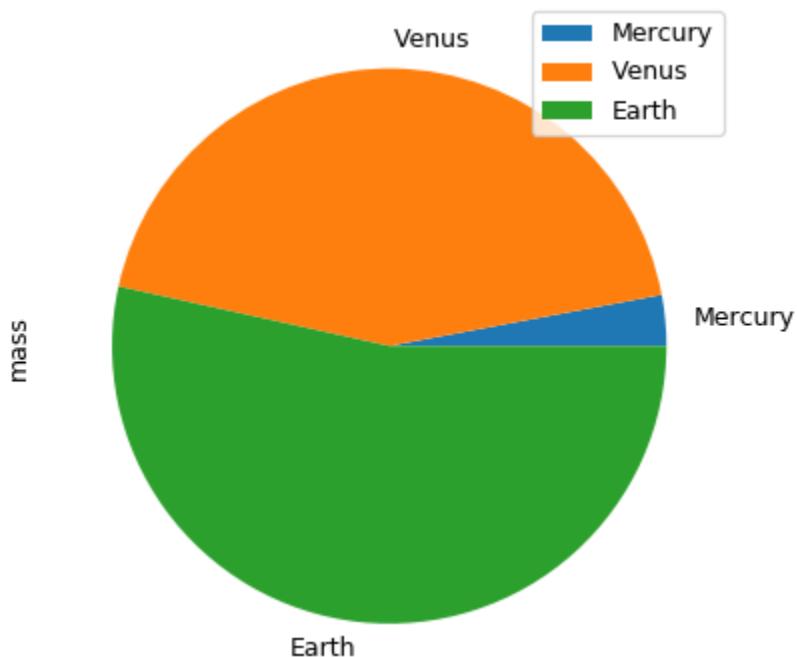
Pie chart

Hướng dẫn:

Cách 1: Dùng DataFrame của package Pandas

Ví dụ:

```
>>> df = pd.DataFrame({'mass': [0.330, 4.87, 5.97],
...                     'radius': [2439.7, 6051.8, 6378.1]},
...                     index=['Mercury', 'Venus', 'Earth'])
>>> plot = df.plot.pie(y='mass', figsize=(5, 5))
```



Cách 2: dùng matplotlib.pyplot.pie

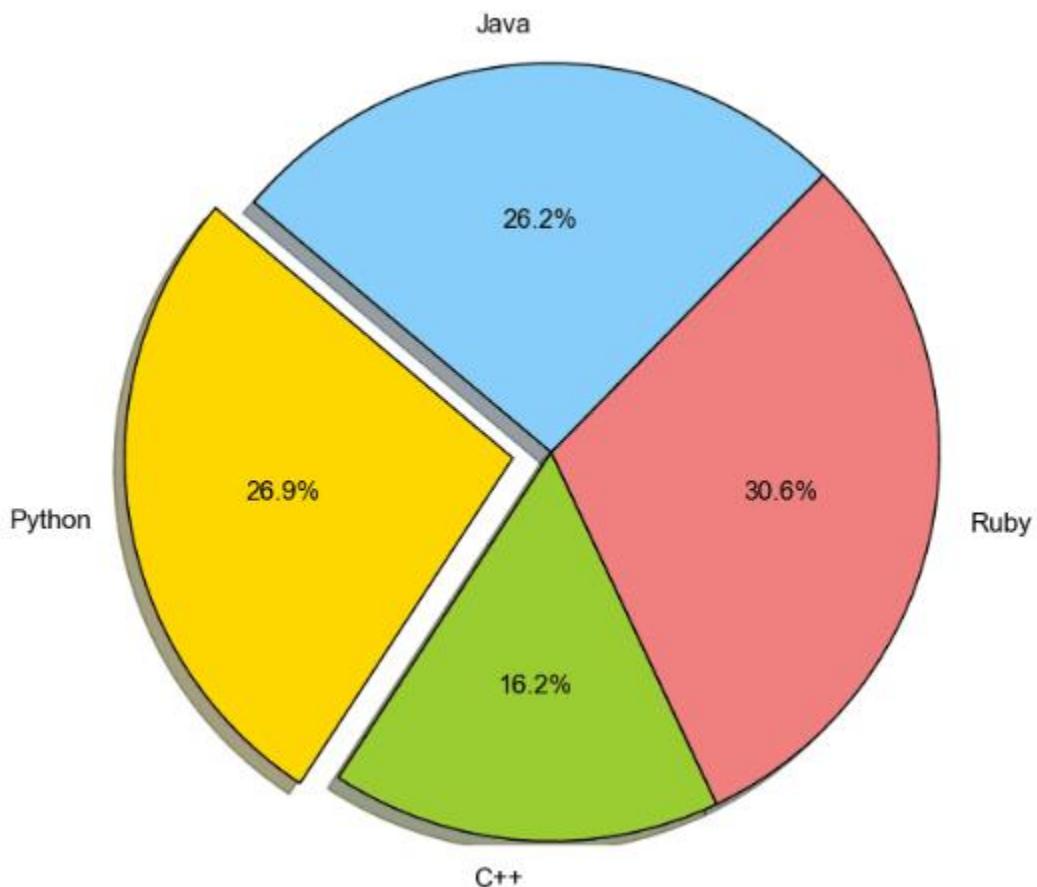
Ví dụ:

```
import matplotlib.pyplot as plt

# Data to plot
labels = 'Python', 'C++', 'Ruby', 'Java'
sizes = [215, 130, 245, 210]
colors = ['gold', 'yellowgreen', 'lightcoral', 'lightskyblue']
explode = (0.1, 0, 0, 0) # explode 1st slice

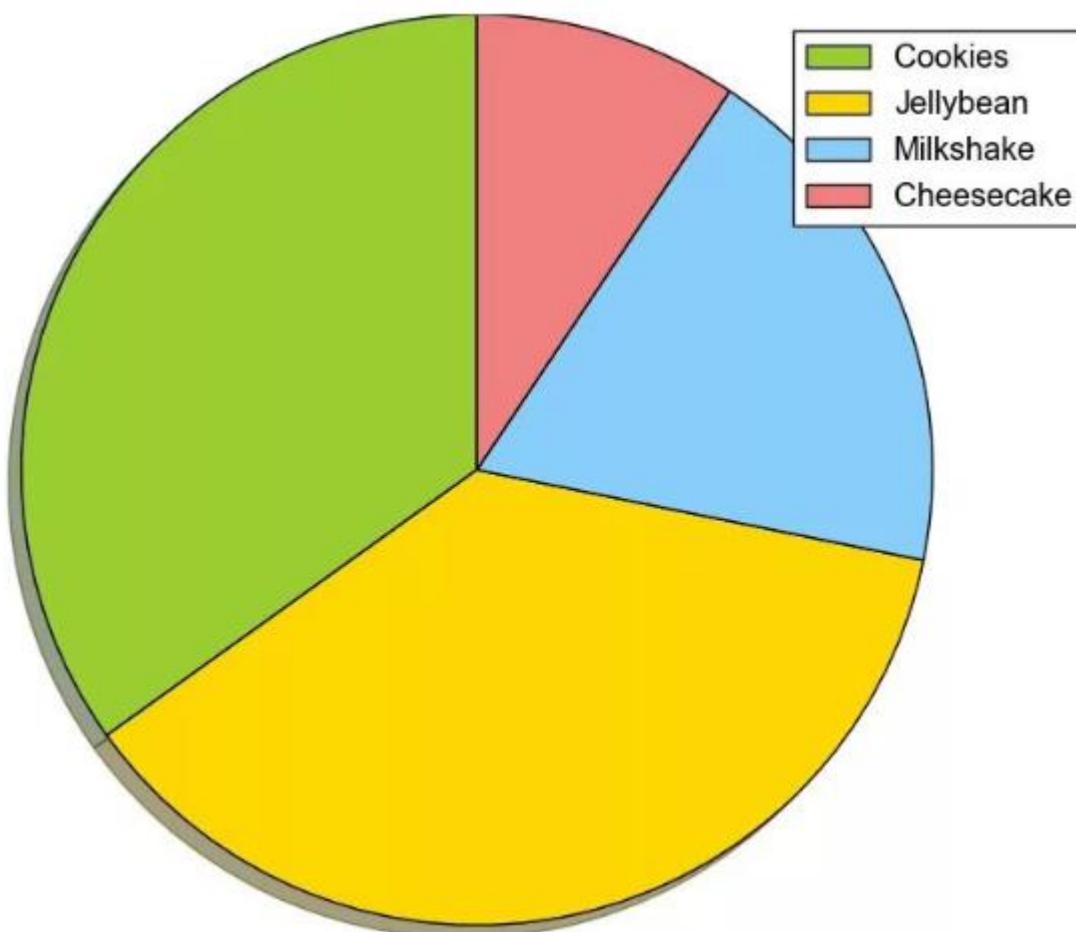
# Plot
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
autopct='%1.1f%%', shadow=True, startangle=140)

plt.axis('equal')
plt.show()
```



```
import matplotlib.pyplot as plt

labels = ['Cookies', 'Jellybean', 'Milkshake', 'Cheesecake']
sizes = [38.4, 40.6, 20.7, 10.3]
colors = ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral']
patches, texts = plt.pie(sizes, colors=colors, shadow=True, startangle=90)
plt.legend(patches, labels, loc="best")
plt.axis('equal')
plt.tight_layout()
plt.show()
```



- Xây dựng bar char và pie char cho dữ liệu ở bảng sau. So sánh 2 biểu đồ trên, biểu đồ nào là hiệu quả hơn trong việc hiển thị thông tin.

College	Relative Frequency
Public 2-Year	36.8%
Public 4-Year	40.0%
Private 2-Year	1.6%
Private 4-Year	21.9%

4. Tính các giá trị thống kê: trung bình (mean), trung vị (median), range (min, max), phương sai (varian), độ lệch chuẩn (standard deviation)

Hướng dẫn:

Cách 1: dùng hàm mean(...), median(...), std(...), var(...), max(...), min(...) của DataFrame trong package pandas

Cách 2: dùng hàm mean(...), median(...), std(...), var(...), max(...), min(...) của package numpy

Tính các giá trị thống kê sau: trung bình (mean), trung vị (median):

- **Nhiệt độ cơ thể:** Sử dụng nhiệt độ cơ thể lúc 12:00 AM vào ngày 2 từ Dataset 2. Các kết quả có hỗ trợ hoặc mâu thuẫn với phát biểu “nhiệt độ trung bình của cơ thể là 98,6°F” hay không?
- **Vít máy:** Sử dụng độ dài được liệt kê của các vít máy từ DataSet 19. Các ốc vít được cho là có chiều dài 3/4 in. Kết quả về độ dài quy định có đúng không?
- **Điện áp gia đình:** So sánh mean và median từ 3 tập dữ liệu khác nhau của các mức điện áp đã đo từ Dataset 13.
- **Phim:** Dataset 9. Xét tổng tiền thu được từ hai thể loại phim khác nhau: những phim có xếp hạng R và những phim có xếp hạng PG hoặc PG-13. Các kết quả tính được có hỗ trợ cho phát biểu sau không: “phim có xếp hạng R có tổng tiền thu được lớn hơn vì chúng thu hút khán giả lớn hơn các bộ phim được xếp hạng PG hoặc PG-13”?

Tính các giá trị thống kê sau: range (min, max), phương sai (varian), độ lệch chuẩn (standard deviation):

- **Nhiệt độ cơ thể:** Sử dụng nhiệt độ cơ thể lúc 12:00 AM vào ngày 2 từ Dataset 2.
- **Vít máy:** Sử dụng độ dài được liệt kê của các vít máy từ DataSet 19.
- **Điện áp gia đình:** So sánh phương sai từ 3 tập dữ liệu khác nhau của các mức điện áp đã đo từ Dataset 13
- **Phim:** Dataset 9. Xét tổng tiền thu được từ hai thể loại phim khác nhau: những phim có xếp hạng R và những phim có xếp hạng PG hoặc PG-13. Xác định xem hai loại có giống nhau về phương sai không.

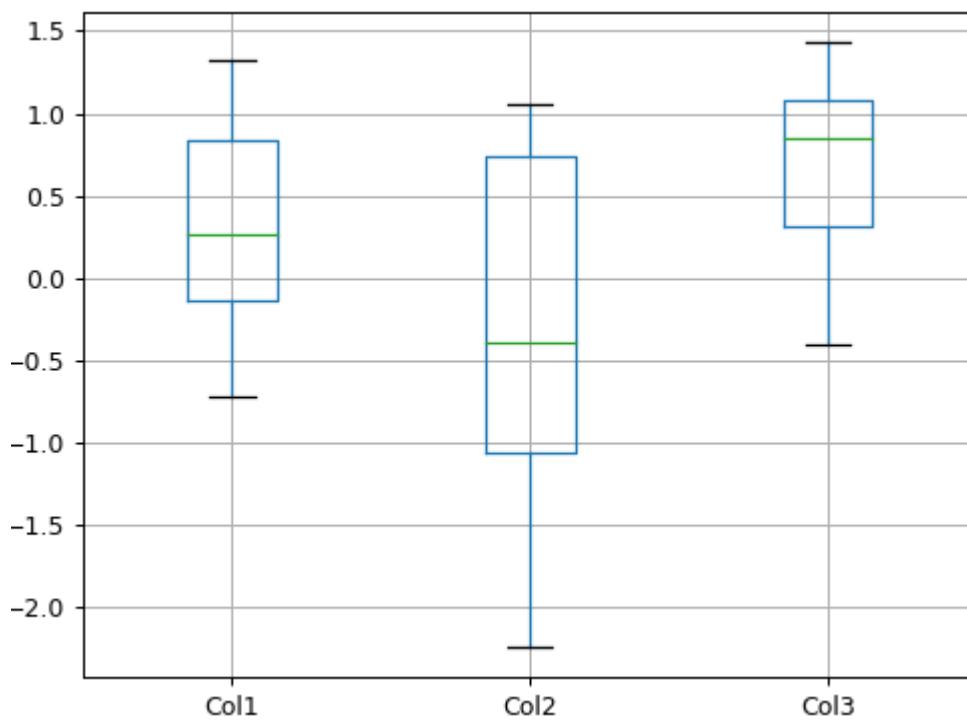
5. Xây dựng box plot

Hướng dẫn:

Cách 1: dùng hàm boxplot trong của DataFrame trong package Pandas.

Ví dụ:

```
>>> np.random.seed(1234)
>>> df = pd.DataFrame(np.random.randn(10,4),
...                     columns=['Col1', 'Col2', 'Col3', 'Col4'])
>>> boxplot = df.boxplot(column=['Col1', 'Col2', 'Col3'])
```

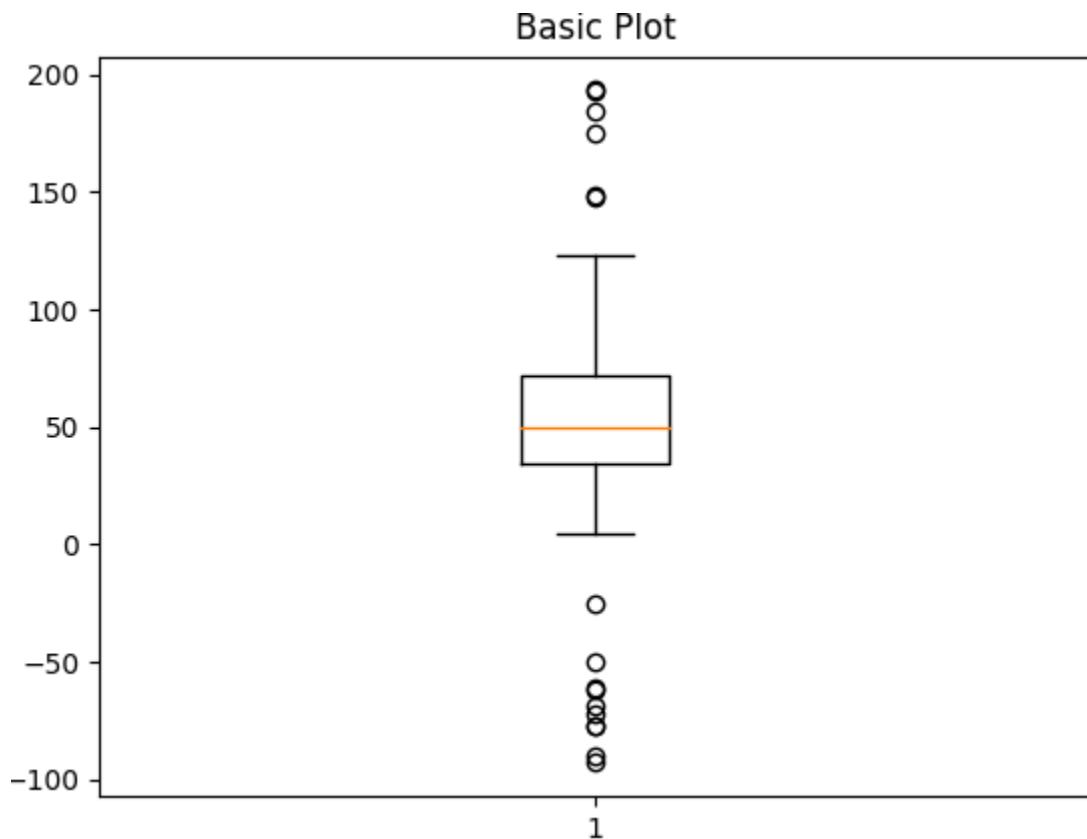
**Cách 2: dùng hàm matplotlib.pyplot.boxplot****Ví dụ:**

```
import numpy as np
import matplotlib.pyplot as plt

# Fixing random state for reproducibility
np.random.seed(19680801)

# fake up some data
spread = np.random.rand(50) * 100
center = np.ones(25) * 50
flier_high = np.random.rand(10) * 100 + 100
flier_low = np.random.rand(10) * -100
data = np.concatenate((spread, center, flier_high, flier_low))
```

```
fig1, ax1 = plt.subplots()
ax1.set_title('Basic Plot')
ax1.boxplot(data)
```



Xây dựng box plot cho các bài tập sau:

- **Trọng lượng của Coke thông thường và Coke ăn kiêng.** Sử dụng cùng một tỷ lệ để xây dựng box plot đối với trọng lượng của Coke thông thường và Coke ăn kiêng từ Dataset 17. Sử dụng box plot để so sánh hai bộ dữ liệu.
- **Trọng lượng của Coke thông thường và Pepsi thông thường.** Sử dụng cùng một tỷ lệ để xây dựng box plot cho trọng lượng của Coke thông thường và Pepsi thông thường từ Dataset 17. Sử dụng box plot để so sánh hai bộ dữ liệu.
- **Trọng lượng của phần tư vị.** Sử dụng cùng một tỷ lệ để xây dựng các box plot cho trọng lượng của các phần tư vị trước năm 1964 và phần tư vị sau năm 1964 từ Dataset 20. Sử dụng box plot để so sánh hai bộ dữ liệu.
- **Số lượng điện áp.** Sử dụng cùng một tỷ lệ để xây dựng các box plot cho lượng điện áp tại nhà và lượng điện áp máy phát từ Dataset 13. Sử dụng box plot để so sánh hai bộ dữ liệu.

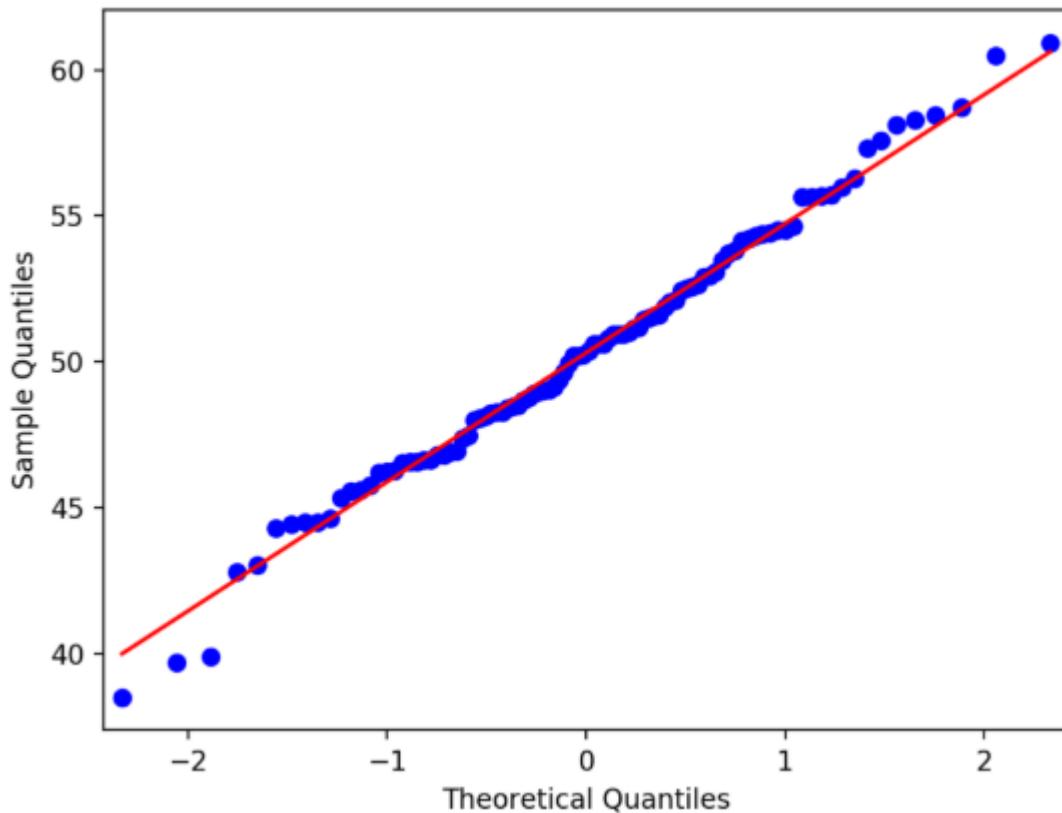
6. Kiểm tra dạng chuẩn

Hướng dẫn:

```

1 # QQ Plot
2 from numpy.random import seed
3 from numpy.random import randn
4 from statsmodels.graphics.gofplots import qqplot
5 from matplotlib import pyplot
6 # seed the random number generator
7 seed(1)
8 # generate univariate observations
9 data = 5 * randn(100) + 50
10 # q-q plot
11 qqplot(data, line='s')
12 pyplot.show()

```



Vẽ QQ-plot trong các bài tập sau, xác định xem dữ liệu mẫu được lấy từ quần thể có phân phối chuẩn có phân phối chuẩn hay không.

- **Old Faithful:** biểu đồ QQ-plot biểu diễn thời gian (tính bằng giây) phun trào Old Faithful từ Dataset 15.
- **Chiều cao của phụ nữ:** biểu đồ QQ-plot biểu diễn chiều cao của phụ nữ từ Dataset 1.
- **Trọng lượng của Coke ăn kiêng:** biểu đồ QQ-plot biểu diễn trọng lượng (tính bằng pound) của Diet Coke từ Dataset 17.

LAB 3: THỐNG KÊ MÔ TẢ

Nội dung:

Trong lab này, chúng ta sẽ:

1. Học cách phân tích dữ liệu thông qua các giá trị tóm tắt dữ liệu và qua biểu diễn hình học của dữ liệu.
2. So sánh hai tập dữ liệu

Dữ liệu: Dữ liệu sử dụng trong lab này là tập dữ liệu về cân nặng của trẻ sơ sinh trong trường hợp bà mẹ hút thuốc lá khi mang thai và trong trường hợp bà mẹ không hút thuốc lá khi mang thai. (Dữ liệu được chuẩn bị sẵn trong tập tin: babies.txt).

Mô tả dữ liệu:

Tên cột	Ý nghĩa
bwt	Cân nặng của trẻ sơ sinh (baby weight), tính theo đơn vị ounce (100 ounce=2.83495kg)
smoke	Tình trạng hút thuốc của bà mẹ khi mang thai. 0=không hút, 1=có hút, 9=không biết

I. CÁC NỘI DUNG CẦN TÌM HIỂU:

Để thực hiện được lab này, sinh viên cần vận dụng các kiến thức ở lab 2 vào bài toán cụ thể:

1) Ước lượng độ biến động của dữ liệu:

Hai yếu tố chính để ước lượng độ biến động của dữ liệu: tâm và đuôi dữ liệu. Qua đó, ta cần tìm hiểu: dữ liệu phân bố như thế nào ở trung tâm (center) và như thế nào ở hai bên đuôi (tail).

Trong dữ liệu một chiều, để đo tính biến động của dữ liệu, ta có thể sử dụng các đại lượng: phương sai (Variance), độ lệch chuẩn (Standard deviation), khoảng cách giữa giá trị lớn nhất và nhỏ nhất (Range) và phần tư vị (IQR-InterQuantile Range). IQR cho phép khảo sát phần tâm dữ liệu trong khoảng từ $\frac{1}{4}$ cho đến $\frac{3}{4}$.

Đôi khi, để dễ hình dung, người phân tích có thể biểu diễn dữ liệu theo boxplot hay histogram, sẽ minh họa sau.

2) Phân tích về hình dạng của phân phối dữ liệu:

Để phân tích hình dạng phân phối dữ liệu, người phân tích cần tính giá trị **KURTOSIS**, là giá trị để đo độ “bè-nhọn” của đỉnh dữ liệu và giá trị **SKEWNESS** để đo độ “lệch (trái, phải)” của dữ liệu.

3) Phân tích tính chuẩn:

Để phân tích xem dữ liệu có phân phối chuẩn hay không, một cách trực quan, ta biểu diễn theo đường cong chuẩn (normal curve) và đôi khi cần một số thao tác chuẩn hóa.

II. CÁC NỘI DUNG THỰC HIỆN:

Trong lab này, ta phân tích các dữ liệu quan sát được để trả lời câu hỏi: “Việc bà mẹ hút thuốc khi mang thai có ảnh hưởng đến cân nặng của trẻ sơ sinh hay không?”

Để trả lời câu hỏi trên, cần thực hiện so sánh cân nặng của trẻ sơ sinh trong hai trường hợp: trường hợp bà mẹ hút thuốc khi mang thai và trường hợp bà mẹ không hút thuốc khi mang thai. Sự khác biệt đó có ý nghĩa hay không?

Để so sánh cân nặng của trẻ sơ sinh trong 2 trường hợp, có thể dựa vào thống kê mô tả: thống kê mô tả bằng số (numerical summaries), thống kê mô tả bằng hình (graphical): histogram, boxplot, quantile plot. Do đó, các nội dung chi tiết cần thực hiện:

1) Tính các đại lượng thống kê mô tả từ đó rút ra nhận xét về từng tập dữ liệu (cân nặng của trẻ trong trường hợp bà mẹ hút thuốc và cân nặng của trẻ trong trường hợp bà mẹ không hút thuốc).

Cụ thể, ta sẽ phân tích sự khác biệt giữa hai tập dữ liệu: cân nặng của trẻ trong trường hợp bà mẹ hút thuốc và cân nặng của trẻ trong trường hợp bà mẹ không hút thuốc dựa vào các đại lượng thống kê mô tả.

2) Biểu diễn dữ liệu dưới các dạng đồ thị từ đó rút ra nhận xét về từng tập dữ liệu (trường hợp bà mẹ hút thuốc và trường hợp bà mẹ không hút thuốc)

Cụ thể, ta sẽ sử dụng các dạng đồ thị: histogram, boxplot, quantile qua đó phân tích sự khác biệt giữa hai tập dữ liệu: cân nặng của trẻ trong trường hợp bà mẹ hút thuốc và cân nặng của trẻ trong trường hợp bà mẹ không hút thuốc dựa vào các đồ thị.

HƯỚNG DẪN THỰC HIỆN:

1. Mô tả dữ liệu bằng các giá trị số:

Bước 1: Tính các đại lượng thống kê cho hai tập dữ liệu:

(Cân nặng của trẻ trong trường hợp bà mẹ hút thuốc khi mang thai và cân nặng của trẻ trong trường hợp bà mẹ không hút thuốc khi mang thai).

Dùng python để thực hiện, kết quả được trình bày trong bảng sau:

	TH1: Bà mẹ hút thuốc	TH2: Bà mẹ không hút thuốc
Số lượng	484	742
Min	58	55
Max	163	176
Mean	114.10950413223141	123.04716981132076
Sd	18.09894568615237	17.39868877808027
Var	327.57183495029346	302.7143711964963
Median	115.0	123.0
Quantile 0%	58.0	55.0
Quanlite 25%	102.0	113.0
Quanlite 50%	115.0	123.0
Quantile 75%	126.0	134.0
IQR	24.0	21.0

Skewness	-0.03359497605204854	-0.18698408606617228
Kurtosis	2.988032478793404	4.037060312433822

Bước 2: Phân tích dữ liệu dựa trên các đại lượng vừa tính.

1. Xét tập dữ liệu ứng với trường hợp bà mẹ có hút thuốc

Vị trí tập trung của dữ liệu: khoảng giá trị: 114-115

Tính biến động của dữ liệu:

- **Phương sai (variance):** var= 327.57183495029346
- **Độ lệch chuẩn (standard deviation):** sd= 18.09894568615237
- **Khoảng giá trị:** min=58, max=163 → range=105
- **Khoảng cách giữa 2 phần tư vị:** IQR=Q3-Q1=126-102=24

Nhận xét: Như vậy dữ liệu phân bố gần nhau.

Hình dạng phân bố của dữ liệu:

- **Độ lệch:** Skewness=-0.03359497605204854
- **Độ bè nhọn của đỉnh dữ liệu:** Kurtosis=2.988032478793404

Nhận xét: Như vậy dữ liệu hơi lệch về phía trái, và đỉnh nhọn, hai bên giảm với tốc độ vừa phải.

2. Xét tập dữ liệu ứng với trường hợp bà mẹ không hút thuốc

Phần này sinh viên tự thực hiện.

Bước 3: So sánh các giá trị thống kê mô tả của hai tập dữ liệu.

Sự khác biệt về vị trí tập trung dữ liệu: chênh lệch khoảng 123 -115 = 8

Nhận xét: khác biệt không đáng kể.

Sự khác biệt về tính biến động của dữ liệu được thể hiện qua bảng sau:

	TH1: Bà mẹ hút thuốc	TH2: Bà mẹ không hút thuốc	Chênh lệch (TH2-TH1)
Sd	18.09894568615237	17.39868877808027	-0.700256908
Var	327.57183495029346	302.7143711964963	-24.85746375
Range	163-58=105	176-55=121	16
IQR	126-102=24	134-113=21	-3

Dữ liệu trong trường hợp bà mẹ không hút thuốc có phân bố rộng hơn nhưng phần dữ liệu tập trung lại hẹp hơn so với trường hợp bà mẹ có hút thuốc. Sự biến động của dữ liệu trong hai trường hợp không khác biệt nhiều.

Sự khác biệt về hình dạng phân bố của dữ liệu: được thể hiện qua bảng sau:

	TH1: Bà mẹ hút thuốc	TH2: Bà mẹ không hút thuốc	Chênh lệch (TH2-TH1)
Skewness	-0.03359497605204854	-0.18698408606617228	-0.15338911
Kurtosis	2.988032478793404	4.037060312433822	1.049027834

- **Nhận xét:** trường hợp bà mẹ hút thuốc có phân bố dữ liệu nhọn hơn, đối xứng hơn so với trường hợp không hút thuốc. Cả 2 trường hợp đều hơi lệch về trái.

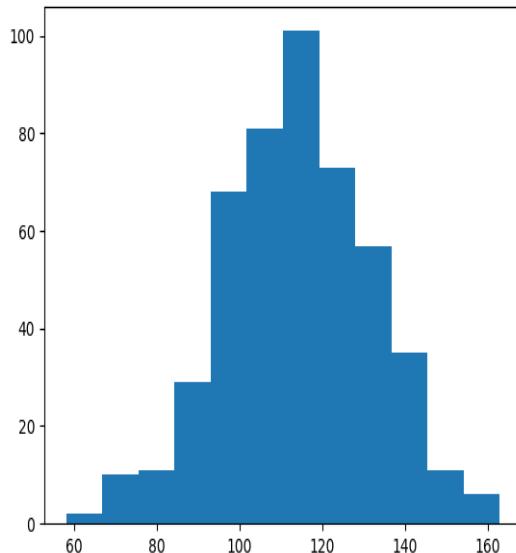
2. Biểu diễn hình học của dữ liệu

1. Dữ liệu cân nặng của trẻ trong trường hợp bà mẹ hút thuốc và bà mẹ không hút thuốc

Ta sẽ phân tích các biểu đồ:

- Histogram
- Boxplot

a) Histogram trong trường hợp bà mẹ có hút thuốc:



Vị trí tập trung dữ liệu: khoảng 110

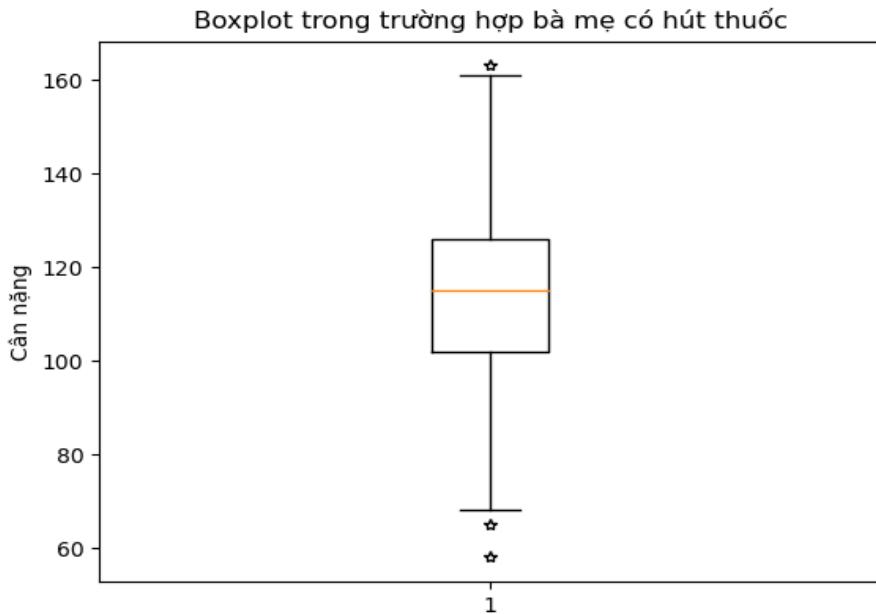
Tính biến động của dữ liệu: dữ liệu phân bố trong khoảng [50-170]

Tính đối xứng của phân bố dữ liệu: dữ liệu chỉ có 1 đỉnh. Bắt đầu từ đỉnh, hai bên giảm dần và tốc độ giảm vừa phải.

Dữ liệu phân bố gần đối xứng, hơi lệch về phía trái. Hai bên đuôi có độ dài vừa phải. Hai bên đỉnh dữ liệu cũng phân bố vừa phải.

Giá trị ngoại lệ: không thấy rõ có giá trị ngoại lệ nào đáng kể

b) Boxplot:



Tính biến động của dữ liệu: dữ liệu phân bố tập trung trong khoảng từ [102,126]
Giá trị ngoại lệ: có một số giá trị ngoại lệ (lớn hơn 162, nhỏ hơn 66) nhưng không nhiều.

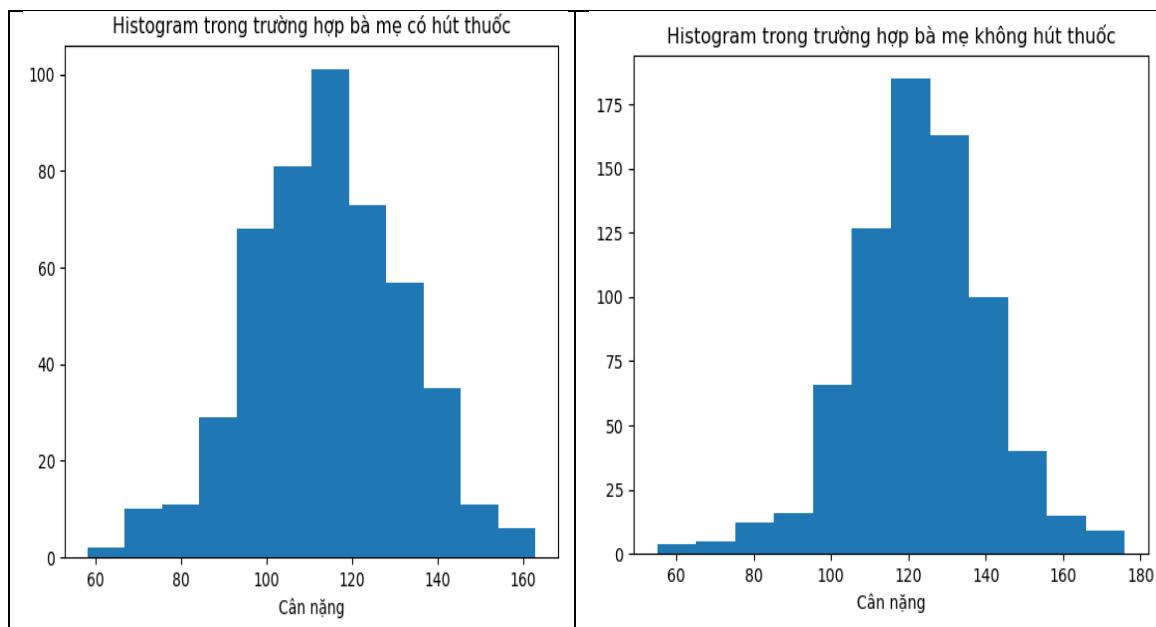
2. Dữ liệu cân nặng của trẻ trong trường hợp bà mẹ hút thuốc không hút thuốc

Phần này sinh viên tự thực hiện

So sánh hai tập dữ liệu dựa vào các biểu diễn hình học:

a) Histogram

Để so sánh, ta vẽ 2 histogram gần nhau:

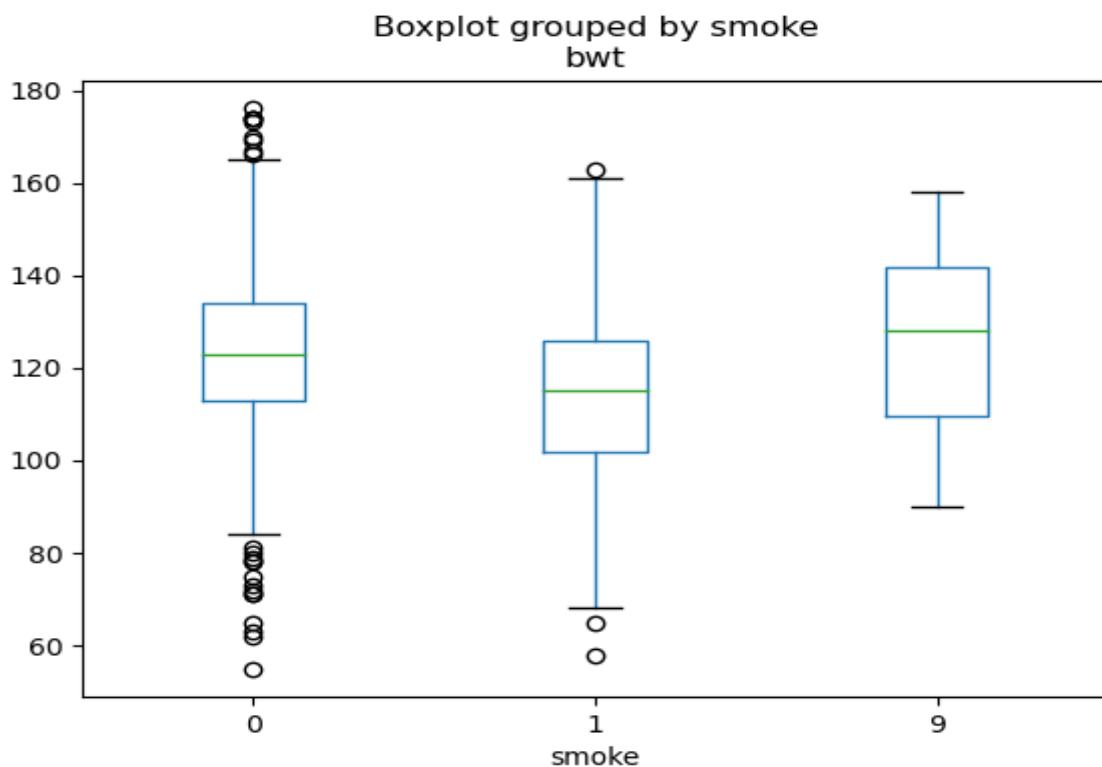


Cân nặng của trẻ trong trường hợp bà mẹ không hút thuốc cáo hơn so với trường hợp bà mẹ có hút thuốc

Tính biến thiên của 2 tập dữ liệu: tương tự nhau

Tính đối xứng của 2 tập dữ liệu: tương tự nhau

Giá trị ngoại lệ: cả 2 đều không có giá trị ngoại lệ đáng chú ý.

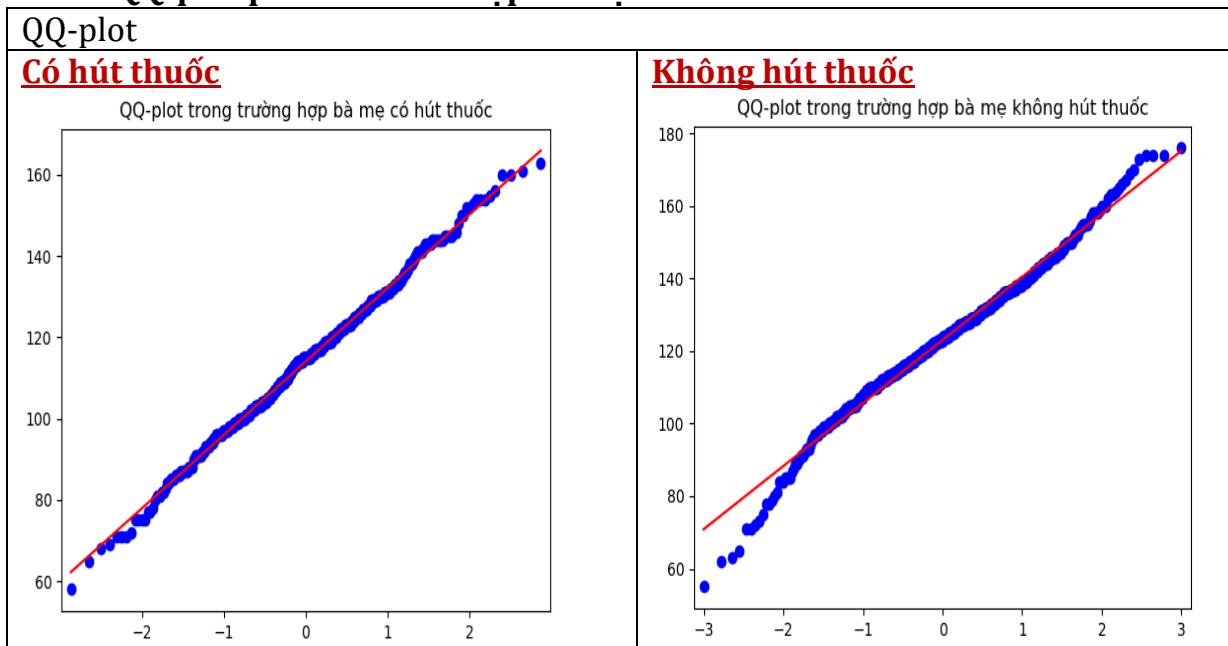
b) Boxplot

Khác biệt về vị trí: giá trị trung vị của trường hợp bà mẹ không hút thuốc lớn hơn trường hợp bà mẹ có hút thuốc (123 và 115). (Trường hợp smoke=9 là trường hợp không biết bà mẹ có hút thuốc hay không, trung vị trong trường hợp này cao hơn so với 2 trường hợp bà mẹ có hút thuốc và không hút thuốc).

Giá trị ngoại lệ: cả 2 trường hợp đều có giá trị ngoại lệ trên và dưới. Trường hợp không hút thuốc có nhiều giá trị ngoại lệ hơn.

Ta dùng thêm đồ thị QQ-plot để phân tích

So sánh QQ-plot phân bố của 2 tập dữ liệu:



QQ-plot có dạng đường thẳng, suy ra dữ liệu của 2 trường hợp có phân bố tương tự nhau.

LAB 4: ƯỚC LƯỢNG

Nội dung:

1. Mô phỏng dữ liệu
2. Ước lượng điểm
3. Ước lượng khoảng
4. Bài tập

1. Mô phỏng dữ liệu:

- Dữ liệu thực tế không phải khi nào cũng có sẵn, để có dữ liệu để thực hiện thống kê đòi hỏi nhiều thời gian và công sức trong việc thu nhập và tiền xử lý dữ liệu. Để thuận lợi cho việc học tập và nghiên cứu, ta có thể tạo một bộ dữ liệu mô phỏng theo mong muốn của mình. Hiện nay, có rất nhiều công cụ cho phép ta mô phỏng dữ liệu, module `<np.random>` là một trong những công cụ hữu ích.
- Một số chức năng mà module `<np.random>` có thể cung cấp:
 - Tạo một số ngẫu nhiên trong khoảng $(0, 1)$:
random()
 - Tạo hạt giống ngẫu nhiên: nhằm mục đích có thể tạo bộ dữ liệu mô phỏng giống lần trước. Giả sử, bạn tạo một bộ dữ liệu để xử lý bằng cách sử dụng hàm `random()`. Sau đó một người khác cũng lặp lại cách làm của bạn, tuy nhiên khi sử dụng hàm `random()` thì được bộ dữ liệu khác với của bạn dẫn đến kết quả xử lý có thể khác nhau nên không thể so sánh được. Để giải quyết vấn đề này, bạn có thể tạo một hạt giống ngẫu nhiên

là một số nguyên bất kỳ trước khi thực hiện mô phỏng dữ liệu, trường hợp tái mô phỏng lại bộ dữ liệu cũ, chỉ cần phát sinh đúng hạt giống ban đầu.

seed()

- Ví dụ: mô phỏng tung đồng xu 4 lần

```
import numpy as np

#Tạo hạt giống ngẫu nhiên là 10
np.random.seed(10)

#In hai số ngẫu nhiên trong khoảng (0, 1)
print(np.random.random())
print(np.random.random())

#Phát sinh 4 số ngẫu nhiên trong khoảng (0, 1)
arr = np.random.random(size=4)
print(arr)

#Tạo mẫu mô phỏng 4 lần tung đồng xu với (True: Sấp, False: Ngửa)
coin_sample = arr < 0.5
print(coin_sample)
```

0.771320643266746
0.0207519493594015
[0.63364823 0.74880388 0.49850701 0.22479665]
[False False True True]

2. Ước lượng điểm:

- **VD1:** Sử dụng ước lượng điểm để ước lượng tham số của quần thể

```
#Khởi tạo một quần thể cho trước để hiện chiều cao(cm) của 5 thanh
SMALL_POP = np.array([186, 182, 157, 158, 152])
print(SMALL_POP)
mean_of_SMALL_POP = np.mean(SMALL_POP)
print('Chiều cao trung bình của Quần Thể : {}'.format(mean_of_SMALL_POP))

#Lấy ngẫu nhiên một mẫu có kích thước là 4, và tính chiều cao trung bình và so sánh với giá trị của quần thể
np.random.seed(24)
sample1 = np.random.choice(SMALL_POP, size=4, replace=True)
sample1_mean = np.mean(sample1)
print('Mẫu ngẫu nhiên 1: ', sample1)
print('Chiều cao trung bình của mẫu 1: {}'.format(sample1_mean))
print('Sai số ước lượng: {}'.format(abs(sample1_mean - mean_of_SMALL_POP)))
```

[186 182 157 158 152]
Chiều cao trung bình của Quần Thể : 167.0
Mẫu ngẫu nhiên 1: [157 158 186 182]
Chiều cao trung bình của mẫu 1: 170.75
Sai số ước lượng: 3.75

Nhận xét: Sai số ước lượng là: 3.75cm. Ta có thể chấp nhận được với bài toán đo chiều cao

- **VD2:** Để cho việc đo sai số khách quan, ta thử lặp lại việc lấy mẫu trên 10 lần, và tính sai số ước lượng

```
mean_array = np.empty(10)
for i in range(10):
    random_sample = np.random.choice(SMALL_POP, size=4, replace=True)
    random_sample_mean = np.mean(random_sample)
    mean_array[i] = random_sample_mean

print('Chiều cao trung bình của 10 mẫu thu được: ', mean_array)
print('Sai số ước lượng: {}'.format(abs(np.mean(mean_array) - mean_of_SMALL_POP)))
```

Chiều cao trung bình của 10 mẫu thu được: [177.75 156.25 171.75 163.5 163.5 171. 170.5 171.5 162. 154.75]
Sai số ước lượng: 0.75

Nhận xét: Ta nhận thấy, khi thực hiện việc lấy mẫu nhiều lần, sai số trung bình có nhỏ hơn so với ví dụ trên.

- **VD3:** Minh họa ảnh hưởng của cỡ mẫu đến độ chính xác của ước lượng
 - o Để rõ ràng ta sẽ tạo một quần thể mới gồm 100 cá thể: MEDUIUM_POP
 - o Lần lượt lấy mẫu với kích cỡ khác nhau sample_size = 1, 2, 3, ... và tính trung bình mẫu: mean_array
 - o Trực quan bằng đồ thị

```
MEDIUM_POP = np.random.randint(130, 200, size=100)
mean_of_MEDIUM_POP = np.mean(MEDIUM_POP)
mean_array = np.empty(100)
mean_array[0] = 0

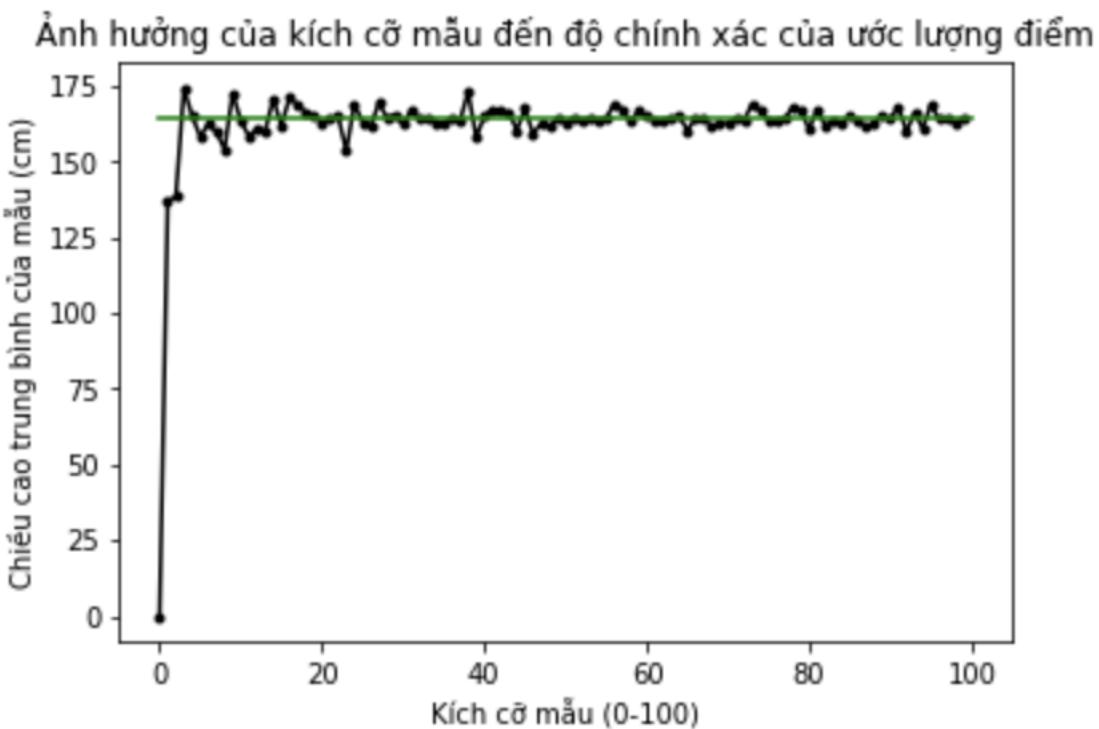
for sample_size in range(1, 100):
    temp = np.random.choice(MEDIUM_POP, size=sample_size)
    mean_array[sample_size] = np.mean(temp)

x = np.arange(100)
_ = plt.plot(x, mean_array, marker='.', color='black')

_ = plt.xlabel('Kích cỡ mẫu (0-100)')
_ = plt.ylabel('Chiều cao trung bình của mẫu (cm)')
_ = plt.title('Ảnh hưởng của kích cỡ mẫu đến độ chính xác của ước lượng điểm')

xx = np.array([0, 100])
yy = np.empty(2)
yy[0] = yy[1] = mean_of_MEDIUM_POP

_ = plt.plot(xx, yy, color='green')
print(mean_array)
plt.show()
```



- Nhận xét: Qua ví dụ trên ta có thể nhận thấy kích thước mẫu có liên quan đến độ chính xác của ước lượng điểm
- Kết luận: Để tăng độ chính xác của ước lượng ta có thể tăng kích thước của quần thể

3. Ước lượng khoảng:

- VD4: Để tăng độ chính xác ta sử dụng ước lượng khoảng thay thế cho ước lượng điểm.
 - Trước tiên, ta thực hiện ước lượng điểm 20 lần với cỡ mẫu mỗi lần là 40 để xem xét kết quả của mỗi lần ước lượng

```

MEDIUM_POP = np.random.randint(130, 200, size=100)
mean_of_MEDIUM_POP = np.mean(MEDIUM_POP)

np.random.seed(24)

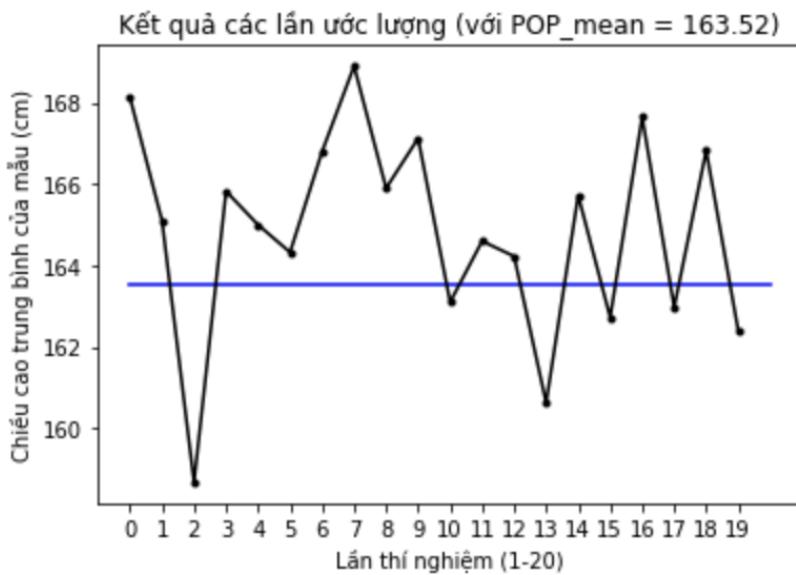
estimate_times = 20
mean_array = np.empty(estimate_times)
for i in range(estimate_times):
    a_sample = np.random.choice(MEDIUM_POP, size=40)
    mean_array[i] = np.mean(a_sample)

#Vẽ giá trị thực tế (mean của quần thể)
_ = plt.plot(np.asarray([0, estimate_times]), np.asarray([mean_of_MEDIUM_POP, mean_of_MEDIUM_POP]), color='blue')

#Vẽ các kết quả ước lượng
x = np.arange(20)
_ = plt.plot(x, mean_array, marker='.', color='black')

_ = plt.xticks(np.arange(0, estimate_times, step=1))
_ = plt.xlabel('Lần thí nghiệm (1-20)')
_ = plt.ylabel('Chiều cao trung bình của mẫu (cm)')
_ = plt.title('Kết quả các lần ước lượng (với POP_mean = {})'.format(mean_of_MEDIUM_POP))
plt.show()

```



- Nhận xét: Kết quả ước lượng nằm dao động xung quanh giá trị thực tế, có kết quả gần giá trị thực tế, nhưng cũng có kết quả rất xa
 - Hạn chế của ước lượng điểm: kết quả mỗi lần khác nhau, và có kết quả có sai số rất lớn

- Để tăng độ chính xác của ước lượng thay vì dùng điểm ước lượng ta dùng một khoảng ước lượng
- VD5: Ta vẽ lại biểu đồ trên nhưng thay vì dùng điểm ước lượng ta dùng khoảng ước lượng
 - Giả sử ta cho phép biên độ lỗi là error_margin = 3cm, như vậy khoảng ước lượng sẽ là [point_estimate - 3, point_estimate + 3]
 - Sau đó ta tỷ lệ chính xác của ước lượng bằng cách tìm tỷ lệ phần trăm kết quả ước lượng đúng

```
#Vẽ giá trị thực tế (mean của quần thể)
_ = plt.plot(np.asarray([0, estimate_times]), np.asarray([mean_of_MEDIUM_POP, mean_of_MEDIUM_POP]), color='blue')

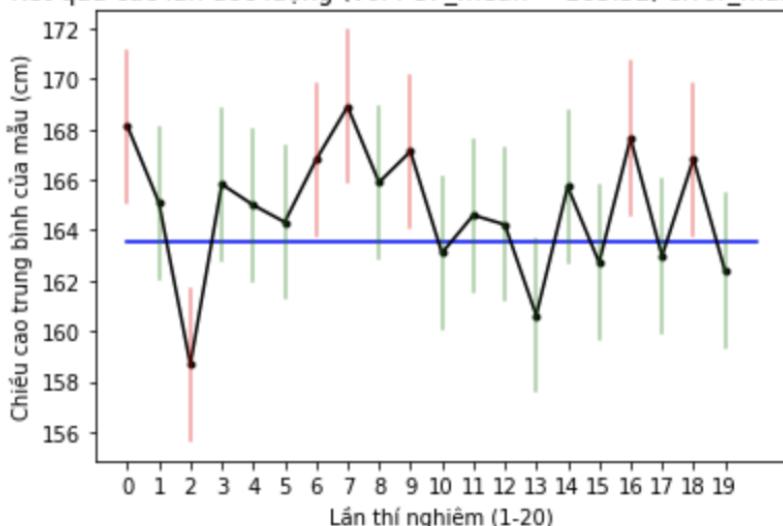
#Vẽ các kết quả ước lượng
x = np.arange(20)
_ = plt.plot(x, mean_array, marker='.', color='black')

#Vẽ khoảng ước lượng với biên độ lỗi là 3cm
error_margin = 3
true_result = 0;
for i in range(estimate_times):
    xx = np.asarray([i, i])
    lower = mean_array[i] - error_margin
    upper = mean_array[i] + error_margin
    yy = np.asarray([lower, upper])
    if (mean_of_MEDIUM_POP <= upper and mean_of_MEDIUM_POP >= lower):
        _ = plt.plot(xx, yy, color='green', alpha=0.4)
        true_result = true_result + 1
    else:
        _ = plt.plot(xx, yy, color='red', alpha=0.4)

_ = plt.xticks(np.arange(0, estimate_times, step=1))
_ = plt.xlabel('Lần thí nghiệm (1-20)')
_ = plt.ylabel('Chiều cao trung bình của mẫu (cm)')
_ = plt.title('Kết quả các lần ước lượng (với POP_mean = {}, error_margin={})'.format(mean_of_MEDIUM_POP, error_margin))
plt.show()

print('Tỷ lệ chính xác của ước lượng: {}'.format(true_result/estimate_times))
```

Kết quả các lần ước lượng (với POP_mean = 163.52, error_margin=3)



Tỷ lệ chính xác của ước lượng: 0.65

- Nhận xét:
 - - Kết quả ước lượng khoảng vẫn có thể sai (không chứa giá trị thực tế)
 - - Để tăng độ chính xác có thể tăng độ rộng của khoảng ước lượng (tăng error_margin) --> Xem như bài tập
 - - Nếu khoảng ước lượng quá rộng thì ta khó tìm giá trị thực sự trong khoảng ấy. Vậy thì độ rộng của khoảng ước lượng bao nhiêu là đủ?
- VD6: Như vậy, để thực hiện bài toán ước lượng, ta phải cân nhắc giữa 2 yếu tố là: độ rộng khoảng và độ chính xác của ước lượng.
 - Giả sử ta chấp nhận giảm độ chính xác của ước lượng xuống 95% để đổi lấy một khoảng ước lượng bé hơn
 - Điều này có nghĩa là xác suất có được một khoảng ước lượng chứa giá trị thực tế là 95%
 - Để làm điều này ta sử dụng định lý Giới Hạn Trung Tâm
 - Với độ tin cậy của ước lượng là 95%, ta sẽ tìm biên độ lỗi dựa vào một chọn ngẫu nhiên một mẫu có cùng kích cỡ

```
#Tạo ngẫu nhiên một mẫu có kích thước 40 từ MEDIUM_POP
sample2 = np.random.choice(MEDIUM_POP, size=40)
mean_sample2 = np.mean(sample2)
#Chuẩn bị các thông số
n = sample2.size
degree_of_freedom = n - 1
confidence_value = 0.95
t_score = stats.t.ppf(confidence_value, degree_of_freedom)
standard_error = sample2.std() / math.sqrt(n)
error_margin = t_score * standard_error
print('Biên độ lỗi với độ tin cậy là 95%: {}'.format(error_margin))
```

Biên độ lỗi với độ tin cậy là 95%: 5.235323284061324

- VD7: Chạy lại VD5 với biên độ lỗi mới, và kiểm tra tỷ lệ ước lượng chính xác

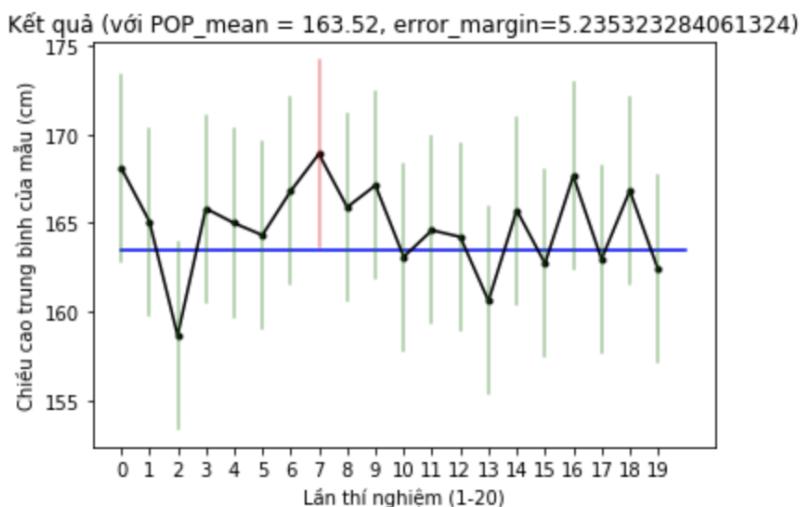
```
#Vẽ giá trị thực tế (mean của quần thể)
_ = plt.plot(np.asarray([0, estimate_times]), np.asarray([mean_of_MEDIUM_POP, mean_of_MEDIUM_POP]), color='blue')

#Vẽ các kết quả ước lượng
x = np.arange(20)
_ = plt.plot(x, mean_array, marker='.', color='black')

#Vẽ khoảng ước lượng với biên độ lỗi mới
true_result = 0;
for i in range(estimate_times):
    xx = np.asarray([i, i])
    lower = mean_array[i] - error_margin
    upper = mean_array[i] + error_margin
    yy = np.asarray([lower, upper])
    if (mean_of_MEDIUM_POP <= upper and mean_of_MEDIUM_POP >= lower):
        _ = plt.plot(xx, yy, color='green', alpha=0.4)
        true_result = true_result + 1
    else:
        _ = plt.plot(xx, yy, color='red', alpha=0.4)

_ = plt.xticks(np.arange(0, estimate_times, step=1))
_ = plt.xlabel('Lần thí nghiệm (1-20)')
_ = plt.ylabel('Chiều cao trung bình của mẫu (cm)')
_ = plt.title('Kết quả (với POP_mean = {}, error_margin={})'.format(mean_of_MEDIUM_POP, error_margin))
plt.show()

print('Tỷ lệ chính xác của ước lượng: {}'.format(true_result/estimate_times))
```



Tỷ lệ chính xác của ước lượng: 0.95

- Nhận xét: Qua ví dụ trên ta thấy với biên độ lỗi được tính từ độ tin cậy mong muốn là 95%. Ta được tỷ lệ chính xác của ước lượng cũng là 95% (với số lần thực hiện là 20)
- Bạn hãy thử tăng số lần thực hiện lên 100 hay 1000 lần xem tỷ lệ này còn chính xác không?

4. Bài tập:

Dùng python để thực hiện các bài tập sau:

Ước lượng tỉ lệ:

Hướng dẫn: áp dụng công thức:

$$\hat{p} - z_{\alpha/2} * \sqrt{\frac{\hat{p}\hat{q}}{n}} \leq p \leq \hat{p} + z_{\alpha/2} * \sqrt{\frac{\hat{p}\hat{q}}{n}}$$

1. **Green M&M Candies** liên quan đến Dataset 18 trong file excel. Tìm tỉ lệ mẫu của M&M có màu xanh lá. Sử dụng kết quả để xây dựng 1 ước lượng khoảng tin cậy 95% của % quần thể M&M có màu xanh lá. Có phải kết quả này có nhất quán với tỉ lệ 16% được báo cáo bởi nhà sản xuất kẹo. Tại sao nhất quán và tại sao không?
2. **Freshman Weight Gain** liên quan đến Dataset 3 trong file excel
 - a. Dựa vào kết quả của mẫu, tìm ước lượng điểm tốt nhất của tỉ lệ phần trăm các sinh viên cao đẳng tăng cân trong năm thứ 1.
 - b. Xây dựng ước lượng khoảng tin cậy 95% về tỉ lệ phần trăm các sinh viên cao đẳng tăng cân trong năm thứ 1.
 - c. Giả sử rằng bạn là nhà báo, viết phát biểu mô tả kết quả trên bao gồm các thông tin liên quan.
3. **Lượng mưa ở Boston**: liên quan đến Dataset 14 trong file excel, và quan tâm đến các ngày với các giá trị lượng mưa khác nhau từ 0 đến các ngày có mưa có giá trị

lượng mưa lớn hơn 0. Xây dựng ước lượng khoảng tin cậy 95% cho tỉ lệ mưa trong các ngày Thứ Tư và xây dựng ước lượng khoảng tin cậy 95% cho tỉ lệ mưa trong các ngày Chủ Nhật. So sánh kết quả. Có phải lượng mưa xuất hiện ở các ngày này nhiều hơn so với các ngày khác hay không?

- Bình chọn phim:** liên quan đến Dataset 19 trong file excel. Tìm tỉ lệ phim với tỉ lệ bình chọn là R. Sử dụng tỉ lệ đó để xây dựng ước lượng khoảng tin cậy 95% cho tỉ lệ các phim với kết quả bình chọn là R. Giả sử rằng các phim trên đã liệt kê trong file được lấy mẫu theo phương pháp lấy mẫu ngẫu nhiên đơn giản, chúng ta có thể kết luận rằng hầu như các phim có tỉ lệ bình chọn khác R không? Tại sao có hoặc tại sao không?

Ước lượng giá trị trung bình khi biết phương sai quần thể:

Hướng dẫn: áp dụng công thức:

$$\bar{x} - Z_{\alpha/2} * \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{x} + Z_{\alpha/2} * \frac{\sigma}{\sqrt{n}}$$

- Tổng số tiền phim:** liên quan đến Dataset 9 trong file excel. Xây dựng ước tính khoảng thời gian tin cậy 95% của tổng số tiền trung bình cho quần thể của tất cả các phim. Giả định rằng độ lệch chuẩn của quần thể được biết là 100 triệu đô la.
- Điểm đánh giá tín dụng FICO:** liên quan đến Dataset 24 trong file excel. Xây dựng ước lượng khoảng tin cậy 99% của điểm FICO trung bình cho quần thể. Giả sử độ lệch chuẩn của quần thể là 92.2.

Ước lượng giá trị trung bình khi chưa biết phương sai quần thể:

Hướng dẫn: áp dụng công thức:

$$\bar{x} - t_{\alpha/2} * \frac{s}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{\alpha/2} * \frac{s}{\sqrt{n}}$$

- Nicotine trong thuốc lá:** Nicotine trong thuốc lá: liên quan đến Dataset 4 trong file excel. Giả định rằng các mẫu là các mẫu ngẫu nhiên đơn giản thu được từ các quần thể có phân phối chuẩn.
 - Xây dựng ước lượng khoảng tin cậy 95% lượng nicotine trung bình trong thuốc lá có kích thước vừa (cigar), không lọc, không menthol, và không ánh sáng.
 - Xây dựng ước lượng khoảng tin cậy 95% lượng nicotine trung bình trong thuốc lá có chiều dài 100 mm, được lọc, không menthol và không ánh sáng.
 - So sánh kết quả. Bộ lọc trên thuốc lá có vẻ hiệu quả không?
- Nhịp tim:** Một bác sĩ muốn phát triển các tiêu chí để xác định xem bệnh nhân có nhịp tim không bình thường, và cô ấy muốn xác định liệu có sự khác biệt đáng kể giữa nam và nữ. Sử dụng nhịp tim mẫu trong Dataset 1.
 - Xây dựng ước lượng khoảng tin cậy 95% của nhịp tim trung bình cho nam.
 - Xây dựng ước tính khoảng tin cậy 95% của nhịp tim trung bình cho nữ.

- c. So sánh các kết quả trước đó. Chúng ta có thể kết luận rằng trung bình quần thể cho nam và nữ có khác nhau không? Tại sao có hay tại sao không?

LAB 5: KIỂM ĐỊNH

Nội dung:

- 1. Kiểm định giá trị trung bình của quần thể
- 2. Kiểm định tỉ lệ của quần thể

Dùng python để thực hiện các bài tập sau:

Kiểm định tỉ lệ của quần thể:

Hướng dẫn: áp dụng công thức:

	Lower Tail Test	Upper Tail Test	Two-Tailed Test
Hypotheses	$H_0: p \geq p_0$ $H_a: p < p_0$	$H_0: p \leq p_0$ $H_a: p > p_0$	$H_0: p = p_0$ $H_a: p \neq p_0$
Test Statistic	$z = \frac{p - p_0}{\sqrt{\frac{p_0(1 - p_0)}{n}}}$	$z = \frac{p - p_0}{\sqrt{\frac{p_0(1 - p_0)}{n}}}$	$z = \frac{p - p_0}{\sqrt{\frac{p_0(1 - p_0)}{n}}}$
Rejection Rule: p-Value Approach	Reject H_0 if $p\text{-value} \leq \alpha$	Reject H_0 if $p\text{-value} \leq \alpha$	Reject H_0 if $p\text{-value} \leq \alpha$
Rejection Rule: Critical Value Approach	Reject H_0 if $z \leq -z_\alpha$	Reject H_0 if $z \geq z_\alpha$	Reject H_0 if $z \leq -z_{\alpha/2}$ or if $z \geq z_{\alpha/2}$

- M&Ms:** liên quan đến Dataset 18 trong file excel. Tìm tỷ lệ mẫu của M & Ms có màu đỏ. Sử dụng kết quả đó để kiểm tra phát biểu của công ty Mars: “20% kẹo M & M một màu có màu đỏ”.
- Sinh viên năm nhất:** liên quan đến Dataset 3 trong file excel bao gồm kết quả từ một nghiên cứu được mô tả trong “Những thay đổi về trọng lượng cơ thể và khối lượng chất béo của nam và nữ trong năm đầu tiên của trường đại học: Một nghiên cứu về “Sinh viên năm nhất” của Hoffman, Policastro, Quick và Lee, tạp chí về sức khỏe của Đại học Mỹ, tập 55, số 1. Hãy tham khảo tập dữ liệu đó và tìm tỷ lệ nam trong nghiên cứu đó. Sử dụng mức ý nghĩa 0.05 để kiểm tra phát biểu: “khi các đối tượng được chọn ra để nghiên cứu, các đối tượng được chọn từ quần thể trong đó tỷ lệ nam là bằng 50%”.

3. **Gấu:** liên quan đến Dataset 6 trong file excel. Tìm tỷ lệ gấu đực trong nghiên cứu. Sử dụng mức ý nghĩa 0.05 để kiểm tra phát biểu: “khi gấu được chọn, chúng được chọn từ quần thể trong đó tỷ lệ gấu đực bằng 50%”.
4. **Phim:** theo phim “**Information Please**”, tỷ lệ phần trăm phim có xếp hạng R là 55% trong thời gian 33 năm gần đây. Tham khảo Dataset 9 trong file excel và tìm tỷ lệ phim có xếp hạng R. Sử dụng mức ý nghĩa 0.01 để kiểm tra phát biểu: “các bộ phim trong Dataset 9 được chọn từ quần thể có 55% phim có xếp hạng R”.

Kiểm định giá trị trung bình của quần thể khi phương sai của quần thể đã biết:

Hướng dẫn: áp dụng công thức:

	Lower Tail Test	Upper Tail Test	Two-Tailed Test
Hypotheses	$H_0: \mu \geq \mu_0$ $H_a: \mu < \mu_0$	$H_0: \mu \leq \mu_0$ $H_a: \mu > \mu_0$	$H_0: \mu = \mu_0$ $H_a: \mu \neq \mu_0$
Test Statistic	$z = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}}$	$z = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}}$	$z = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}}$
Rejection Rule: p-Value Approach	Reject H_0 if $p\text{-value} \leq \alpha$	Reject H_0 if $p\text{-value} \leq \alpha$	Reject H_0 if $p\text{-value} \leq \alpha$
Rejection Rule: Critical Value Approach	Reject H_0 if $z \leq -z_\alpha$	Reject H_0 if $z \geq z_\alpha$	Reject H_0 if $z \leq -z_{\alpha/2}$ or if $z \geq z_{\alpha/2}$

5. **Các vít có chiều dài 3/4 in?** Một mẫu ngẫu nhiên đơn giản là 50 vít kim loại bằng thép không gỉ được lấy từ những vít được cung cấp bởi công ty Crown Bolt, và chiều dài của mỗi vít được đo bằng cách sử dụng một thước caliper. Độ dài được liệt kê trong Dataset 19 trong file excel. Giả sử rằng độ lệch chuẩn của các chiều dài vít là 0.012. Sử dụng mức ý nghĩa 0.05 để kiểm tra phát biểu: “các vít có chiều dài trung bình bằng 3/4 in (hoặc 0.75 in.), như ghi chú trên nhãn gói”. Chiều dài vít có nhất quán với số liệu ghi trên nhãn gói hay không?
6. **Cung cấp điện:** Dataset 13 trong file excel liệt kê các mức điện áp được cung cấp trực tiếp từ nhà của chủ hộ. Công ty cung cấp điện Hudson nói rằng: “mục tiêu cung cấp điện là 120 volt”. Sử dụng những mức điện áp đó, giả sử rằng độ lệch chuẩn của tất cả các mức điện áp là 0.24, hãy kiểm tra phát biểu: “trung bình các mức điện áp là 120 volt”. Sử dụng mức ý nghĩa 0.01.

Kiểm định giá trị trung bình của quần thể khi phương sai của quần thể chưa biết:

Hướng dẫn: áp dụng công thức:

	Lower Tail Test	Upper Tail Test	Two-Tailed Test
Hypotheses	$H_0: \mu \geq \mu_0$ $H_a: \mu < \mu_0$	$H_0: \mu \leq \mu_0$ $H_a: \mu > \mu_0$	$H_0: \mu = \mu_0$ $H_a: \mu \neq \mu_0$
Test Statistic	$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$	$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$	$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$
Rejection Rule: <i>p</i>-Value Approach	Reject H_0 if <i>p</i> -value $\leq \alpha$	Reject H_0 if <i>p</i> -value $\leq \alpha$	Reject H_0 if <i>p</i> -value $\leq \alpha$
Rejection Rule: Critical Value Approach	Reject H_0 if $t \leq -t_{\alpha/2}$	Reject H_0 if $t \geq t_{\alpha/2}$	Reject H_0 if $t \leq -t_{\alpha/2}$ or if $t \geq t_{\alpha/2}$

7. **Các vít có chiều dài $3/4$ in?** Một mẫu ngẫu nhiên đơn giản là 50 vít kim loại bằng thép không gỉ được lấy từ những vít được cung cấp bởi công ty Crown Bolt, và chiều dài của mỗi vít được đo bằng cách sử dụng một thước caliper. Độ dài được liệt kê trong Dataset 19 trong file excel. Sử dụng mức ý nghĩa 0.05 để kiểm tra phát biểu: “các vít có chiều dài trung bình bằng $3/4$ in (hoặc 0.75 in.), như ghi chú trên nhãn gói”. Chiều dài vít có nhất quán với số liệu ghi trên nhãn gói hay không?
8. **Cung cấp điện:** Dataset 13 trong file excel liệt kê các mức điện áp đo được được cung cấp trực tiếp từ nhà của chủ hộ. Công ty cung cấp điện Hudson nói rằng: “mục tiêu cung cấp điện là 120 volt”. Sử dụng những số điện áp nhà đó, hãy kiểm tra phát biểu: “trung bình là 120 volt”. Sử dụng mức ý nghĩa 0.01.
9. **$98,6^\circ F$ có sai không?** Dataset 2 trong file excel bao gồm nhiệt độ cơ thể người đo được. Sử dụng nhiệt độ được liệt kê lúc 12 giờ sáng vào ngày thứ 2 để kiểm tra niềm tin chung rằng nhiệt độ cơ thể là $98.6^\circ F$. Liệu niềm tin này có vé sai không?
10. **Điểm đánh giá tín dụng FICO:** Dataset 24 trong file excel bao gồm một mẫu ngẫu nhiên đơn giản về điểm xếp hạng tín dụng FICO. Theo dữ liệu này, số điểm FICO trung bình được báo cáo là 678. Sử dụng mức có ý nghĩa 0.05 để kiểm tra phát biểu: “mẫu điểm FICO được lấy từ quần thể có giá trị trung bình bằng 678”.

LAB 6: HỒI QUY TUYẾN TÍNH ĐƠN BIỂN

Nội dung:

1. Vẽ đồ thị phân tán (scatter plot) thể hiện mối tương quan giữa 2 đại lượng
2. Tính hệ số tương quan giữa 2 đại lượng
3. Xây dựng phương trình hồi quy tuyến tính
4. Kiểm định phương trình hồi quy tuyến tính
5. Tính khoảng sai số khi dự đoán các đại lượng
6. Xác định và xử lý các giá trị có ảnh hưởng đến phương trình hồi quy
7. Dựa vào phương trình hồi quy đã xây dựng để dự đoán

Dữ liệu: Dữ liệu sử dụng trong lab này là dữ liệu về kích thước giáp cua. (Dữ liệu được chuẩn bị sẵn trong tập tin: crabs.txt).

Mô tả dữ liệu:

Tên cột	Ý nghĩa
Premolt	Kích thước giáp cua trước khi lột vỏ (tính bằng mm)
Postmolt	Kích thước giáp cua sau khi lột vỏ (tính bằng mm)
Increment	Hiệu số giữa postmolt và premolt
Year	Năm (81: năm 1981, 82: năm 1982, 92: 1992).
Source	Nguồn gốc của cua: 1: lột vỏ trong phòng thí nghiệm; 0: lột vỏ trong tự nhiên.

Trong lab này, ta xem xét các vấn đề sau:

- **Tìm mối quan hệ giữa kích thước của giáp cua trước khi lột vỏ và sau khi lột vỏ.**
- **Dự đoán kích thước của giáp cua trước khi lột vỏ dựa vào thông tin về kích thước của giáp cua sau khi lột vỏ.**

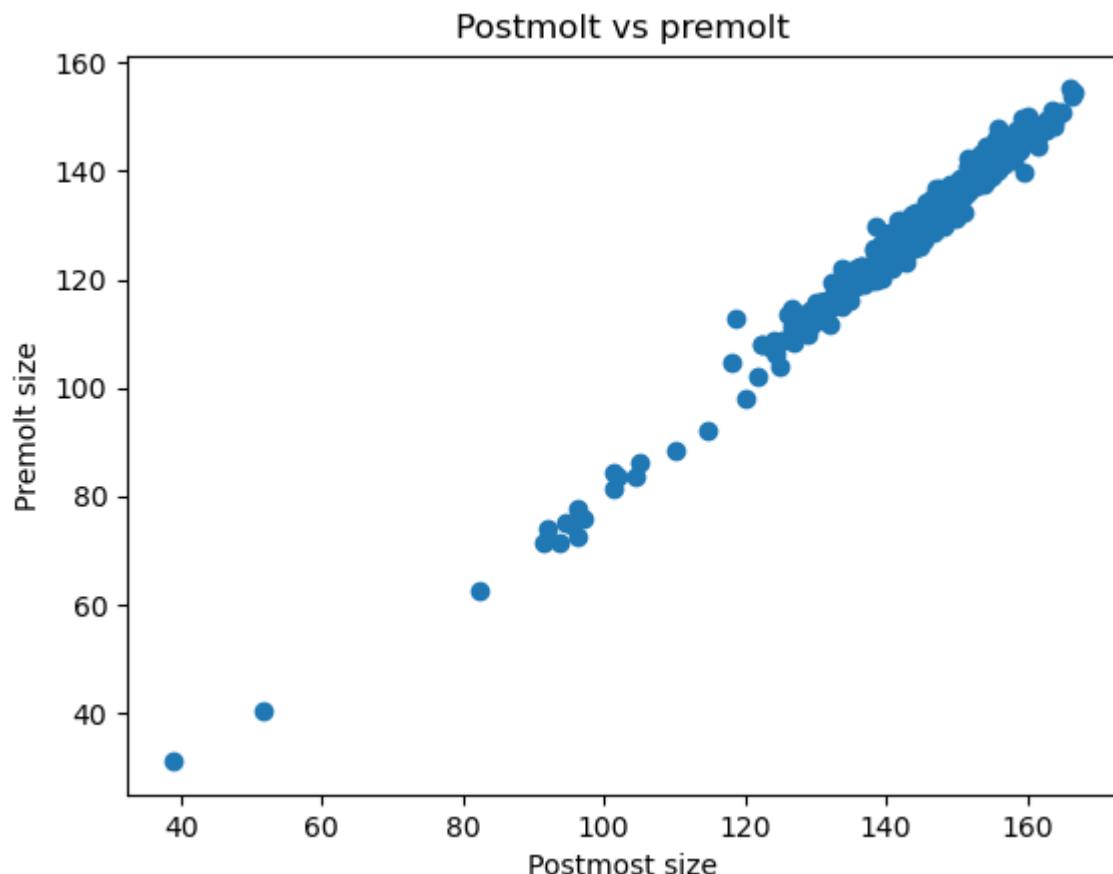
Trong lab này, ta thực hiện các nội dung sau:

- Vẽ đồ thị phân tán thể hiện mối tương quan giữa kích thước của giáp cua sau khi lột vỏ và trước khi lột vỏ (postmolt và premolt)
- Tính hệ số tương quan giữa kích thước của giáp cua sau khi lột vỏ và trước khi lột vỏ (postmolt và premolt)
- Xây dựng phương trình hồi quy
- Kiểm định xem phương trình hồi quy có khớp với dữ liệu không
- Tính khoảng sai số khi dự đoán giá trị premolt dựa vào postmolt
- Xác định và xử lý các giá trị có ảnh hưởng đến phương trình hồi quy
- Dựa vào phương trình hồi quy đã xây dựng để dự đoán

1. Vẽ đồ thị scatter plot thể hiện mối tương quan giữa postmolt và premolt

Dùng python để vẽ scatter plot thể hiện mối tương quan giữa postmolt và premolt

Kết quả:



Nhận xét: dữ liệu tập trung theo dạng đường thẳng.

2. Tính hệ số tương quan giữa postmolt và premolt

Dùng python tính hệ số tương quan giữa postmolt và premolt

Có nhận xét gì về hệ số tương quan đã tính được?

Kết quả:

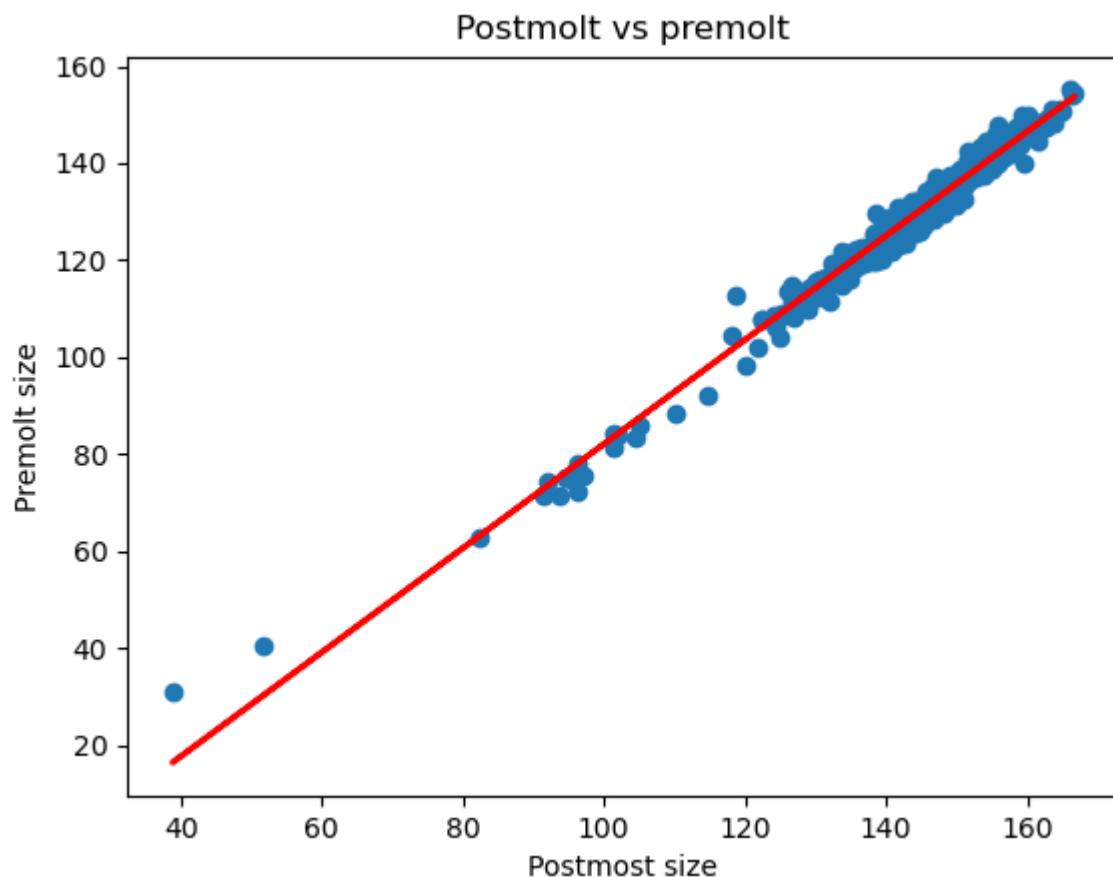
He so tuong quan la:
(0.9903699282533851, 0.0)

Nhận xét: Hệ số tương quan là 0.9903699282533851, có giá trị gần với 1, P-value=0.0 < α (0.05) nghĩa là giữa 2 đại lượng Postmolt và Premolt có mối quan hệ tuyến tính mạnh, mối quan hệ này có ý nghĩa thống kê.

3. Xây dựng phương trình hồi quy tuyến tính

Dùng python để xây dựng phương trình hồi quy tuyến tính giữa postmolt và premolt.

Kết quả:



Kết quả:

OLS Regression Results						
Dep. Variable:	PreMolt	R-squared:	0.981			
Model:	OLS	Adj. R-squared:	0.981			
Method:	Least Squares	F-statistic:	2.405e+04			
Date:	Tue, 15 Sep 2020	Prob (F-statistic):	0.00			
Time:	12:41:00	Log-Likelihood:	-1040.6			
No. Observations:	472	AIC:	2085.			
Df Residuals:	470	BIC:	2094.			
Df Model:	1					
Covariance Type:	nonrobust					

	coef	std err	t	P> t	[0.025	0.975]

Intercept	-25.2137	1.001	-25.191	0.000	-27.180	-23.247
PostMolt	1.0732	0.007	155.083	0.000	1.060	1.087

Omnibus:	107.875	Durbin-Watson:	1.684			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	625.191			
Skew:	0.845	Prob(JB):	1.74e-136			
Kurtosis:	8.379	Cond. No.	1.43e+03			

Giải thích:

No. Observations: số lượng đối tượng trong mẫu quan sát là n=472

Df Residuals: bậc tự do của phần dư = n- k (k: số lượng tham số trong phương trình hồi quy) = 472-2=470.

Df Model: bậc tự do của mô hình = k-1=2-1=1.

R-squared: có nghĩa là 98.1% kích thước giáp cua trước khi lột vỏ có thể được giải thích bởi biến dự báo.

Adj. R-squared: được sử dụng trong hồi quy đa biến. Trong hồi quy đơn biến thì **Adj. R-squared= R-squared**. Trong bài này **Adj. R-squared= R-squared=0.981**. Dùng **Adj. R-squared** để xác định phương trình hồi quy với số biến tham gia nào là tốt nhất. Chọn các phương trình hồi quy có giá trị **Adj. R-squared** cao và chỉ bao gồm một ít biến.

F-statistic: dùng trong hồi quy đa biến, ta kiểm định lại các hệ số $\beta_1, \beta_2, \dots, \beta_n$ bằng 0 hay không bằng cách kiểm định giả thuyết sau:

- $H_0: \beta_1 = \beta_2 = \dots = \beta_n = 0$
- $H_A: \beta_j \neq 0$

Prob (F-statistic): P-value (F-statistic). Nếu P-value (F-statistic) < α thì bác bỏ giả thuyết H_0 .

AIC và BIC: là viết tắt của Akaike's Information Criteria và được sử dụng để lựa chọn mô hình **trong hồi quy đa biến**. AIC là phép tính để dung hòa tổng bình phương lỗi và số biến độc lập tham gia vào mô hình. AIC thấp hơn ngụ ý một mô hình tốt hơn. BIC là viết tắt Bayesian information criteria và là một biến thể của AIC. BIC thấp hơn ngụ ý một mô hình tốt hơn.

Log-Likelihood: một cách tiếp cận rất phổ biến trong thống kê là ý tưởng về ước tính khả năng xảy ra tối đa (maximum likelihood). Ý tưởng cơ bản hoàn toàn khác với cách tiếp cận OLS (bình phương nhỏ nhất): trong phương pháp tiếp cận bình phương nhỏ nhất, mô hình là không đổi, và sai số của biến phản hồi có thể thay đổi; ngược lại, trong phương pháp tiếp cận ước tính khả năng xảy ra tối đa, các giá trị phản hồi dữ liệu là được coi là không đổi, và khả năng của mô hình được tối đa hóa.

Regression coefficient (coef): hệ số hồi quy. Kết quả tính toán cho thấy β_0 (Intercept) = -25.2137 và β_1 (PostMolt) = 1.0732. Với 2 thông số này, chúng ta có thể ước tính của kích thước giáp cua trước khi lột cho bất cứ kích thước của giáp cua sau khi lột (trong khoảng kích thước giáp cua sau khi lột của mẫu) bằng phương trình tuyến tính:

$$\hat{y}_i = -25.2137 + 1.0732 * \text{PostMolt}.$$

Phương trình này có nghĩa là khi tăng kích thước giáp cua sau khi lột vỏ lên 1 đơn vị thì kích thước giáp cua trước khi lột vỏ tăng lên 1.0732 đơn vị.

Standard error: đo độ chính xác của hệ số β_1 (PostMolt) bằng cách ước tính sự biến thiên của hệ số nếu cùng 1 thử nghiệm chạy trên một mẫu khác nhau được lấy mẫu từ quần thể. Tương tự đối với hệ số hồi quy β_0 (Intercept).

t: ta kiểm định lại hệ số $\beta_0 = 0$, $\beta_1 = 0$ hay không bằng cách kiểm định giả thuyết sau:

Kiểm định hệ số β_0 :

- $H_0: \beta_0 = 0$
- $H_A: \beta_0 \neq 0$

Và kiểm định hệ số β_1 :

- $H_0: \beta_1 = 0$
- $H_A: \beta_1 \neq 0$

Với $\hat{\beta}_0 = -25.2137$, $\hat{\beta}_1 = 1.0732$

$$t_0 = \frac{\hat{\beta}_0 - \beta_0}{\text{s.e.}(\hat{\beta}_0)} = \frac{\hat{\beta}_0 - 0}{\text{s.e.}(\hat{\beta}_0)} = -25.191$$

$$t_1 = \frac{\hat{\beta}_1 - \beta_1}{\text{s.e.}(\hat{\beta}_1)} = \frac{\hat{\beta}_1 - 0}{\text{s.e.}(\hat{\beta}_1)} = 155.083$$

P-value (t_0) = $P > |t_0| = 2 * (1 - t_{(n-2, \alpha/2)}(|t_0|)) = 0$

P-value (t_1) = $P > |t_1| = 2 * (1 - t_{(n-2, \alpha/2)}(|t_1|)) = 0$

Vì P-value (t_0) = $0 < \alpha (0.05)$ nên bác bỏ giả thuyết H_0

Vì P-value (t_1) = $0 < \alpha (0.05)$ nên bác bỏ giả thuyết H_0

Chúng ta có bằng chứng để cho rằng có mối liên hệ giữa kích thước giáp cua trước khi lột và kích thước giáp cua sau khi lột, mối liên hệ này có ý nghĩa thống kê.

P>|t|: P-value (t) = $\Pr(T > |t|)$. Nếu P-value (t) $< \alpha$ thì bác bỏ giả thuyết H_0 .

Confidence interval: phạm vi mà hệ số hồi quy dao động. Kết quả cho thấy rằng, chúng ta tin tưởng 95% rằng hệ số β_0 dao động từ -27.180 đến -23.247, hệ số β_1 dao động từ 1.060 đến 1.087. Kết quả trên được tính như sau:

Khoảng tin cậy cho hệ số β_0 là từ $\hat{\beta}_0 - t_{(n-2,\alpha/2)} * s.e.(\hat{\beta}_0)$ đến $\hat{\beta}_0 + t_{(n-2,\alpha/2)} * s.e.(\hat{\beta}_0)$

Tính $t_{(n-2, \alpha/2)} = t_{(470, 0.0025)} = \text{stats.t}(470).ppf(0.975)$ (from scipy import stats)

$$t_{(n-2, \alpha/2)} = t_{(470, 0.0025)} = 1.965$$

Thay $\hat{\beta}_0 = -25.2137$ và $t_{(n-2, \alpha/2)} = 1.965$ vào biểu thức trên ta được khoảng tin cậy cho hệ số β_0 là:

$$\begin{aligned} -25.2137 - 1.965 * 1.001 &\text{ đến } -25.2137 + 1.965 * 1.001 \\ &- 27.180 \text{ đến } -23.247 \end{aligned}$$

Khoảng tin cậy cho hệ số β_1 là từ $\hat{\beta}_1 - t_{(n-2,\alpha/2)} * s.e.(\hat{\beta}_1)$ đến $\hat{\beta}_1 + t_{(n-2,\alpha/2)} * s.e.(\hat{\beta}_1)$

Thay $\hat{\beta}_1 = 1.0732$ và $t_{(n-2, \alpha/2)} = 1.965$ vào biểu thức trên ta được khoảng tin cậy cho hệ số β_1 là:

$$\begin{aligned} 1.0732 - 1.965 * 0.007 &\text{ đến } 1.0732 + 1.965 * 0.007 \\ &1.060 \text{ đến } 1.087 \end{aligned}$$

Skew và kurtosis: Skew và kurtosis đề cập đến hình dạng của một phân phối, giá trị **skew** để đo độ “lệch (trái, phải)” của dữ liệu (đối với dữ liệu được phân phối chuẩn, skew có giá trị khoảng bằng 0), **kurtosis**, là giá trị để đo độ “bè-nhọn” của đỉnh dữ liệu (đối với dữ liệu được phân phối chuẩn, kurtosis có giá trị khoảng bằng 3). Trong bài này, **skew=0.845** và **kurtosis=8.379** nên phần dư không phân phối chuẩn.

Omnibus: kiểm định **Omnibus** sử dụng **skew** và **kurtosis** để kiểm tra giả thuyết Null: phân phối của phần dư là phân phối chuẩn. Nếu **P-value (Omnibus) < α**, thì phần dư không phân phối chuẩn, chúng ta cần xem xét lại mô hình.

Prob (Omnibus): P-value của Omnibus. Trong bài này, **Prob (Omnibus)=0.00<α** nên phần dư không phân phối chuẩn.

Durbin-Watson: kiểm định **Durbin - Watson** được sử dụng để phát hiện **sự hiện diện của sự tự tương quan** trong phần dư từ phân tích hồi quy. Giá trị thống kê Durbin-Watson sẽ luôn có giá trị từ 0 đến 4. Giá trị 2.0 có nghĩa là không có hiện tượng tự tương quan được phát hiện trong mẫu. Các giá trị từ 0 đến nhỏ hơn 2 cho biết tự tương quan dương và các giá trị từ 2 đến 4 cho biết tự tương quan âm.

Jarque-Bera: kiểm định **Jarque-Bera** là một dạng kiểm định khác xem xét độ lệch **skew** và **kurtosis**. Giả thuyết Null: phân phối của phần dư là phân phối chuẩn, hoặc nói một cách khác, **skew=0** và **kurtosis=3**. Nếu **P-value (JB) < α**, thì phần dư không phân phối chuẩn, chúng ta cần xem xét lại mô hình.

Prob (JB): P-value của JB: Trong bài này, Prob (JB)=1.74e-136<α nên phần dư không phân phối chuẩn.

Cond. No.: (dùng trong hồi quy đa biến) đo lường độ nhạy của đầu ra của một hàm đối với đầu vào của hàm. Khi hai biến dự báo có tương quan cao, được gọi là **multicollinearity**, các hệ số hồi quy có thể dao động thất thường đối với những

thay đổi nhỏ trong dữ liệu hoặc mô hình. **Multicollinearity** có thể gây ra kết quả không chính xác, cần xem lại mô hình. Nếu **Cond. No.** lớn hơn 30, thì hồi quy có thể gặp phải trường hợp **multicollinearity**.

Kết luận: qua các phân tích trên, phần dư không tuân theo phân phối chuẩn, do đó, mô hình hồi quy trên cần xây dựng lại bằng cách loại bỏ các giá trị có ảnh hưởng đến phương trình hồi quy.

4. Kiểm định phương trình hồi quy tuyến tính

Dùng python để kiểm định lại phương trình hồi quy bằng các đồ thị sau:

- Dùng đồ thị Residual value vs Fitted value
- Dùng đồ thị Normal Q-Q
- Dùng đồ thị Scale-location
- Dùng đồ thị Residual vs Leverage

Hướng dẫn:

- Đồ thị Residual value vs Fitted value:

Đồ thị vẽ phần dư e_i và giá trị dự đoán Premolt \hat{y}_i . Đồ thị này cho thấy các giá trị phần dư tập trung quanh đường $y=0$, tuy nhiên, có một vài điểm dữ liệu không tập trung quanh đường $y=0$ cho nên **giả định e_i có giá trị trung bình là 0 là không chấp nhận được**.

- Đồ thị Normal Q-Q:

Đồ thị vẽ giá trị phần dư và giá trị kỳ vọng dựa vào phân phối chuẩn. Chúng ta thấy các số phần dư tập trung rất gần các giá trị trên đường chuẩn, tuy nhiên có một số điểm bị lệch nhiều khỏi đường chuẩn, và do đó, **giả định e_i phân phối theo luật phân phối chuẩn không thể đáp ứng**.

- Đồ thị Scale-location:

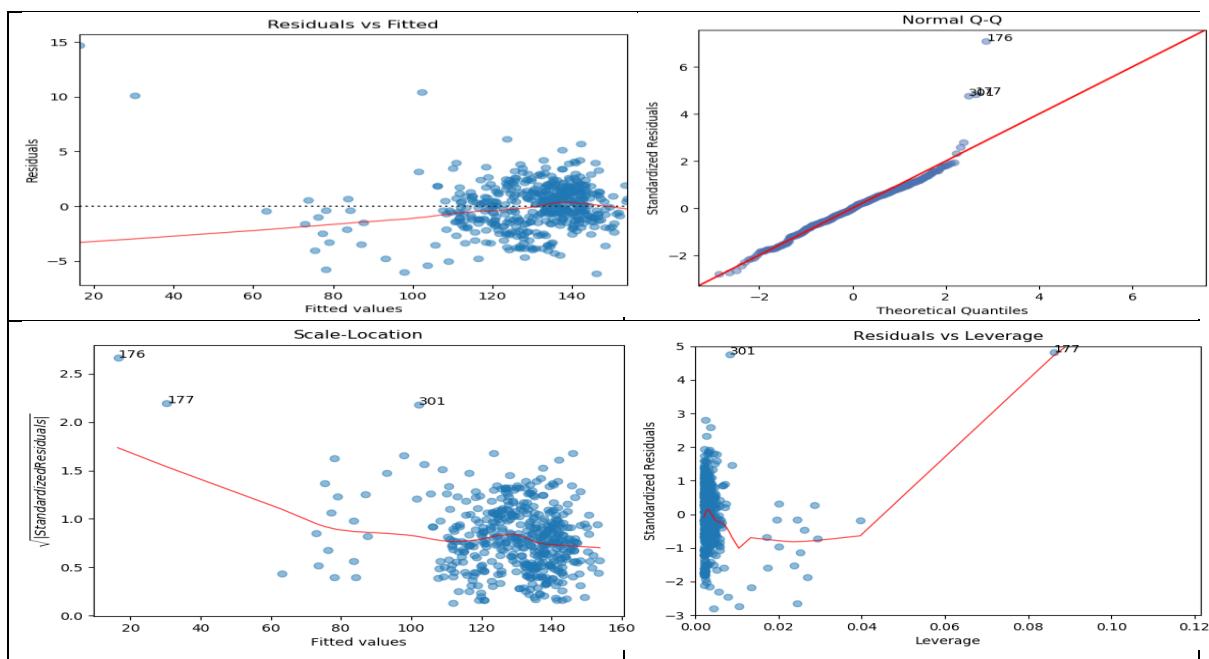
Đồ thị vẽ căn bậc 2 của phần dư chuẩn (standardized residual) và giá trị dự đoán \hat{y}_i . Đồ thị này cho thấy không có gì khác giữa các số phần dư chuẩn cho các giá trị dự đoán \hat{y}_i , và do đó, **giả định các e_i có phương sai σ^2 cố định cho tất cả các x_i có thể áp dụng**.

- Đồ thị Residual vs Leverage:

Đồ thị này giúp xem các giá trị ngoại lệ trong mô hình hồi quy tuyến tính có ảnh hưởng đến việc phân tích hồi quy hay không. Nếu có thì cần loại bỏ các giá trị ngoại lệ khỏi tập dữ liệu.

Dựa vào đồ thị này:

- Các điểm có leverage cao khi $h_{ii} > \frac{3p}{n} = \frac{3*2}{472} = 0.0127$ (p: số các tham số cần ước lượng (trong bài này cần ước lượng 2 tham số β_0 và β_1 nên p=2), n: kích thước mẫu)
- Các điểm là outlier khi Standard Residual > 3
- Các điểm có ảnh hưởng đến phương trình hồi quy (influence point) cần loại bỏ là các điểm outlier và có leverage cao



Dựa vào các đồ thị phân tích phần dư, bạn có kết luận gì về tính hợp lý của phương trình hồi quy đã xây dựng?

5. Tính khoảng sai số khi dự đoán

Dùng python để ước lượng các hệ số β_0, β_1 với độ tin cậy $1-\alpha=0.95$

Kết quả:

```
0      1
Intercept -27.180474 -23.246931
postmolt   1.059565  1.086760
```

Khoảng tin cậy cho hệ số β_0 là từ -27.180 đến -23.247

Khoảng tin cậy cho hệ số β_1 là từ 1.060 đến 1.087

6. Xác định và xử lý các giá trị có ảnh hưởng đến phương trình hồi quy

- Sử dụng đồ thị Residual vs Leverage (hoặc sử dụng khoảng cách Cook (đồ thị Cook's dist vs Leverage), khoảng cách Dffits (đồ thị Cook's dist vs Leverage)) để xác định các điểm có ảnh hưởng đến phương trình hồi quy.
- Xây dựng phương trình hồi quy nếu loại bỏ các giá trị có ảnh hưởng đến phương trình hồi quy (phần này sinh viên tự thực hiện).
- So sánh sự khác biệt giữa hai mô hình: đánh giá xem sự khác biệt có đáng kể không. Kết luận về sự quan trọng của các giá trị có ảnh hưởng đến phương trình hồi quy (phần này sinh viên tự thực hiện).
- Kết luận: bỏ hay giữ các giá trị có ảnh hưởng đến phương trình hồi quy (phần này sinh viên tự thực hiện).

7. Dựa vào phương trình hồi quy đã xây dựng để dự đoán

Giả sử giá trị postmolt size là: 85, dựa vào phương trình hồi quy đã xây dựng, với độ tin cậy là $1-\alpha=0.95$, bạn dự đoán giá trị premolt size nằm trong khoảng nào?

Bài tập làm thêm:

Dùng python để thực hiện các bài tập sau:

1. **Chiều cao và cân nặng:** DataSet 1 liệt kê chiều cao (tính bằng inch) và cân nặng (tính bằng lb) của 40 nam được chọn ngẫu nhiên.
 - a. Xây dựng scatterplot thể hiện mối tương quan giữa tính giữa chiều cao và cân nặng của nam.
 - b. Tính hệ số tương quan giữa chiều cao và cân nặng của nam.
 - c. Từ scatter plot, và hệ số tương quan, có mối quan hệ tương quan tuyến tính giữa chiều cao và cân nặng của nam hay không?
 - d. Tìm phương trình hồi quy, giả sử trục y đại diện cho cân nặng của nam và để x đại diện cho chiều cao của nam.
 - e. Dựa trên dữ liệu mẫu đã cho, trọng lượng của nam được dự đoán tốt nhất là bao nhiêu với chiều cao là 72 inch.
2. **Nhiệt độ cơ thể:** DataSet 2 liệt kê nhiệt độ cơ thể (tính bằng °F) của các đối tượng được đo lúc 8:00 sáng và vào lúc nửa đêm.
 - a. Xây dựng scatterplot thể hiện mối tương quan giữa nhiệt độ cơ thể lúc 8:00 sáng và nhiệt độ cơ thể lúc nửa đêm.
 - b. Tính hệ số tương quan giữa nhiệt độ cơ thể lúc 8:00 sáng và nhiệt độ cơ thể lúc nửa đêm
 - c. Từ scatter plot, và hệ số tương quan, có mối quan hệ tương quan tuyến tính giữa nhiệt độ cơ thể lúc 8:00 sáng và nhiệt độ cơ thể lúc nửa đêm hay không?
 - d. Tìm phương trình hồi quy, giả sử trục y đại diện cho nhiệt độ lúc nửa đêm và để x đại diện cho nhiệt độ lúc 8:00 sáng.
 - e. Dựa trên dữ liệu mẫu đã cho, nhiệt độ cơ thể nửa đêm của người nào đó được dự đoán tốt nhất là bao nhiêu với thân nhiệt là 98.3°F đo lúc 8:00 sáng?

LAB 7: HỒI QUY ĐA BIẾN

Nội dung:

- | | |
|-----------|--|
| 1. | Xây dựng phương trình hồi quy đa biến |
| 2. | Lựa chọn phương trình hồi quy tốt nhất |

Bài 1:

Dữ liệu: Dữ liệu sử dụng trong lab này là tập dữ liệu về lượng nhựa, nicotine, CO trong thuốc lá cõi king. (Dữ liệu được chuẩn bị sẵn trong tập tin: 04_CIGARET.xls).

Mô tả dữ liệu:

Tên cột	Ý nghĩa
KgTar	lượng nhựa trong 1 điếu thuốc lá
KgNic	lượng nicotine trong 1 điếu thuốc lá
KgCO	lượng CO trong 1 điếu thuốc lá

1. Xây dựng phương trình hồi quy thể hiện mối liên hệ giữa lượng nicotine trong thuốc lá và lượng nhựa, CO trong thuốc lá.

2. Bạn hãy xác định phương trình hồi quy trên có thể sử dụng để dự đoán lượng nicotine trong thuốc lá khi biết lượng nhựa và CO trong thuốc lá không? Vì sao có hoặc vì sao không?

Bài 2:

Dữ liệu: Dữ liệu sử dụng trong lab này là dữ liệu về giá bán nhà. (Dữ liệu được chuẩn bị sẵn trong tập tin: 23_HOMES.xls).

Mô tả dữ liệu:

Tên cột	Ý nghĩa
Selling_Price	giá bán
List_Price	giá niêm yết
Area	diện tích sử dụng của ngôi nhà
Acres	diện tích đất

1. Nếu chỉ sử dụng 1 biến x để dự đoán giá nhà, phương trình hồi quy 1 biến dự đoán (predictor) nào sau đây là tốt nhất? Tại sao?

Predictor (x) Variables	P-value	Adjusted <i>R</i> ²		Regression Equation
		<i>R</i> ²	<i>R</i> ²	
LP, LA, Lot	0.000	0.990	0.989	$\hat{y} = 1120 + 0.972 LP + 0.281 LA + 465 LOT$
LP, LA	0.000	0.990	0.989	$\hat{y} = -40.5 + 0.985 LP - 0.985 LA$
LP, LOT	0.000	0.990	0.989	$\hat{y} = 1004 + 0.974 LP + 429 LOT$
LA, LOT	0.000	0.815	0.805	$\hat{y} = 111,309 + 98.2 LA + 17,269 LOT$
LP	0.000	0.990	0.990	$\hat{y} = 99.2 + 0.979 LP$
LA	0.000	0.643	0.633	$\hat{y} = 133,936 + 101 LA$
LOT	0.003	0.215	0.194	$\hat{y} = 310,191 + 19,217 LOT$

2. Nếu sử dụng đúng 2 biến dự đoán để dự đoán giá nhà, phương trình hồi quy 2 biến dự đoán (predictor) nào ở trên là tốt nhất? Tại sao?
 3. Phương trình hồi quy nào trong số các phương trình hồi quy trên là tốt nhất để dự đoán giá nhà? Tại sao?
 4. Một ngôi nhà được rao bán với giá niêm yết 400.000 USD, nó có diện tích là 3000 feet vuông, và diện tích đất rộng 2 mẫu. Giá trị dự đoán tốt nhất của giá bán là bao nhiêu? Giá bán dự đoán có thể là ước lượng tốt hay không? Giá trị dự đoán đó có khả năng rất chính xác không?

PHỤ LỤC: CÁC DATASET DÙNG TRONG BÀI TẬP THỰC HÀNH

Data Set 1: Body Measurements

File name: 01_FHEALTH.XLS, 01_MHEALTH.XLS

01_FHEALTH.XLS: for female

01_MHEALTH.XLS: for male

Age: in years

HT: height in inches

WT: weight in pounds

WAIST: waist circumference in cm

PULSE: pulse rate in beats per minute

SYS: systolic blood pressure in mmHg

DIAS: diastolic blood pressure in mmHg

CHOL: cholesterol in mg

BMI: body mass index

LEG: upper leg length in cm

ELBOW: elbow breadth in cm

WRIST: wrist breadth in cm

ARM: arm circumference in cm

Data are from the U.S. Department of Health and Human Services, National Center for Health Statistics, Third National Health and Nutrition Examination Survey.

Data Set 2: Body Temperatures (in degrees Fahrenheit) of Healthy Adults

File name: 02_BODYTEMP.XLS

Body temperatures, in degrees Fahrenheit, of healthy adults.

Data Set 3: Freshman 15 Data

File name: 03_FRESH15.XLS

Weights are in kilograms, and BMI denotes measured body mass index. Measurements were made in September of freshman year and then later in April of freshman year. Results are published in Hoffman, D.J., Policastro, P., Quick, V., Lee, S.K.: "Changes in Body Weight and Fat Mass of Men and Women in the First Year of College: A Study of the 'Freshman 15.'" *Journal of American College Health*, July 1, 2006, vol. 55, no. 1, p. 41. Copyright ©2006. Reprinted by permission.

Data Set 4: Cigarette Tar, Nicotine, and Carbon Monoxide

File name: 04_CIGARET.XLS

All measurements are in milligrams per cigarette. CO denotes carbon monoxide. The king size cigarettes are nonfiltered, nonmenthol, and nonlight. The menthol cigarettes are 100 mm long, filtered, and nonlight. The cigarettes in the third group are 100 mm long, filtered, nonmenthol, and nonlight. Data are from the Federal Trade Commission. KGTAR, KGNIC, KGCO, MNTAR, MNNIC, MNCO, FLTAR, FLNIC, FLCO (where KG denotes the king size cigarettes, MN denotes the menthol cigarettes, and FL denotes the filtered cigarettes that are not menthol types).

Data Set 5: Passive and Active Smoke

File name: 05_COTININE.XLS

All values are measured levels of serum cotinine (in ng/mL), a metabolite of nicotine. (When nicotine is absorbed by the body, cotinine is produced.) Data are from the U.S. Department

of Health and Human Services, National Center for Health Statistics, Third National Health and Nutrition Examination Survey.

Data Set 6: Bears (measurements from anesthetized wild bears)

File name: 06_BEARS.XLS

AGE is in months, MONTH is the month of measurement (1 = January), SEX is coded with 1 = male and 2 = female, HEADLEN is head length (inches), HEADWTH is width of head (inches), NECK is distance around neck (in inches), LENGTH is length of body (inches), CHEST is distance around chest (inches), and WEIGHT is measured in pounds. Data are from Gary Alt and Minitab, Inc.

Data Set 7: Alcohol and Tobacco Use in Animated Children's Movies

File name: 07_CHMOVIE.XLS

Movie lengths are in minutes, tobacco use times are in seconds, and alcohol use times are in seconds. The data are based on Goldstein, Adam O., Sobel, Rachel A., Newman, Glen R.; "Tobacco and Alcohol Use in G-Rated Children's Animated Films." *Journal of the American Medical Association*, March 24/31, 1999, vol. 281, no. 12, p. 1132. Copyright © 1999. All rights reserved.

Data Set 8: Word Counts by Males and Females

File name: 08_WORDS.XLS

The columns are counts of the numbers of words spoken in a day by male (M) and female (F) subjects in six different sample groups. Column M1 denotes the word counts for males in Sample 1, F1 is the count for females in Sample 1, and so on.

Sample 1: Recruited couples ranging in age from 18 to 29

Sample 2: Students recruited in introductory psychology classes, aged 17 to 23

Sample 3: Students recruited in introductory psychology classes in Mexico, aged 17 to 25

Sample 4: Students recruited in introductory psychology classes, aged 17 to 22

Sample 5: Students recruited in introductory psychology classes, aged 18 to 26

Sample 6: Students recruited in introductory psychology classes, aged 17 to 23

Data Set 9: Movies

File name: 09_MOVIES.XLS

Movie data: title, year, rating, budge, gross, length, viewer rating.

Data Set 10: NASA space Transport System Data

File name: 10_NASA.XLS

Data Set 11: Forecast and Actual Temperatures

File name: 11_WEATHER.XLS

Forecast and actual temperatures.

Data Set 12: Electricity Consumption of a Home

File name: 12_ELECTRIC.XLS

Data Set 13: Voltage Measurements from a Home

File name: 13_VOLTAGE.XLS

All measurements are from the author's home. The voltage measurements are from the electricity supplied directly to the home, an independent Generac generator (model PP 5000), and an uninterruptible power supply (APC model CS 350) connected to the home power supply.

Data Set 14: Rainfall (in inches) in Boston for One Year

File name: 14_BOSTRAIN.XLS

Weekly rainfall in Boston.

Data Set 15: Old Faithful Geyser

File name: 15_OLDFAIRTH.XLS

Old Faithful geyser: duration, interval, and height.

Data Set 16: Car Measurements

File name: 16_CARS.XLS

The data are measurements from cars that have automatic transmissions and were manufactured in the same recent year. WT is weight (lb), LN is length (inches), BRK is braking distance (feet) from 60 mi/h, CYL is the number of cylinders, DISP is the engine displacement (liters), CITY is the fuel consumption (mi/gal) for city driving conditions, HWY is the fuel consumption (mi/gal) for highway driving conditions, and GHG is a measure of greenhouse gas emissions (in tons/year, expressed as CO₂ equivalents).

Data Set 17: Cola Weights and Volumes

File name: 17_COLA.XLS

Weights are in pounds and volumes are in ounces.

Data Set 18: M&M Plain Candy Weights (grams)

File name: 18_M&M.XLS

Weights of M&M candies.

Data Set 19: Screw Lengths (inches)

File name: 19_SCREW.XLS

Data Set 20: Coin Weights (grams)

File name: 20_COINS.XLS

The “pre-1983 pennies” were made after the Indian and wheat pennies, and they are 97% copper and 3% zinc. The “post-1983 pennies” are 3% copper and 97% zinc. The “pre-1964 silver quarters” are 90% silver and 10% copper. The “post-1964 quarters” are made with a copper-nickel alloy.

Data Set 21: Axial Loads of Aluminum Cans

File name: 21_CANS.XLS

Axial loads are measured in pounds. Axial loads are applied when the tops are pressed into place.

Data Set 22: Weights of Discarded Garbage for One Week

File name: 22_GARBAGE.XLS

Weights are in pounds. HHSIZE is the household size. Data provided by Masakuza Tani, the Garbage Project, University of Arizona.

Data Set 23: Home Sales

File name: 23_HOME.XLS

Homes sold in Dutchess country.

Data Set 24: FICO Credit Rating Scores

File name: 24_FICO.XLS