TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN/ĐỒ ÁN GIỮA KÌ MÔN XÁC SUẤT THỐNG KÊ ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

**Xác Suất Thống Kê Ứng Dụng Công Nghệ Thông Tin**

*Người hướng dẫn*: **CÔ NGUYỄN THỊ HUỲNH TRÂM**

*Người thực hiện*: **NGUYỄN TẤN THÀNH – 52100841**

Lớp **: 21050201**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN/ĐỒ ÁN GIỮA KÌ MÔN XÁC SUẤT THỐNG KÊ ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

**Xác Suất Thống Kê Ứng Dụng Công Nghệ Thông Tin**

Người hướng dẫn: **CÔ NGUYỄN THỊ HUỲNH TRÂM**

Người thực hiện: **NGUYỄN TẤN THÀNH - 52100841**

Lớp **: 21050201**

Khoá  **: 25**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

LỜI CẢM ƠN

Để hoàn thành được bài báo cáo giữa kỳ này ngoài sự nỗ lực và cố gắng của bản thân ra còn có sự hỗ trợ đồng hành đáng trân quý của cô Nguyễn Thị Huỳnh Trâm. Trong suốt quá trình học tập cô luôn tìm kiếm và áp dụng phương pháp giảng dạy tốt nhất để sinh viên chúng em tiếp thu sâu hơn về kiến thức đó,điều này như là một hành trang quý báu giúp em và các bạn có đầy đủ kiến thức để thực hiện được bài báo cáo như ngày hôm nay. Bên cạnh đó em xin gửi lời cảm ơn sâu sắc đến tất cả quý thầy, cô trong khoa Công Nghệ Thông Tin, Trường Đại Học Tôn Đức Thắng đã tận tình giúp đỡ và tạo điều kiện cho em hoàn thành bài báo cáo cuối kỳ này.

Lời cuối cùng, em xin kính chúc toàn thể quý thầy, cô tràn đầy sức khỏe và gặt hái được

nhiều thành công vang dội trong sự nghiệp cao quý của mình.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi và được sự hướng dẫn của Cô Nguyễn Thị Huỳnh Trâm. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 9 tháng 11 năm 2022*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Thành*

*Nguyễn Tấn Thành*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Sau đây là bài báo cáo giữa kì môn xác suất thống kê ứng dụng công nghệ thông

tin.

Sau quá trình nghiên cứu và tìm hiểu bài báo cáo mà giảng viên đã đưa cho, thì

tôi đã hiểu biết nhiều hơn về phương pháp thống kê để phân tích dữ liệu và tóm

tắt định lượng cho tập hợp dữ liệu thực nghiệm hoặc nghiên cứu nhất định nhằm

phục vụ cho quá trình phân tích, dự đoán và quyết định. Đồng thời cũng hiểu rõ

về việc xử lý ảnh diễn ra như thế nào, hiểu rõ them được nhiều thật ngữ, nội dung

mới liên quan đến những vấn đề trên.

Tuy nhiên vẫn còn tồn tại một số sai sót trong bài báo cáo này kính mong quý

thầy cô vui lòng cảm thông và góp ý để em có thể hoàn thiện hơn bài báo cáo giữa

kỳ này.

Mục Lục

[CHƯƠNG 1 – STATISTICS LIBRARY IN PYTHON 2](#_Toc118919506)

[1.1 Overview about statistics library in Python 2](#_Toc118919507)

[1.1.1 Khái quát về thống kê 2](#_Toc118919508)

[1.1.2 Tại sao lại sử dụng thư viện statistics trong Python 2](#_Toc118919509)

[1.1.3 Khái quát về thư viện statistics trong Python 2](#_Toc118919510)

[1.2 Functions in the library statistics 3](#_Toc118919511)

[*1.2.1 Measure of central tendency* 3](#_Toc118919512)

[1.2.2 Measure of variability 13](#_Toc118919513)

[1.2.3 Measure of correlation between pairs of data 16](#_Toc118919514)

[1.3 My code 18](#_Toc118919515)

[CHƯƠNG 2 – HISTOGRAM EQUALIZATION ALGORITHM 22](#_Toc118919516)

[2.1 Histogram equalization . 22](#_Toc118919517)

[2.2 Tại sao lại cần đến thuật toán cân bằng biểu đồ mức xàm 22](#_Toc118919518)

[2.3 Phương pháp cân bằng biểu đồ mức xám 22](#_Toc118919519)

[2.3.1 Cách tính histogram 22](#_Toc118919520)

[2.3.2 Ví dụ Histogram 23](#_Toc118919521)

[2.3.1 Cân bằng histogram 24](#_Toc118919522)

[2.3.4 Ví dụ cân bằng Histogram 25](#_Toc118919523)

[CHƯƠNG 3 – IMPLEMENTATION 27](#_Toc118919524)

[3.1 Import libarary 27](#_Toc118919525)

[3.2 Show image and Plot histogram 27](#_Toc118919526)

[3.3 Caculate histogram 28](#_Toc118919527)

[3.4 Equal histogram 28](#_Toc118919528)

[3.5 Experimental results 29](#_Toc118919529)

[TÀI LIỆU THAM KHẢO 31](#_Toc118919530)

CHƯƠNG 1 – STATISTICS LIBRARY IN PYTHON

1.1 Overview about statistics library in Python

Trước khi bắt đầu tìm hiểu về thư viện statistics trong Python thì trước hết chúng ta hãy cùng bắt đầu tìm hiểu về thống kê mô tả là gì?

1.1.1 Khái quát về thống kê

Thống kê mô tả là sự mô tả và tóm tắt dữ liệu:

Có hái cách tiếp cận chính:

* Phương pháp định lượng mô tả tóm tắt dữ liệu bằng số.
* Phương pháp tiếp cận trực quan minh họa dữ liệu bằng biểu đồ, đồ thị, biểu đồ và các đồ thị khác.

Phân tích số liệu thống kê giúp chúng ta hiểu được toàn bộ thông tin.

1.1.2 Tại sao lại sử dụng thư viện statistics trong Python

Trong quá trình thống kê thay vì duyệt qua các data point riêng lẻ, thì chỉ cần nhìn vào giá trị trung bình hoặc phương sai của bộ dữ liệu là có thể biết được xu hướng và các đặc trưng mà chúng ta có thể đã bỏ qua khi quan sát tất cả các dữ liệu ở định dạng thô.

1.1.3 Khái quát về thư viện statistics trong Python

Module này không phải sinh ra để cạnh tranh và trở thành đối thủ với các thư viện bên thứ 3 như Numpy, SciPy,…. Nó hướng đến trình độ vẽ đồ thị và máy tính khoa học.

Khi sử dụng các hàm thống kê, ta sẽ thực hiện chúng trên một sample hoặc một population.

Thư viện statistics cung cấp cho ta một cách tiếp cận một số vấn đề tính toán như giá trị trung bình, trung vị, yếu vị, phương sai, và độ lệch chuẩn. với các hàm như mean(), media(), mode(),….

1.2 Functions in the library statistics

Các loại thống kê mô tả:

* Measure of central tendency (Đo lường xu hướng trung tâm)
* Measure of variability (Đo lường sự thay đổi)
* Satistics for relations between two inputs(Thống kê cho mối quan hệ giữa hai đầu vào)

### *1.2.1 Measure of central tendency*

The measure of central tendency (thước đo xu hướng trung tâm) là một giá trị duy nhất cố gắng mô tả toàn bộ dữ liệu. Dựa trên ba xu hướng là số trung bình, số trung vị và và số yếu vị. Đây là các số có tần số xuất hiện nhiều nhất trong mẫu.

* FUNCTION MEAN():

Mean hay trung bình số học là trung bình cộng của tất cả các mục trong một tập dữ liệu. Xem như giá trị trung bình của tập dữ liệu là 𝑥 được biểu thị bằng toán học là Σᵢ𝑥ᵢ / 𝑛, trong đó 𝑖 = 1, 2,…, 𝑛. Nói cách khác, nó là tổng của tất cả các phần tử 𝑥ᵢ chia cho số mục trong tập dữ liệu 𝑥.

* Cú pháp: statistics.mean(data)
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu trả về : hàm mean(data) trả về giá trị trung bình của bộ dữ liệu được truyền trong đối số của nó. Nếu đối số là trống thì sẽ xuất hiện StatisticsError.
* Code mẫu :

from statistics import mean

from fractions import Fraction as fr

data1 = (11, 3, 4, 5, 7, 9, 2)

data2 = (-1, -2, -4, -7, -12, -19)

data4 = (fr(1, 2), fr(44, 12), fr(10, 3), fr(2, 3))

data5 = {1:"one", 2:"two", 3:"three"}

print("Mean of data set 1 is % s" % (mean(data1)))

print("Mean of data set 2 is % s" % (mean(data2)))

print("Mean of data set 4 is % s" % (mean(data4)))

print("Mean of data set 5 is % s" % (mean(data5)))

# OUTPUT:

Mean of data set 1 is 5.857142857142857

Mean of data set 2 is -7.5

Mean of data set 4 is 49/24

Mean of data set 5 is 2

* FUNCTION FMEAN():

Hàm fmean() khá là giống hàm mean() nhưng có một số khác biệt sau.

Hàm fmean() được giới thiệu trong Python 3.8 như một sự bổ sung thay thế nhanh hơn cho hàm mean().

Chuyển đổi dữ liệu thành float và tính giá trị trung bình của số học. Nó luôn trả về một số dấu phẩy động.

* Cú pháp: statistics.fmean(data,weights = None).
* Đầu vào:
* data là bộ dữ liệu (sequence hoặc iterable).
* Weights là trọng số. Thay vì mỗi điểm dữ liệu đóng góp như nhau vào giá trị trung bình cuối cùng, một số điểm dữ liệu đóng góp nhiều “trọng số” hơn những điểm khác. Nếu tất cả các trọng số bằng nhau, thì giá trị trung bình có trọng số sẽ bằng trung bình cộng ("trung bình" thông thường mà bạn quen dùng).
* Kiểu trả về :
* hàm mean(data) trả về giá trị trung bình của bộ dữ liệu được theo kiể float.
* Nếu trọng số được cung cấp, nó phải có cùng độ dài với dữ liệu nếu không ValueError sẽ xuất hiện.
* Code mẫu :

Ta có bài toán sau : một giáo sư ấn định điểm cho một khóa học bằng cách đánh giá các câu đố ở mức 20%, bài tập về nhà là 20%, bài kiểm tra giữa kỳ là 30% và bài kiểm tra cuối kỳ là 30%:

import statistics

grades = [85, 92, 83, 91]

weights = [0.20, 0.20, 0.30, 0.30]

statistics.fmean(grades,weights) #return 87.6

* FUNCTION GEOMETRIC\_MEAN():

Rất hữu ích khi muốn biết xu hướng trung tâm hoặc giá trị điển hình của một tập hợp số bằng cách sử dụng tích các giá trị của chúng. Giá trị trung bình hình học được định nghĩa là căn thứ n của tích n số, tức là đối với một tập hợp các số x1, x2, ..., xn, trung bình hình học được định nghĩa là: Diagram

Description automatically generated

Rất hữu ích để tính trung bình của các tỉ lệ.

* Cú pháp: statistics.geometric\_mean (data).
* Đầu vào:
  + - là bộ dữ liệu (sequencehoặc iterable)
    - Nếu tập dữ liệu đầu vào trống, hoặc chứa số 0 hoặc nếu nó chứa giá trị âm thì  StatisticsError sẽ được gọi đến.
* Kiểu trả về : Giá trị trung bình hình học của dữ liệu.
* Code mẫu :

import statistics

data = [54, 24, 36]

result = statistics.geometric\_mean(data)

print(result) # 36.000000000000014

* FUNCTION HARMONIC\_MEAN():

Thường được sử dụng trong các tình huống khi tỷ giá trung bình được mong muốn. Trung bình điều hòa cũng là nghịch đảo của trung bình cộng của các nghịch đảo của một tập hợp quan sát nhất định. Ví dụ, trung bình điều hòa của ba giá trị a, b và c sẽ tương đương với 3 / (1 / a + 1 / b + 1 / c). Nếu một trong các giá trị là 0, kết quả sẽ là 0.

Nó thường thích hợp khi lấy tỷ lệ hoặc tỷ lệ trung bình, ví dụ như tốc độ.

* Cú pháp: statistics.harmonic\_mean (data,weights = None).
* Đầu vào:
  + data là bộ dữ liệu (sequence hoặc iterable, list).
  + Weight là trọng số. Để hiểu về nó chúng ta hãy xét ví dụ bên dưới.
* Kiểu trả về :
  + hàm trả về giá trị trung bình của bộ dữ liệu.
  + StatisticsError khi một tập dữ liệu trống hoặc nếu tập dữ liệu bao gồm các giá trị âm.
  + TypeError cho tập dữ liệu của các giá trị kiểu không phải số.
* Code mẫu:

# Import statistics Library

import statistics

# Calculate harmonic mean

print(statistics.harmonic\_mean([40, 60, 80])) #55.38461538461538

print(statistics.harmonic\_mean([10, 30, 50, 70, 90])) #27,97513321492007

* KẾT LUẬN CHUNG VỀ CÁC HÀM MEAN (GIÁ TRỊ TRUNG BÌNH):
* Mean rất dễ bị “lệch” khi dữ liệu có những giá trị rất khác biệt so với các giá trị còn lại, ví dụ số 100 trong data ở trên là một giá trị rất khác. Bởi thế, nếu chỉ dùng mean để đưa ra “cảm nhận" về điểm trung tâm của bộ dữ liệu sẽ không được chính xác. Người ta đưa ra thêm khái niệm median để tính điểm trung tâm theo một cách khác, mà giá trị thu được sẽ biểu diễn điểm trung tâm một cách “tốt hơn”.
* Giá trị trung bình bị ảnh hưởng mạnh bởi các giá trị ngoại lai và không nhất thiết phải là ví dụ điển hình của các điểm dữ liệu.
* Để có thước đo xu hướng trung tâm mạnh mẽ hơn, mặc dù kém hiệu quả hơn, Chúng ta hãy cùng tìm hiểu về trung vị xem trung vị ().
* FUNCTION MEDIAN():

Nó sẽ trả về giá trị trung vị của dữ liệu số cho trước bằng cách tính trung bình của hai điểm giữa nếu cần.

Median được tính như sau: Sắp xếp dữ liệu và lấy giá trị ở giữa. Nếu số data point là lẻ, nó sẽ trả về điểm giữa. Nếu số data point là chẵn, nó sẽ trả về trung bình của hai giá trị trung vị.

Trung vị là một thước đo mạnh mẽ về vị trí trung tâm và ít bị ảnh hưởng bởi sự hiện diện của các điểm ngoại lệ.

* Cú pháp: statistics.median(data).
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu dữ liệu trả về :
  + - trả vị giá trị trung vị của bộ dữ liệu.
    - Nếu dữ liệu trống, StatisticsError sẽ xuất hiện.
* Code mẫu :

import statistics

# Calculate middle values

print(statistics.median([1, 3, 5, 7, 9, 11])) # 6.0

print(statistics.median([-11, 5.5, -3.4, 7.1, -9, 22])) # 1.05

* Note: Vấn đề với hàm median() là giá trị cuối cùng có thể không phải là một data point thật sự khi số data point là chẵn. Trong những trường hợp như vậy, bạn có thể sử dụng median\_low() hoặc median\_high() để tính toán trung vị. Với số data point chẵn, các hàm này sẽ trả về giá trị nhỏ hơn và lớn hơn của hai điểm giữa tương ứng. Tiếp theo chúng ta sẽ cùng tìm hiểu về chúng.
* FUNCTION MEDIAN\_LOW():

Hàm median\_low () trả về giá trị trung bình của dữ liệu trong trường hợp số phần tử lẻ, nhưng trong trường hợp số phần tử chẵn, trả về giá trị thấp hơn của hai phần tử ở giữa.

Sử dụng trung vị thấp khi dữ liệu của bạn rời rạc và bạn muốn trung vị là một điểm dữ liệu thực tế hơn là nội suy.

* Cú pháp: statistics.median\_low(data).
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu dữ liệu trả về :
  + - trả vị giá trị trung vị của bộ dữ liệu.
    - Nếu dữ liệu trống, StatisticsError sẽ xuất hiện.
* Code mẫu :

import statistics

# Calculate the low middle values

print(statistics.median\_low([1, 3, 5, 7, 9, 11, 13])) # 7

print(statistics.median\_low([1, 3, 5, 7, 9, 11])) # 5

print(statistics.median\_low([-11, 5.5, -3.4, 7.1, -9, 22])) # -3.4

* FUNCTION MEDIAN\_HIGH():

Hàm median\_high () trả về giá trị trung bình của dữ liệu trong trường hợp số phần tử lẻ, nhưng trong trường hợp số phần tử chẵn, trả về giá trị cao hơn của hai phần tử ở giữa.

Sử dụng trung vị cao khi dữ liệu của bạn rời rạc và bạn muốn trung vị là một điểm dữ liệu thực tế hơn là nội suy.

* Cú pháp: statistics.median\_high(data).
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu dữ liệu trả về :
  + - trả vị giá trị trung vị của bộ dữ liệu.
    - Nếu dữ liệu trống, StatisticsError sẽ xuất hiện.
* Code mẫu :

import statistics

# Calculate the high middle values

print(statistics.median\_high([1, 3, 5, 7, 9, 11, 13])) # 7

print(statistics.median\_high([1, 3, 5, 7, 9, 11])) # 7

print(statistics.median\_high([-11, 5.5, -3.4, 7.1, -9, 22])) # 5.5

* FUNCTION MEDIAN\_GROUPED():

Dữ liệu được giả định là được nhóm lại thành các khoảng có khoảng chiều rộng. Mỗi điểm dữ liệu trong mảng là điểm giữa của khoảng chứa giá trị thực. Giá trị trung bình được tính bằng cách nội suy trong khoảng trung vị (khoảng chứa giá trị trung vị), giả sử rằng các giá trị thực trong khoảng đó được phân phối đồng nhất.

* Công thức toán học cho Trung vị được nhóm là: GMedian = L + khoảng \* (N / 2 - CF) / F.
  + - L = Giới hạn dưới của khoảng trung bình ( khoảng chiều rộng )
    - N = Tổng số điểm dữ liệu
    - CF = Số điểm dữ liệu dưới khoảng trung bình
    - F = Số điểm dữ liệu trong khoảng trung bình
* Cú pháp: statistics.median\_grouped(data, interval = 1)
* Đầu vào:
  + - là bộ dữ liệu (sequence hoặc iterable).
    - interval (1 by default) :Xác định độ rộng của dữ liệu được nhóm và thay đổi. Nó cũng sẽ thay đổi nội suy của giá trị trung vị được tính toán.
* Kiểu dữ liệu trả về :
  + - Trả về giá trị trung bình của dữ liệu liên tục được nhóm lại, được tính bằng phân vị thứ 50.
    - Nếu dữ liệu trống, StatisticsError sẽ xuất hiện.
* Code mẫu :

import statistics

# Calculate the median of grouped continuous data

print(statistics.median\_grouped([1, 2, 3, 4])) #2.5

print(statistics.median\_grouped([1, 2, 3, 4, 5])) #3.0

print(statistics.median\_grouped([1, 2, 3, 4], 2)) #2.0

print(statistics.median\_grouped([1, 2, 3, 4], 3)) #1.5

print(statistics.median\_grouped([1, 2, 3, 4], 5)) #0.5

* KẾT LUẬN:

Mean hay Median là một chỉ số xác định giá trị trung bình rất tốt, nhưng một vài giá trị phân cực có thể dẫn đến một trung bình xa điểm giữa thật sự. Trong một số trường hợp, cần xác định data-point thường xuyên nhất trong một tập dữ liệu.

* FUNCTION MODE():

Hàm mode() hay yếu vị sẽ trả về data point phổ biến nhất từ ​​số rời rạc cũng như dữ liệu không phải là số. Đây là hàm thống kê duy nhất có thể được sử dụng với dữ liệu không phải là số.

* Cú pháp: statistics.mode(data).
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable hoặc String).
* Kiểu dữ liệu trả về :
  + - Trả về điểm dữ liệu phổ biến nhất từ dữ liệu rời rạc hoặc danh nghĩa.
    - Nếu dữ liệu trống, StatisticsError sẽ xuất hiện.
* Code mẫu :

import statistics

# Calculate the mode

print(statistics.mode([1, 3, 3, 3, 5, 7, 7, 9])) # 3

print(statistics.mode([1, 1, -3, 3, 7, -9])) # 1

print(statistics.mode(['red', 'green', 'blue', 'red'])) # red

* FUNCTION MULTIMODE():

Đôi khi chúng ta có thể có nhiều hơn 1 yếu vị. Tình huống này được gọi là đa yếu vị.

* Cú pháp: statistics.multimode(data).
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable hoặc String).
* Kiểu dữ liệu trả về : Trả lại danh sách các giá trị xuất hiện thường xuyên nhất theo thứ tự lần đầu tiên chúng gặp trong dữ liệu. Sẽ trả về nhiều hơn một kết quả nếu có nhiều yếu vị hoặc danh sách trống nếu dữ liệu trống.
* Code mẫu :

import statistics

# Calculate the mode

print(statistics.multimode([1, 3, 3, 3, 3, 5,7,7, 7, 7, 9])) # [3, 7]

print(statistics.multimode(['red', 'green', 'blue', 'red', ‘green’])) # [‘red’, ‘green’]

* FUNCTION QUANTILES()

Chia dữ liệu thành các khoảng với xác suất bằng nhau.

Phương thức Statistics.quantiles () trong Python được sử dụng để trả về các lượng tử tương ứng với các số n chứa trong dữ liệu có thể lặp lại.

Một lượng tử xác định một tập dữ liệu cụ thể. Nói cách khác, một lượng tử xác định có bao nhiêu giá trị trong một phân phối nằm trên hoặc dưới một giới hạn nhất định. Một lượng tử có thể là một phần tư, phần trăm và ngũ phân vị

* Cú pháp: statistics.quantiles(data,\*,n = 4)
* Đầu vào:
  + - là bộ dữ liệu (sequence hoặc iterable).
    - n: Số lượng tử bạn muốn. Tham số này nhận một giá trị số nguyên, theo mặc định là 4. Đây là tùy chọn.
    - Method: Phương pháp tính toán các lượng tử; exclusive hoặc inclusive. exclusive là mặc định.
      * Exclusive: được sử dụng cho dữ liệu được lấy mẫu từ một tập hợp có thể có nhiều giá trị cực đoan hơn được tìm thấy trong các mẫu.
      * Inclusive: được sử dụng để mô tả dữ liệu dân số hoặc cho các mẫu được biết là bao gồm các giá trị cực đoan nhất từ ​​tổng thể. Giá trị nhỏ nhất trong dữ liệu được coi là phân vị thứ 0 và giá trị lớn nhất được coi là phân vị thứ 100.
* Kiểu dữ liệu trả về : trả về một danh sách chứa các giá trị số của n-1 lượng tử phía trên.
* Code mẫu :

import statistics

data1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

# to return the upper three quartiles

third\_quartiles = statistics.quantiles(data1, n=4)

print('The third quartile of the given data set is', third\_quartiles)

#Output:

The third quartile of the given data set is [4.0, 8.0, 12.0]

1.2.2 Measure of variability

Measure of variability (độ đo lệch dữ liệu) Xác định có bao nhiêu data point lệch khỏi giá trị điển hình hoặc trung bình của bộ dữ liệu cũng quan trọng không kém tính giá trị trung tâm hoặc trung bình.

Phép đo độ biến thiên được gọi là mức độ lan truyền của dữ liệu hoặc mức độ phân phối dữ liệu của chúng tôi. Các biện pháp thay đổi phổ biến nhất là: range, variance, Standard deviation.

* FUNCTION VARIANCE()

Trả về phương sai mẫu của dữ liệu, có thể lặp lại của ít nhất hai số có giá trị thực. Phương sai, hay thời điểm thứ hai về giá trị trung bình, là thước đo tính biến thiên (trải rộng hoặc phân tán) của dữ liệu. Một phương sai lớn chỉ ra rằng dữ liệu được trải rộng; một phương sai nhỏ cho thấy nó được tập hợp chặt chẽ xung quanh giá trị trung bình.

Sử dụng khi dữ liệu là một mẫu từ một tập hợp .

* Cú pháp: statistics.variance (data, xbar = None).
* Đầu vào:
  + - là bộ dữ liệu (sequence hoặc iterable).
    - Xbar là giá trị trung bình của dữ liệu. mặc định nó sẽ tự tính mean.
  + Kiểu dữ liệu trả về :
    - Trả về phương sai mẫu của dữ liệu.
    - Lỗi khi dữ liệu có nhiều hơn hai giá trị.
* Code ở phần dưới .
* FUNCTION PVARIANCE()

Trả về phương sai tổng thể của dữ liệu, một chuỗi không rỗng hoặc có thể lặp lại của các số có giá trị thực. Phương sai, hay thời điểm thứ hai về giá trị trung bình, là thước đo tính biến thiên (trải rộng hoặc phân tán) của dữ liệu. Một phương sai lớn chỉ ra rằng dữ liệu được trải rộng; một phương sai nhỏ cho thấy nó được tập hợp chặt chẽ xung quanh giá trị trung bình.

Tính phương sai từ toàn bộ tập hợp.

* Cú pháp: statistics.pvariance (data, mu = None).
* Đầu vào:
  + - là bộ dữ liệu (sequence hoặc iterable).
    - mu là giá trị trung bình của dữ liệu. mặc định nó sẽ tự tính mean.

Nó cũng có thể được sử dụng để tính thời điểm thứ hai xung quanh một điểm không phải là điểm trung bình.

* Kiểu dữ liệu trả về :
  + - Trả về phương sai mẫu của dữ liệu.
    - Lỗi khi dữ liệu có nhiều hơn hai giá trị.
* Code ở phần dưới.
* FUNCTION STDEV()

Để tính toán độ rõ tiêu chuẩn của tập hợp và độ lệch tiêu chuẩn của mẫu dữ liệu.

* Cú pháp: statistics.stdev (data,xbar = None)
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu dữ liệu trả về : Trả về độ lệch chuẩn của mẫu (căn bậc hai của phương sai mẫu)
* Code ở phần dưới.
* FUNCTION PSTDEV()

Để tính toán độ rõ tiêu chuẩn của tập hợp và độ lệch tiêu chuẩn của tổng thể dữ liệu.

* Cú pháp: statistics.pstdev (data,mu = None)
* Đầu vào: là bộ dữ liệu (sequence hoặc iterable).
* Kiểu dữ liệu trả về : Trả về độ lệch chuẩn của tổng thể dữ liệu (căn bậc hai của phương sai tổng thể)
* Code mẫu :

import statistics

from fractions import Fraction as F

data = [1, 2, 3, 4, 5, 6, 7, 8, 9]

statistics.pvariance(data) # returns 6.666666666666667

statistics.pstdev(data) # returns 2.581988897471611

statistics.variance(data) # returns 7.5

statistics.stdev(data) # returns 2.7386127875258306

* Như minh chứng từ ví dụ trên, phương sai nhỏ hơn hàm ý rằng nhiều data point có giá trị gần hơn với giá trị trung bình. Bạn cũng có thể tính độ lệch chuẩn của các số thập phân và phân số.

1.2.3 Measure of correlation between pairs of data

Các hàm này tính toán số liệu thống kê liên quan đến quan hệ giữa hai đầu vào.

* FUNCTION CORRELATION()

Trong lý thuyết xác suất và thống kê, hệ số tương quan (Coefficient Correlation) cho biết độ mạnh của mối quan hệ tuyến tính giữa hai biến số ngẫu nhiên.

Hệ số tương quan giữa 2 biến có thể dương (positive) hoặc âm (negative). Hệ số tương quan dương cho biết rằng giá trị 2 biến tăng cùng nhau còn hệ số tương quan âm thì nếu một biến tăng thì biến kia giảm.

* + - Positive Correlation: cả hai biến đều thay đổi theo cùng một hướng.
    - Neutral Correlation: Không có mối quan hệ nào trong sự thay đổi của các biến.
    - Negative Correlation: các biến thay đổi theo hướng ngược nhau.
  + Cú pháp: statistics.correlation(x,y)
  + Dữ liệu đầu vào: Cả hai đầu vào phải có cùng độ dài (không ít hơn hai) và không cần phải là hằng số, nếu không, Lỗi thống kê sẽ xuất hiện.
  + Kiểu trả về : Trả về hệ số tương quan của 2 tập dữ liệu.
  + Code mẫu:

import statistics

x = [1, 2, 3, 4, 5, 6, 7, 8, 9]

y = [9, 8, 7, 6, 5, 4, 3, 2, 1]

statistics.correlation(x, x) # 1.0

statistics.correlation(x, y) # -1.0

* FUNCTION COVARIANCE()

Hiệp phương sai (covariance). Nếu phương sai dùng để đo lường sự biến thiên của một biến ngẫu nhiên (hay dữ liệu trên một tập mẫu) thì hiệp phương sai đo lường sự biến thiên của hai biến ngẫu nhiên (hay dữ liệu trên hai tập mẫu cùng số cá thể).

Nó được tính bằng giá trị trung bình của sản phẩm giữa các giá trị từ mỗi mẫu, trong đó các giá trị đã được căn giữa (đã trừ giá trị trung bình của chúng).

* + Cú pháp: statistics.covariance(x,y)
  + Dữ liệu đầu vào: Cả hai đầu vào phải có cùng độ dài (không ít hơn hai) và không cần phải là hằng số, nếu không, Lỗi thống kê sẽ xuất hiện.
  + Code mẫu:

import statistics

x = [2, 3, 4, 2]

y = [3, 5, 9, 0]

print("Covariance - ", statistics.covariance(x, y))

Output:

Covariance - 3.4166666666666665

Ở đây, giá trị hiệp phương sai là dương. Điều này chỉ ra rằng cả hai biến / đầu vào ngẫu nhiên đều di chuyển theo cùng một hướng.

* FUNCTION LINEAR\_REGRESSION()

Trong một hàm tuyến tính, hồi quy tuyến tính đơn giản xác định mối liên hệ giữa một biến độc lập x và một biến phụ thuộc y. y = slope \* x + intercept

Phương thức linear\_regression () được sử dụng để lấy slope và intercept của một hồi quy tuyến tính đơn giản với các giá trị x và y bằng cách sử dụng phương pháp ước lượng bình phương nhỏ nhất thông thường.

* + Cú pháp: statistics.linear\_regression(x,y)
  + *Dữ* liệu đầu vào:
    - x: Đây là biến độc lập.
    - y: Đây là biến phụ thuộc.
  + Kiểu dữ liệu trả về : slope và intercept.
  + Code mẫu:

Ví dụ: chúng ta có thể sử dụng ngày phát hành của các bộ phim Monty Python để dự đoán số lượng tích lũy các bộ phim Monty Python sẽ được sản xuất vào năm 2019 với giả định rằng chúng đã giữ được tốc độ.

import statistics

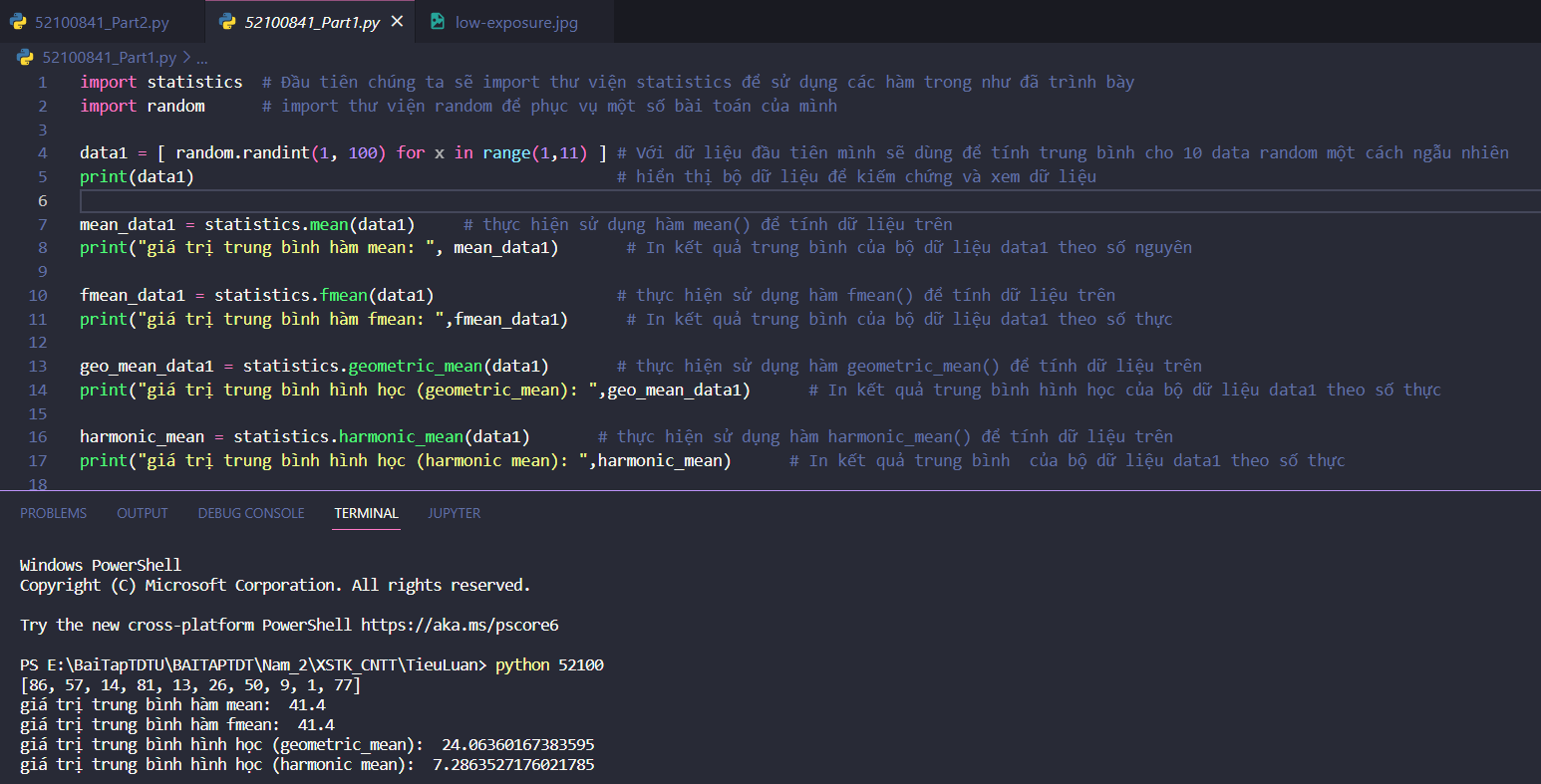
year = [1971, 1975, 1979, 1982, 1983]

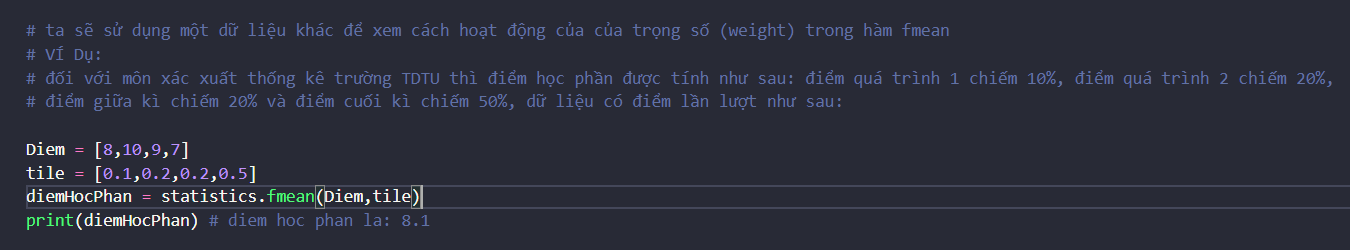
films\_total = [1, 2, 3, 4, 5]

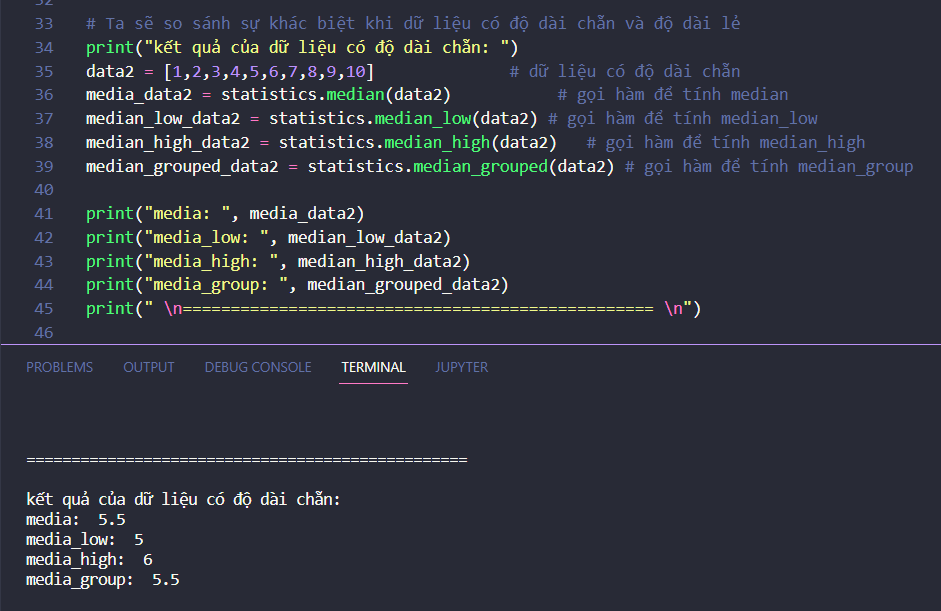
slope, intercept = linear\_regression(year, films\_total)

round(slope \* 2019 + intercept) # output: 16

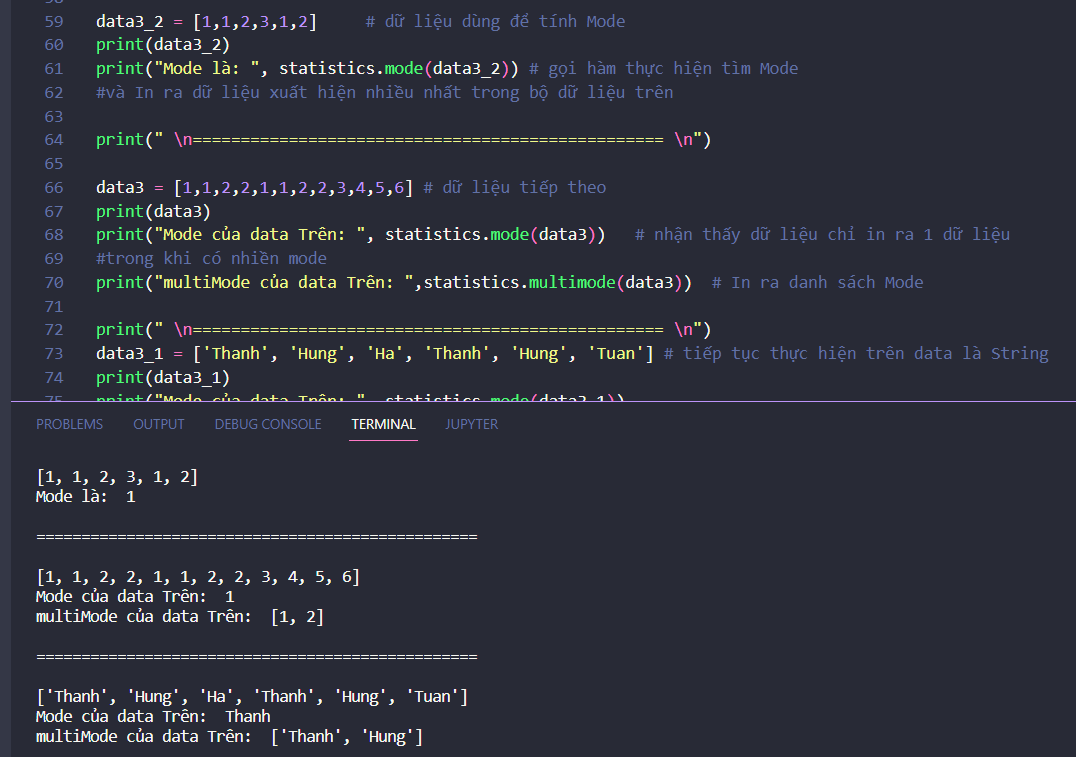
1.3 My code

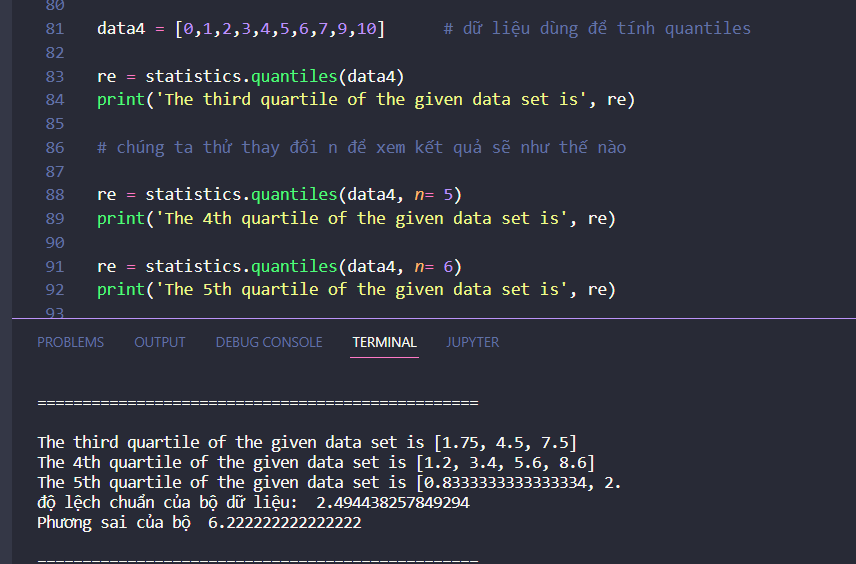


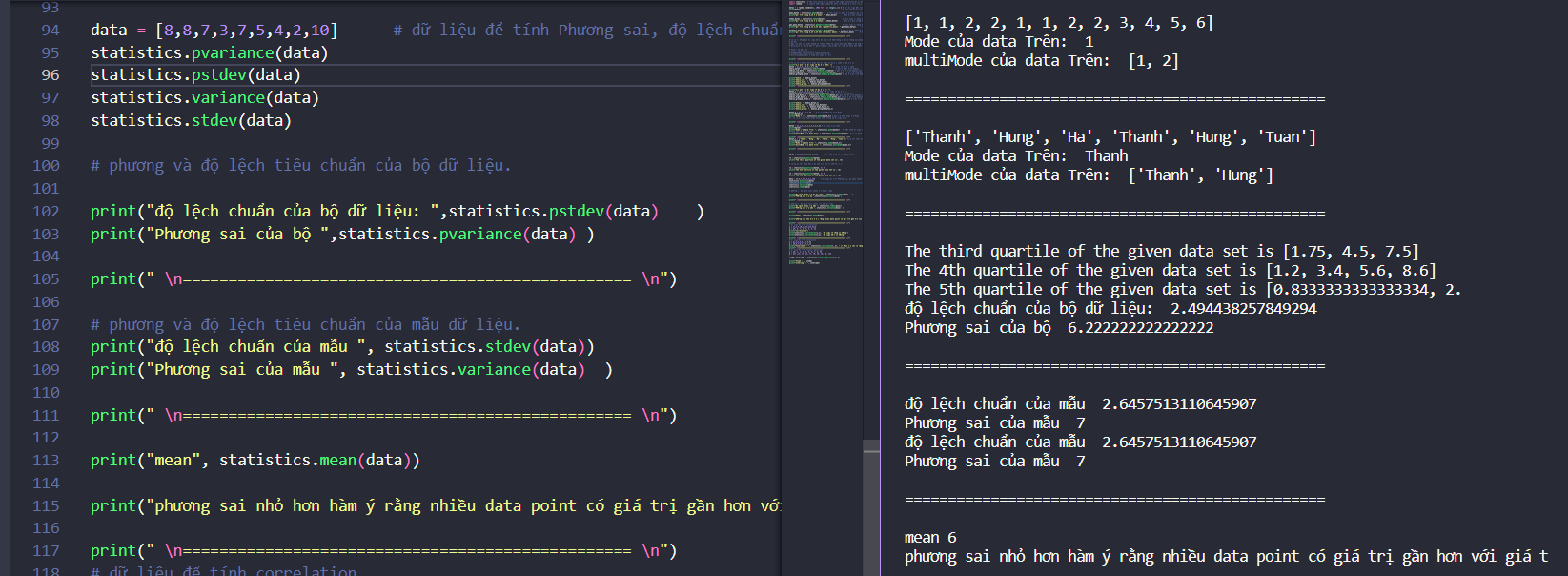


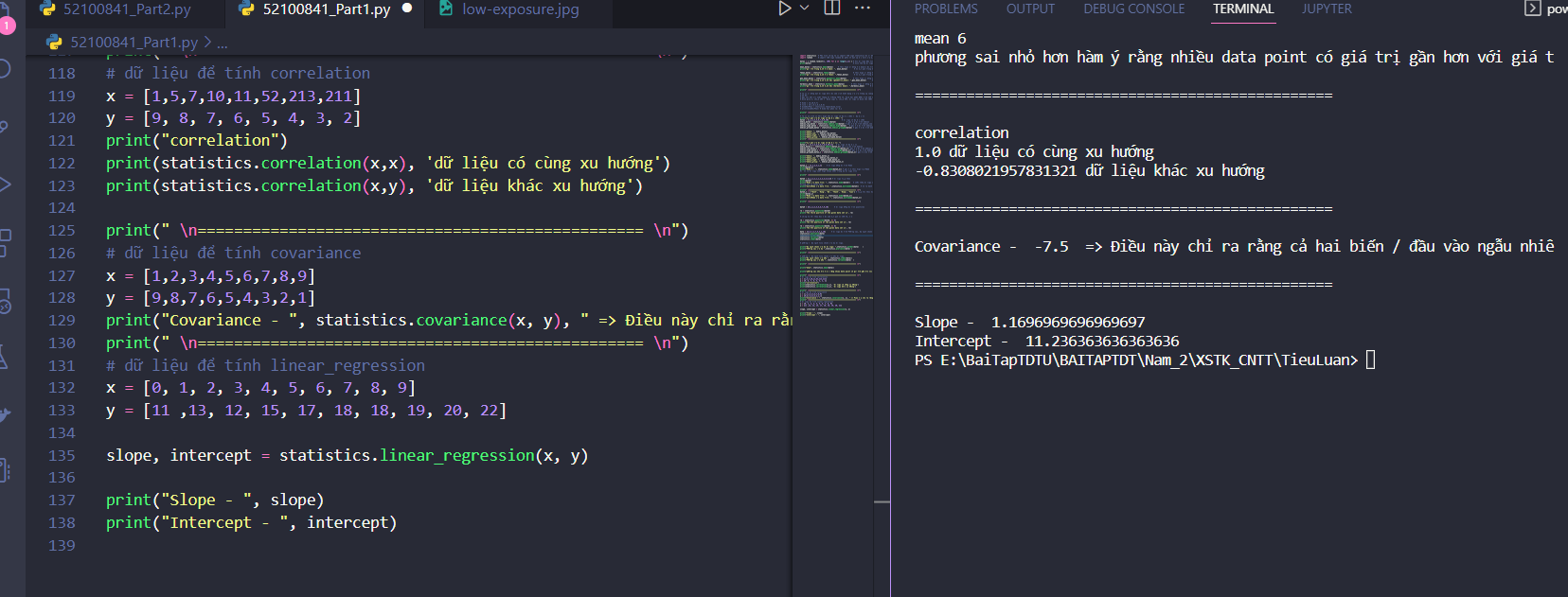












CHƯƠNG 2 – HISTOGRAM EQUALIZATION ALGORITHM

2.1 Histogram equalization .

Biểu đồ là một đồ thị thể hiện sự phân bố tần số của một tập dữ liệu. Một histogram có hai trục x và y. Trục x chứa các điểm có sự phân bố tần số. Trục y chứa các tần số. Các độ cao khác nhau của thanh cho thấy sự khác nhau về sự xuất hiện của dữ liệu. Một số ứng dụng như phân tích hình ảnh, cân bằng hình ảnh, tạo ngưỡng ảnh.

Trong lĩnh vực xử lí ảnh, histogram (lược đồ xám) là biểu đồ tần xuất được dùng để thống kê số lần xuất hiện các mức sáng trong ảnh.

Histogram equalization là một phương pháp dùng để tăng chất lượng của ảnh, cụ thể là tăng độ tương phản. Làm cho phân bố giá trị pixel không bị co cụm tại một khoảng hẹp mà được “kẽo dãn” ra.

2.2 Tại sao lại cần đến thuật toán cân bằng biểu đồ mức xàm

Trong thực tế, camera thường chịu tác động từ điều kiện sáng. Điều đó khiến cho nhiều ảnh bị tối hoặc quá sáng. Cân bằng histogram là một phương pháp tiền/hậu xử lí ảnh rất mạnh mẽ, và được ứng dụng trong computer vision.

2.3 Phương pháp cân bằng biểu đồ mức xám

2.3.1 Cách tính histogram

**-** Histogram của ảnh (Lược đồ xám) chỉ ra phân bố mức xám trên ảnh, mô tả tổng thể phân bố mức xám trên ảnh.

**-** Gọi rk là giá trị mức xám của pixel thứ k của ảnh f(x,y) có L mức xám trong khoảng [0-255].

* + Với k = 0,1,2,…L-1

Đặt h(rk) = nk; với nk là tổng số pixel có giá trị mức xám rk.

Lúc đó: h(rk) gọi là Histogram không chuẩn hóa.

Đặt p(rk) = : lược đồ xám chuẩn hóa .

* + p(rk) gọi là Histogram chuẩn hóa hay Histogram của ảnh.
  + với M là số hàng và N là số cột của ma trận ảnh.

2.3.2 Ví dụ Histogram

- Ta lần lượt có ảnh và histogram tương ứng với ảnh tối, ảnh sáng, ảnh tương phản thấp, ảnh tương phản cao.

Diagram

Description automatically generated with medium confidence

* Nhận xét:
  + Ảnh tối thì Histogram phân bố về phía trái (về pixel có mức xám = 0)
  + Ảnh sáng thì Histogram phân bố về phía phải (về pixel có mức xám = 255)
  + Ảnh độ tương phản thấp thì Histogram phân bố không đều
  + Ảnh có độ tương phản cao thì Histogram phân bố đều
* Sự dụng kỹ thuật thay đổi Histogram của ảnh để tăng cường ảnh

2.3.1 Cân bằng histogram

Trong trường hợp ảnh là một hàm liên tục.

* Gọi r là cấp độ ảnh xám cần được cân bằng 0 <= r <= L – 1. Với r = 0 là ảnh tối, r = L -1 là ảnh.
* Gọi s = T(r) là một ánh xạ, khi đó mỗi điểm ảnh có giá trị r trong ảnh gốc được ánh xạ tạo nên cấp độ s cho ảnh đầu ra, thỏa mãn :
  + T(r) đơn điệu tăng trong khoảng 0 <= r <= L – 1.
  + 0 <= T(r) <= L – 1 với 0 <= r <= L – 1.
* Gọi pr(r) và ps(s) lần lượt biểu diễn hàm mật độ xác suất của ảnh gốc và ảnh đầu ra

Mục tiêu là cần tìm hàm biến đổi T() : s = T(r) sao cho : pr(rk) -> ps(sk) với hàm T() Phân bố đều.

* + Ta có thể suy ra hàm mật độ xác xuất ps(s) = pr(r)
  + Text, letter

    Description automatically generatedHàm phân phối tích lũy của biến ngẫu nhiên r, hàm mật độ sác xuất s, xác định bởi cấp độ xám ảnh đầu vào. s = T(r) = L – 1
  + Tương tự, tích phân của hàm mật độ xác suất trong khoảng [0,L−1] :

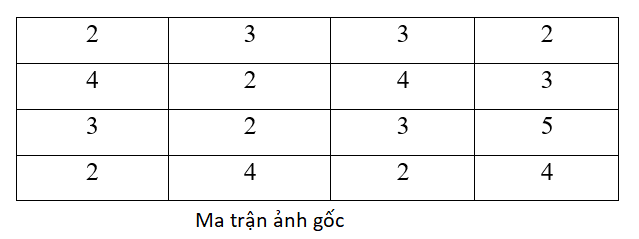
Các Bước Làm:

* Bước 1: Tính xác suất giá trị mức xám rk có trong ảnh là pr(rk)=h(rk) / (M \* N)
  + Trong đó:
    - h(rk) = nk là tổng số pixel có giá trị mức xám rk
    - M \* N là tổng số pixel trong ảnh
* Bước 2: Tính hàm mật độ xác suất
  + sk = (L-1) \* với k = 0,1,2,…L-1
* Bước 3: Làm tròn sk
  + Ảnh thu được từ sk gọi là ảnh cân bằng histogram
  + Giá trị làm tròn sk gọi là mức xám của pixel thứ k trong ảnh cân bằng histogram
* Text

  Description automatically generated with low confidenceTrên thực thế giá trị mức xám được xác định như sau :

2.3.4 Ví dụ cân bằng Histogram

* Bước 1: vẽ biểu đồ vùng ảnh



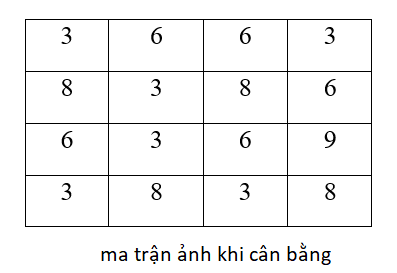
Chart, histogram

Description automatically generated

* Bước 2: Lập bảng với L = 10 là số mức xám mới, n = 16 là tổng pixel của vùng ảnh.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Mức xám rk | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Số pixel nk | 0 | 0 | 6 | 5 | 4 | 1 | 0 | 0 | 0 | 0 |
| với j = 0,…k | 0 | 0 | 6 | 11 | 15 | 1  6 | 1  6 | 1  6 | 1  6 | 16 |
| sk = )/n với j = 0,…k | 0 | 0 | 6/16 | 11/16 | 15/16 | 1 | 1 | 1 | 1 | 1 |
| Kết quả sk = round(sk\*(L-1)) | 0 | 0 | 3 | 6 | 8 | 9 | 9 | 9 | 9 | 9 |

Bước 3: Thay thế mức xám rk trong vùng ảnh ban đầu thành mức xám sk tương ứng và vẽ biểu đồ Histogram thu được:

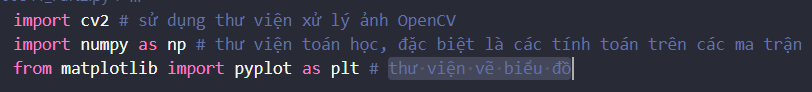


Chart, histogram, box and whisker chart

Description automatically generated

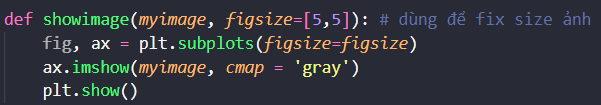
CHƯƠNG 3 – IMPLEMENTATION

* 1. Import libarary
* Đầu tiên ta sẽ import các thư viện cần thiết để thực hiện một số chức năng sau :
* OpenCV2 để sử dụng thư viện xử lý ảnh.
* Numpy thư viện toán học, đặc biệt là các tính toán trên các ma trận.
* Matplitlib thư viện vẽ biểu đồ.

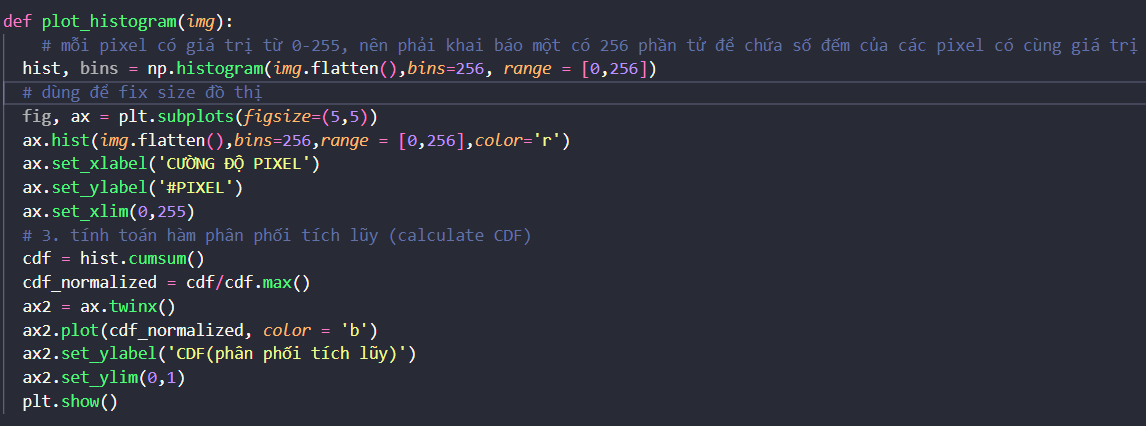


3.2 Show image and Plot histogram

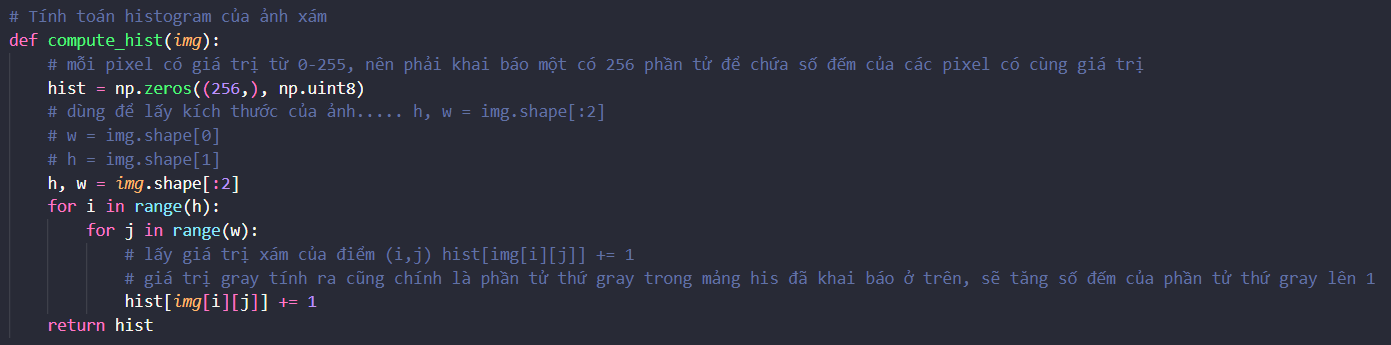
* Dùng để hiện thị ảnh cho người dùng xem .
* Figsize cố định kích thước hình ảnh
* Imshow() hiển thị hình ảnh myimage và ảnh xám.



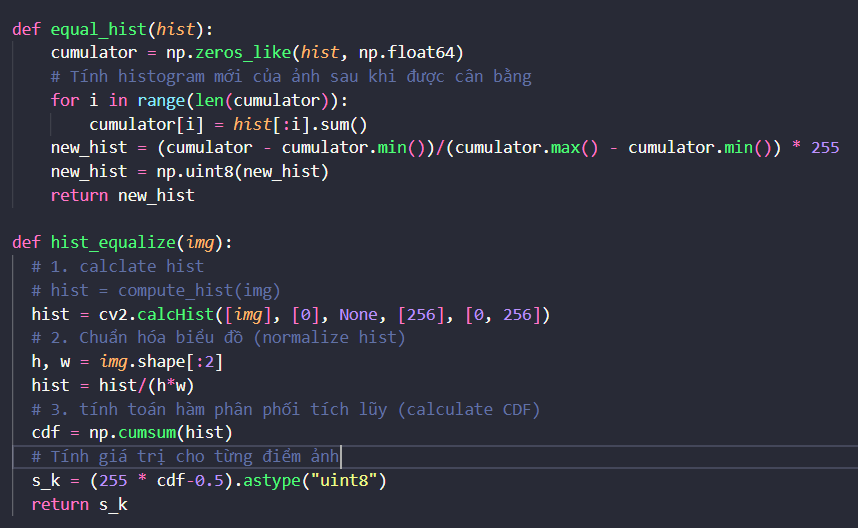
* Dùng để hiện thị biểu đồ cân bằng.



* 1. Caculate histogram



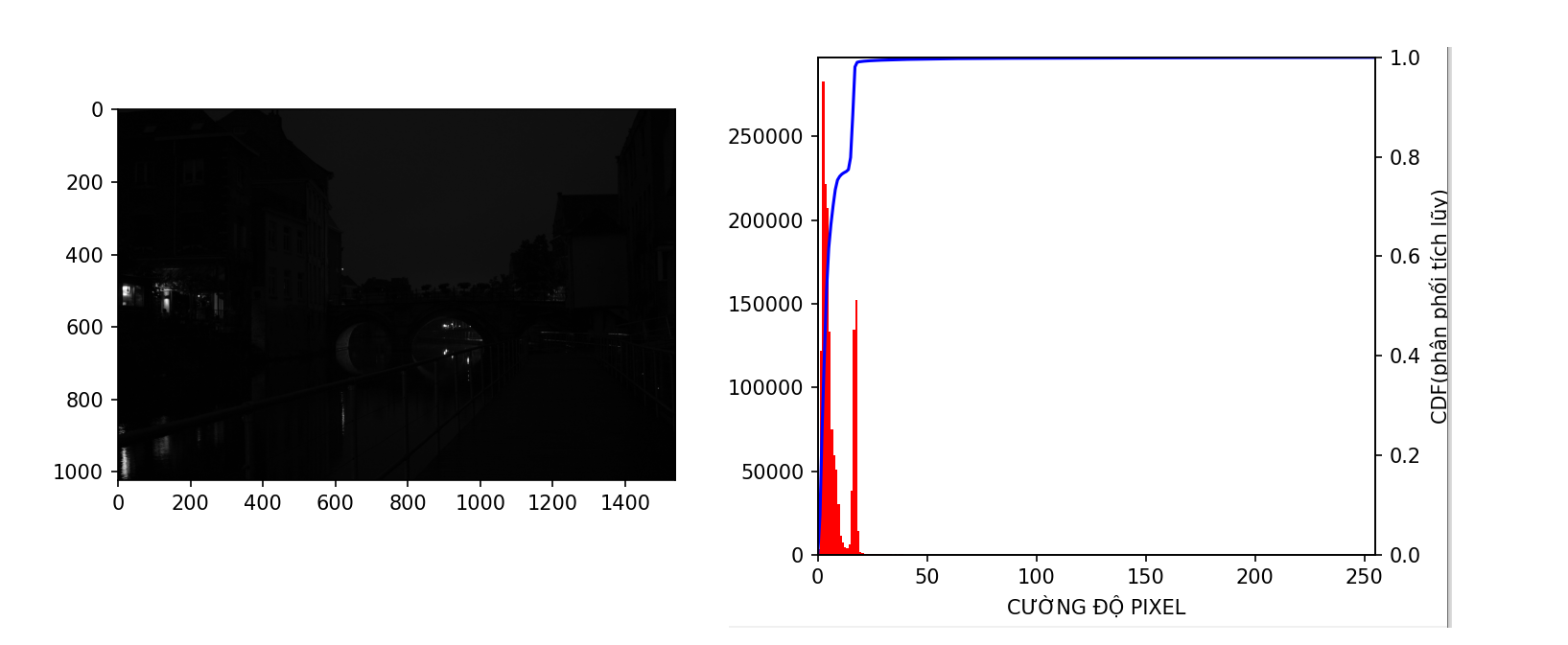
* 1. Equal histogram



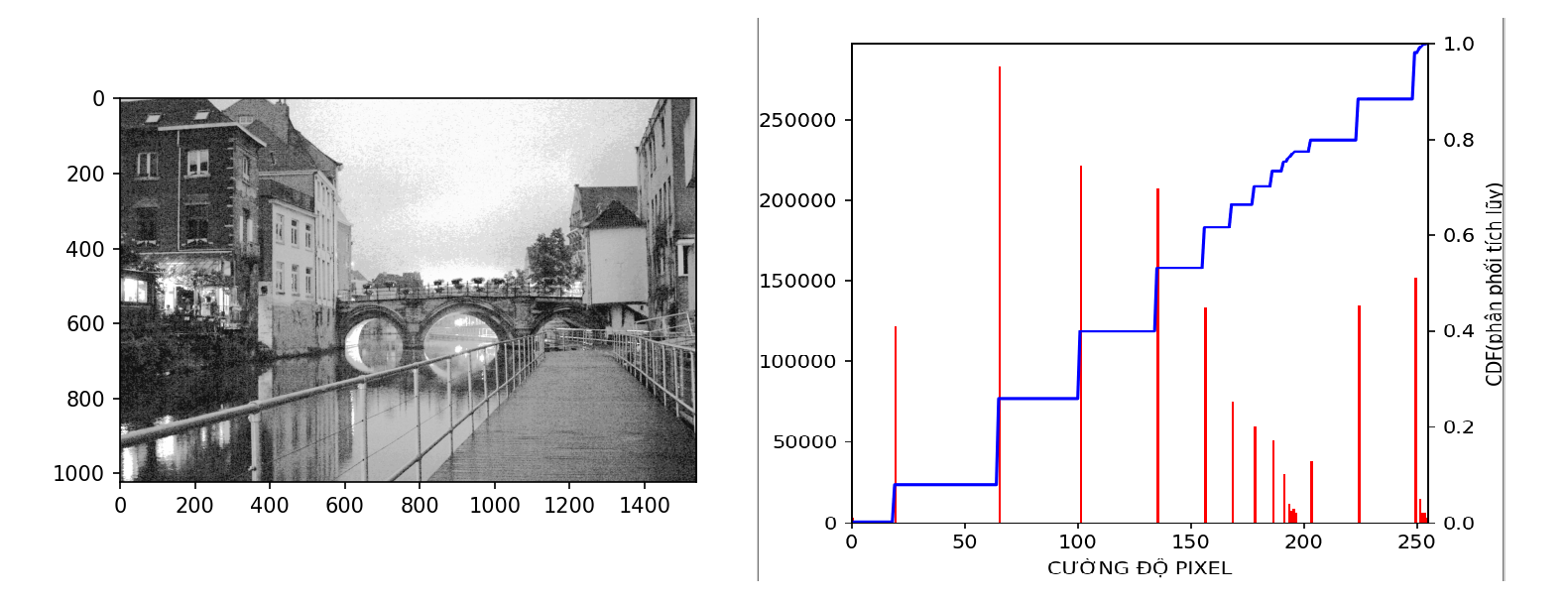
3.5 Experimental results



* Ảnh và đồ thị trước khi xử lý cân bằng



* Ảnh và đồ thị sau khi xử lý cân bằng



# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

1. Lương Mạnh Bá, Nguyễn Thanh Thủy (2008), “Nhập môn xử lý

**Tiếng Anh**

3. Rafael C. Gonzalez, Richard E. Woods (2002), “Digital Image Processing”, Prentice-Hall, New Jersey.

4. <https://docs.python.org/3/library/statistics.html>

5. <https://www.educative.io/answers/what-is-histogram-equalization-in-python>

6. <https://levelup.gitconnected.com/introduction-to-histogram-equalization-for-digital-image-enhancement-420696db9e43?gi=a84cc663c9da>