TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TIỂU LUẬN MÔN XÁC SUẤT VÀ THỐNG KÊ**

**ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

**MIDTERM ESSAY**

*Người hướng dẫn*:**TRẦN LƯƠNG QUỐC ĐẠI**

*Người thực hiện*:**ĐẶNG THÀNH NHÂN– 522H0006**

Lớp **:** **22H50201**

Khoá  **:** **26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TIỂU LUẬN MÔN XÁC SUẤT VÀ THỐNG KÊ**

**ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

**MIDTERM ESSAY**

Người hướng dẫn: **TRẦN LƯƠNG QUỐC ĐẠI**

Người thực hiện: **ĐẶNG THÀNH NHÂN**

Lớp **: 22H50201**

Khoá  **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

* Lời đầu tiên em xin cảm ơn sâu sắc đến thầy Trần Lương Quốc Đại đã dành thời gian để hỗ trợ và giúp đỡ em trong quá trình học tập và hoàn thành bài tiểu luận. Những lời khuyên và nhận xét đã giúp em hiểu rõ hơn và hoàn thành bài tiểu luận này.
* Xin cảm ơn khoa Công Nghệ Thông Tin đã giao cho em bài tiểu luận này. Em tin chắc rằng bài tiểu luận này sẽ giúp em rèn luyện, tiếp thu và phát triển những kĩ năng để phục vụ cho công việc tương lai của mình.
* Do chưa có nhiều kinh nghiệm làm tiểu luận cũng như những hạn chế về kiến thức nên trong bài tiểu luận chắc chắn sẽ có những sai sót. Rất mong nhận được những nhận xét và ý kiến đóng góp từ thầy/cô để em có thể hoàn thiện hơn về kĩ năng này.
* Lời cuối cùng em chân thành cảm ơn thầy/cô đã đọc và cho những đánh giá. Chúc thầy/cô thầy thật nhiều sức khỏe, thành công và hạnh phúc.

**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 Thầy Trần Lương Quốc Đại. 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.

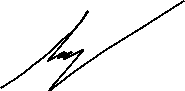
Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**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 tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*



*Đặng Thành Nhân*

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

Trình bày tóm tắt vấn đề nghiên cứu, các hướng tiếp cận, cách giải quyết vấn đề và một số kết quả đạt được, những phát hiện cơ bản trong vòng 1 -2 trang.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc149684586)

[CAM KẾT i](#_Toc149684586)i

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc149684587)

[TÓM TẮT iv](#_Toc149684588)

[MỤC LỤC 1](#_Toc149684589)

[CHƯƠNG 1 – THƯ VIỆN THỐNG KÊ TRONG PYTHON 2](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509191)

[1.1 Tổng quan về thư viện thống kê trong Python. 2](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509192)

[1.2 Giải thích ý nghĩa của từng hàm 2](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509193)

[CHƯƠNG 2 – GIẢI THUẬT CÂN BẰNG LƯỢC ĐỒ (HISTOGRAM EQUALIZATION) 3](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509195)

[2.1 Các vấn đề, ràng buộc/điều kiện, phương pháp/thuật toán. 3](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509196)

[2.2 Hình ảnh minh họa. 3](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509197)

[2.3 Nhận xét, phân tích, đánh giá. 3](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509198)

[CHƯƠNG 3 – GIẢI THUẬT CÂN BẰNG LƯỢC ĐỒ (HISTOGRAM MATCHING) 4](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509199)

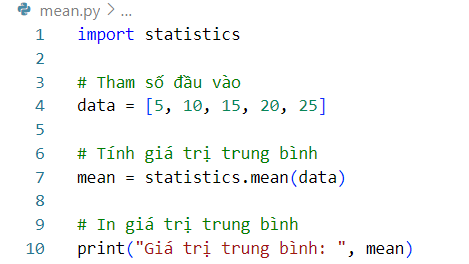
[3.1 Các vấn đề, ràng buộc/điều kiện, phương pháp/thuật toán. 4](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509200)

[3.2 Hình ảnh minh họa. 4](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509201)

[3.3 Nhận xét, phân tích, đánh giá. 4](file:///C:\Users\ASUS\Downloads\522H0090.docx#_Toc149509202)

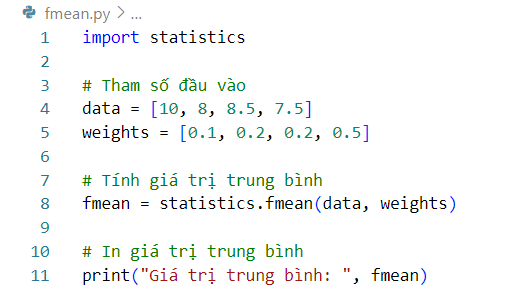
CHƯƠNG 1 – THƯ VIỆN THỐNG KÊ TRONG PYTHON

* 1. Tổng quan về thư viện thống kê trong Python:
     1. *Trung bình và phương pháp tìm vị trí trung tâm*
* **mean()**: Hàm tính trung bình cộng dữ liệu.
* **fmean()**: Hàm tính trung bình cộng dữ liệu với khả năng đặt trọng số tùy chọn.
* **geomectric\_mean()**: Hàm tính trung bình hình học của dữ liệu.
* **harmonic\_mean()**: Hàm tính trung bình điều hòa của dữ liệu.
* **median()**: Hàm tính trung vị (giá trị ở giữa) của dữ liệu.
* **median\_low()**: Hàm tính trung vị thấp của dữ liệu.
* **median\_high()**: Hàm tính trung vị cao của dữ liệu.
* **median\_grouped()**: Hàm tính trung vị, hoặc phân vị thứ 50, của dữ liệu đã được nhóm lại.
* **mode()**: Hàm trả về duy nhất một giá trị mode (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu danh nghĩa.
* **multimode()**: Hàm trả về một danh sách chứa giá trị mode (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu danh nghĩa.
* **quantiles()**: Hàm chia dữ liệu thành các khoảng có xác xuất bằng nhau.
  + 1. *Đo lường về sự phân tán*
* **pstdev()**: Hàm tính độ lệch chuẩn của quần thể dữ liệu (population).
* **pvariance()**: Hàm tính phương sai của quần thể dữ liệu (population).
* **stdev()**: Hàm tính độ lệch chuẩn của dữ liệu mẫu (sample).
* **variance()**: Đây là hàm tính phương sai của dữ liệu mẫu (sample).
  + 1. *Thống kê quan hệ cho hai dữ liệu đầu vào*
* **covariance():** Hiệp phương sai mẫu của hai biến.
* **correlation():** Hệ số tương quan Pearson và Spearman.
* **linear\_regression**(): Độ dốc và điểm cắt của phương trình hồi quy tuyến tính đơn giản.
  1. Giải thích ý nghĩa của từng hàm.
     1. *Trung bình và phương pháp tìm vị trí trung tâm*
* **Hàm mean():**
* **Công dụng:** Dùng để tính giá trị trung bình (arithmetic mean) từ dữ liệu đầu vào. Thường được dùng để xác định được xu hướng trung tâm của dữ liệu.
* **Cú pháp sử dụng hàm:** statistics.mean(data).
* **Tham số đầu vào:** Tham số đầu vào hàm mean() là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là giá trị trung bình của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ1: Hàm mean() nhận vào dữ liệu [5, 10, 15, 20, 25].
* Kết quả trả về sẽ là 15



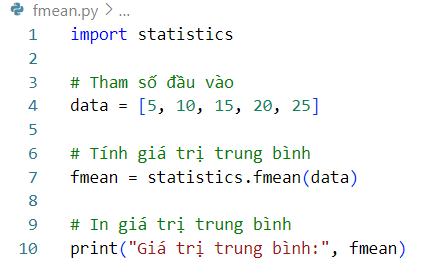


* **Hàm fmean():**
* **Công dụng**: Hàm fmean() có chức năng tương tự với hàm mean() và là một phiên bản nhanh hơn của hàm mean(). Hàm fmean() cho kết quả chính xác hơn hàm mean() với việc chuyển đổi dữ liệu sang kiểu dữ liệu thực (float) trước khi tính toán.
* **Cú pháp sử dụng hàm:** statistics.fmean(data) hoặc statistics.fmean(data, weights).
* **Tham số đầu vào:**
* **data**: Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **weights**: Là một tập dữ liệu số (numberic) tương tự như data, wights chứa các trọng số tương ứng với từng phần tử trong data. Tham số này là tùy chọn và mặc định là None.
* **Tham số đầu ra:** Là một giá trị số thực (float) chứa giá trị trung bình của giá trị đầu vào data theo tỉ lệ tương ướng weights.
* Ví dụ:
* Ví dụ 1: Tính điểm trung bình của một môn học bao gồm:
* Điểm quá trình 1 (10%): 10
* Điểm quá trình 2 (20%): 8
* Điểm quá giữa kì (20%): 8.5
* Điểm quá cuối kì (50%): 7.5
* Điểm trung bình môn học sẽ là: 8.05.



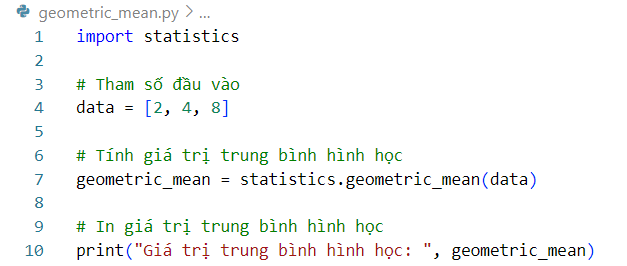


* Ví dụ 2: Hàm fmean() nhận vào dữ liệu [5, 10, 15, 20, 25]. Trong trường hợp này tham số weights được bỏ trống nên sẽ được mặc định là None**.**
* Kết quả trả về sẽ là 15.0



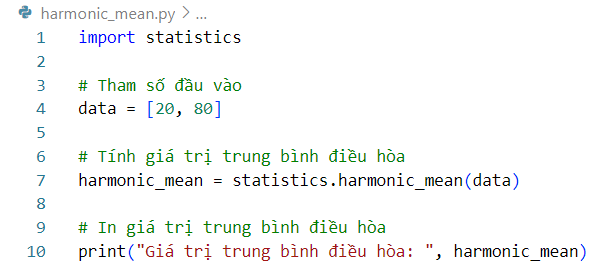


* **Hàm geomectric\_mean():**
* **Công dụng:** Hàm tính trung bình hình học chỉ ra giá trị trung tâm hoặc giá trị điển hình của dữ liệu bằng công thức căn bậc n của tích n phần tử trong tham số đầu vào(với n là tổng số phần tử có trong tham số đầu vào).
* **Cú pháp sử dụng hàm:** statistics.geometric\_mean(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Trong đó tất cả giá trị phải là số dương. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một giá trị rỗng (null), chứa một phần tử âm hoặc bằng 0 thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra**: Là một giá trị số thực (float) chứa giá trị trung bình hình học của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm **geometric mean()** nhận vào dữ liệu[2, 4, 8].
* 4 \* 8 = 64
* Căn bậc 3 của 64 = 4.
* Giá trị trung bình hình học là: 4.0



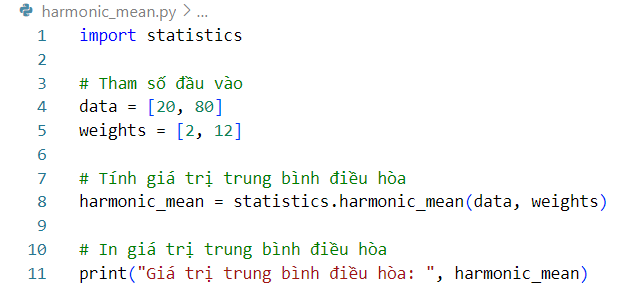


* **Hàm harmonic\_mean():**
* **Công dụng:** Hàm harmonic\_mean() là hàm tính giá trị trung bình điều hòa dữ liệu bằng công thức nghịch đảo trung bình cộng của nghịch đảo các số trong tham số đầu vào.
* **Cú pháp sử dụng hàm:** harmonic\_mean(data, weights).
* **Tham số đầu vào:**
* **data**: Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Trong đó tất cả giá trị phải là số dương. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data của hàm harmonic\_mean là một giá trị rỗng (null) hoặc chứa một phần tử là số âm thì hàm sẽ trả về lỗi StatisticsError. Nếu tham số đầu vào data chứa một phần tử là số 0 thì kết quả trả về sẽ là 0.
* **weights**: Có kiểu dữ liệu tương tự như data, weights chứa các trọng số tương ứng với từng phần tử trong data. Tham số này là tùy chọn và mặc định là None.
* **Tham số đầu ra:** Là một giá trị số thực (float) chứa giá trị trung bình điều hòa của giá trị đầu vào data theo tỉ lệ tương ứng weights.
* **Ví dụ:**
* Ví dụ 1: Giả sử một chiếc xe máy đang đi trong 2 đoạn với tốc độ lần lượt là 20, 80. Trong trường hợp này giá trị weightsđược bỏ trống nên sẽ được mặc định là None.
* Giá trị trung bình điều hòa tốc độ của xe máy trên 2 đoạn đường là nghịch đảo của 1/32: 32.0



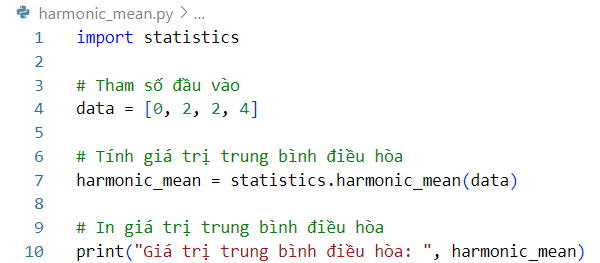


* Ví dụ 2: Giả sử một chiếc xe máy đang đi trong đường hẹp nên duy trì tốc độ 20 km/h trong 2km. Sau khi đi ra khỏi đường hẹp nên xe tăng tốc độ lên 80km/h trong 12km tiếp theo.
* Giá trị trung bình điều hòa tốc độ xe máy trên 2 quãng đường là: 56.0



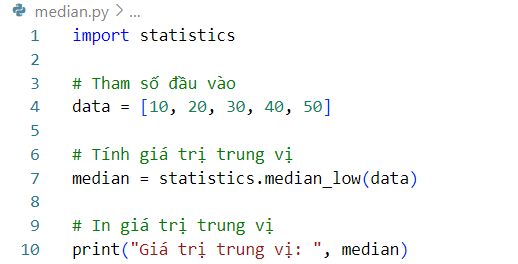


* Ví dụ 3: Hàm harmonic\_mean() nhận vào dữ liệu [0, 2, 4, 4]. Trong trường hợp này có một phần tử bằng 0 trong tham số đầu vào data.
* Kết quả trả về sẽ là: 0



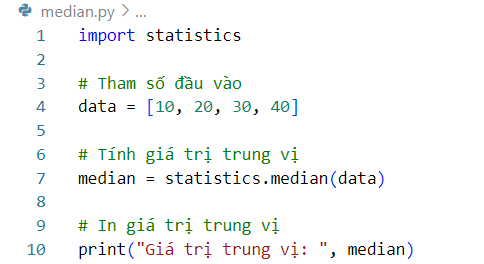


* **Hàm median():**
* **Công dụng:** Hàm median() cho biết giá trị ỡ giữa sau sắp xếp của dữ liệu đầu vào của n phần trong tham số đầu vào. Với n (là độ dài của tham số đầu vào):
* Nếu n là một giá trị lẻ thì giá trị trung vị nằm ở ví giữa của tập tham số đầu vào.
* Nếu n là một giá trị chẵn thì giá trị trung vị sẽ bằng với trung bình cộng của 2 phần tử ở giữa.
* **Cú pháp sử dụng hàm:** statistics.median(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vàodata là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) chứa giá trị trung vị của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm median() nhận tham số đầu vào data = [10, 20, 30, 40, 50]. Trong trường hợp này dữ liệu đầu vào chứa 5 tham số vậy n là số lẻ.
* Giá trị trung vị sẽ là phần tử ở giữa mảng truyền vào: 30



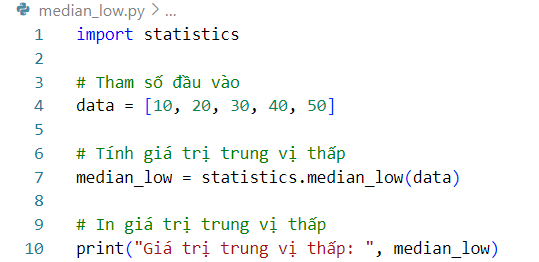


* Ví dụ 2: Hàm median() nhận tham số đầu vào data = [10, 20, 30, 40]. Trong trường hợp này dữ liệu đầu vào chứa 4 tham số vậy n là số chẵn.
* Giá trị trung vị bằng trung bình cộng của 2 phần tử ở giữa: (20+30) / 2 = 25.0



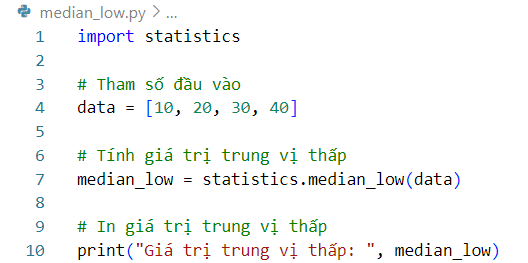


* **Hàm median\_low():**
* **Công dụng:** Hàm median\_low() dùng để tính trung vị thấp của n phần tử có trong dữ liệu truyền vào. Với n (là độ dài của tham số đầu vào):
* Nếu n là một giá trị lẻ thì giá trị trung vị thấp nằm ở ví giữa của tập tham số đầu vào.
* Nếu n là một giá trị chẵn thì giá trị trung vị thấp sẽ là giá trị nhỏ hơn giữa 2 phần tử nằm giữa mảng.
* **Cú pháp sử dụng hàm:** statistics.median\_low(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vàodata là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) chứa giá trị trung vị thấp của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm median\_low() nhận tham số đầu vào data = [10, 20, 30, 40, 50]. Trong trường hợp này dữ liệu đầu vào chứa 5 tham số vậy n là số lẻ.
* Giá trị trung vị thấp là phần tử ở giữa mảng truyền vào: 30



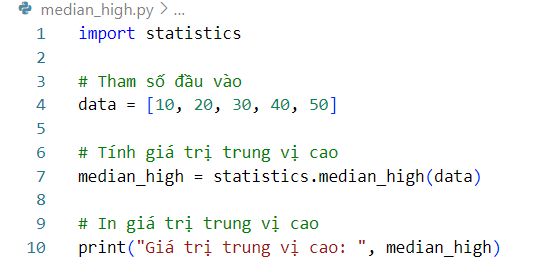


* Ví dụ 2: Hàm median\_low() nhận tham số đầu vào data = [10, 20, 30, 40]. Trong trường hợp này dữ liệu đầu vào chứa 4 tham số vậy n là số chẵn.
* Giá trị trung vị thấp là giá trị nhỏ hơn giữa 2 phần tử ở giữa mảng: 20 (Vì 20 < 30)



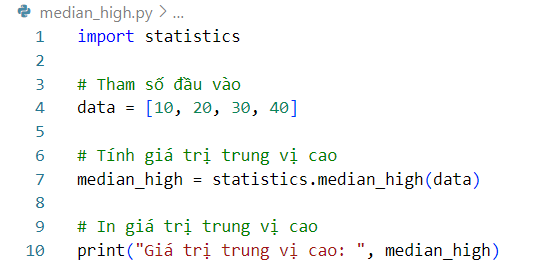


* **Hàm median\_high():**
* **Công dụng:** Dùng để tính trung vị thấp của n phần tử có trong dữ liệu truyền vào. Với n (là độ dài của tham số đầu vào):
* Nếu n là một giá trị lẻ thì giá trị trung vị cao nằm ở ví giữa của tập tham số đầu vào.
* Nếu n là một giá trị chẵn thì giá trị trung vị cao sẽ là giá trị lớn hơn giữa 2 phần tử nằm giữa mảng.
* **Cú pháp sử dụng hàm:** statistics.median\_high(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vàodata là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) chứa giá trị trung vị cao của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm median\_high() nhận tham số đầu vào data = [10, 20, 30, 40, 50]. Trong trường hợp này dữ liệu đầu vào chứa 5 tham số vậy n là số lẻ.
* Giá trị trung vị cao là phần tử ở giữa mảng truyền vào: 30



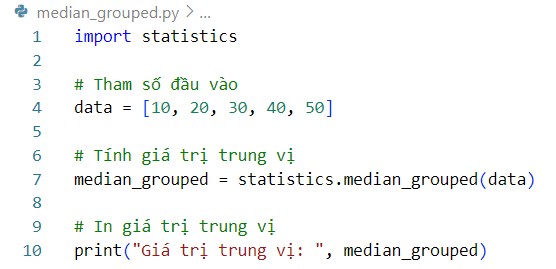


* Ví dụ 2: Hàm median\_low() nhận tham số đầu vào data = [10, 20, 30, 40]. Trong trường hợp này dữ liệu đầu vào chứa 4 tham số vậy n là số chẵn.
* Giá trị trung vị cao là giá trị lớn hơn giữa 2 phần tử ở giữa mảng: 30 (Vì 30 > 20)



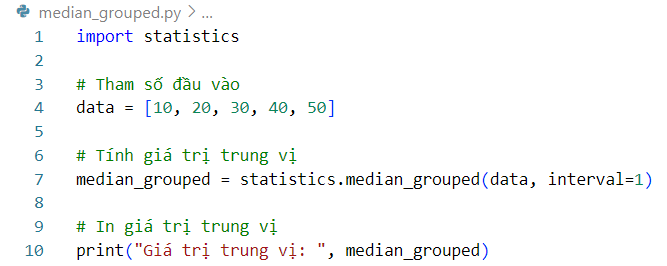


* **Hàm median\_grouped():**
* **Công dụng:** Hàm median\_grouped() dùng để tính trung vị(median) của tham số đầu vào khi chúng đã được nhóm thành các khoảng (interval). Phương pháp này thường được sử dụng khi dữ liệu liên tục và không đơn **lẻ.**
* **Cú pháp sử dụng hàm:** statistics.median\_grouped(data, interval).
* **Tham số đầu vào:**
* **data:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Trong đó tất cả giá trị phải là số dương. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một giá trị rỗng thì hàm sẽ trả về lỗi StatisticsError.
* **interval:** là một số nguyên dương. Tham số này là tùy chọn và mặc định là 1.
* **Tham số đầu ra**: Là một giá trị số thực (float) chứa giá trị trung vị của tham số đầu vào datađã được nhóm lại hoặc tính ở phận vị thứ 50.
* **Ví dụ:**
* Ví dụ 1: Hàm median\_grouped() nhận dữ liệu đầu vào data = [10, 20, 30, 40, 50]. Trong trường hợp này giá trị interval được bỏ trống sẽ được mặc định là 1.
* Giá trị trung vị sẽ là: 30.0



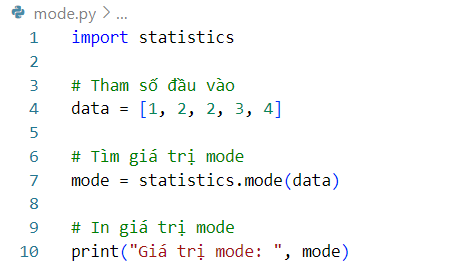


* Ví dụ 2: Hàm **median\_grouped()** nhận dữ liệu đầu vào **data** = [10, 20, 30, 40, 50] và **interval = 1.**
* Giá trị trung vị sẽ là: 30.0



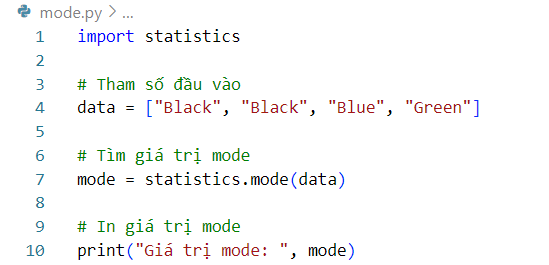


* **Hàm mode():**
* **Công dụng:** Hàm mode() dùng để tính giá trị mode của một tập tham số đầu vào. Mode là giá trị xuất hiện với tần suất nhiều nhất trong tập tham số đầu vào. Thường được sử dụng để xác định giá trị phổ biến nhất trong tập dữ liệu.
* **Cú pháp sử dụng hàm:** statistics.mode(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vàodata là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) hoặc chuỗi(string) chứa giá trị xuất hiện với tần suất nhiều nhất trong tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm mode() nhận tham số đầu vào data = [1, 2, 2, 3, 4]
* Giá trị mode trả về sẽ là: 2 (với 2 lần xuất hiện).



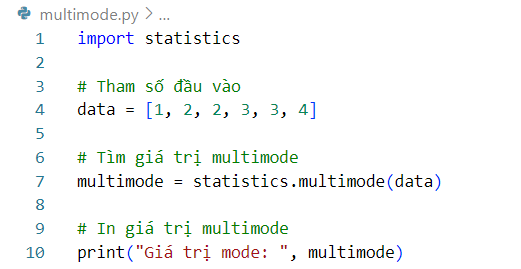


* Ví dụ 2: Hàm mode() nhận tham số đầu vào data = [“Black”, “Black”, “Blue”, “Green”]
* Giá trị mode trả về sẽ là: Black (với 2 lần xuất hiện).



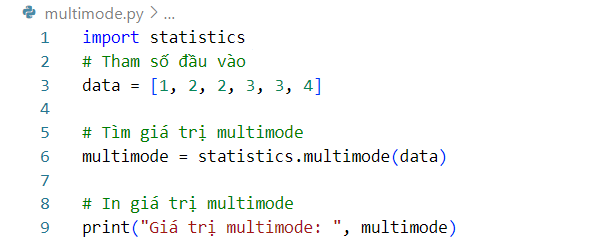


* **Hàm multimode():**
* **Công dụng:** Khác với hàm mode() chỉ trả về một giá trị với tần suất xuất hiện nhiều nhất trong tham số đầu vào hàm multimode() trả về một tập các các giá trị có tần suất xuất hiện nhiều nhất trong tập tham số đầu vào.
* **Cú pháp sử dụng hàm:** statistics.multimode(data).
* **Tham số đầu vào:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vàodata là một giá trị rỗng (null) thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một tập các giá trị số thực (float) hoặc chuỗi(string) xuất hiện với tần suất nhiều nhất trong tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm multimode() nhận tham số đầu vào data = [1, 2, 2, 3, 3, 4]
* Giá trị multimode trả về sẽ là: 2 và 3 (với 2 lần xuất hiện).



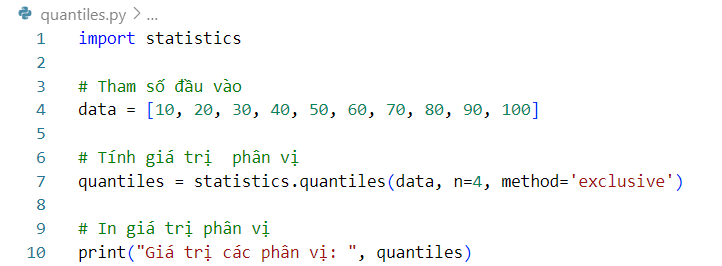


* Ví dụ 2: Hàm multimode() nhận tham số đầu vào data = [“Black”, “Black”, “Blue”, “Blue”, “Green”]
* Giá trị multimode trả về sẽ là: Black và Blue (với 2 lần xuất hiện).



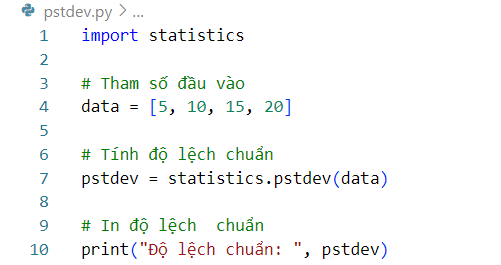


* **Hàm quantiles():**
* **Công dụng:** Hàm quantiles() dùng để tính các giá trị phân vị (quantiles) của tham số đầu vào.
* **Cú pháp sử dụng hàm:** statistics.quantiles(data, n, method).
* **Tham số đầu vào:**
* **data** : Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống.
* **n**: Số lượng khoảng quantiles muốn chia dữ liệu thành. Giá trị mặc định là 4, tương đương với việc chia dữ liệu thành 4 khoảng tương đương với việc tính các phân vị 25%, 50%, 75%.
* **method**: Phương pháp sử dụng để tính quantiles. Có hai phương pháp được hỗ trợ: ‘inclusive’ và ‘exclusive’. Phương pháp inclusive tính quantiles bằng cách lấy giá trị trung bình của hai giá trị gần nhất khi chia dữ liệu thành các khoảng. Phương pháp ‘exclusive’ tính quantiles bằng cách lấy giá trị gần nhất khi chia dữ liệu thành các khoảng. Giá trị mặc định là ‘exclusive’.
* Nếu tham số đầu vàon của hàm quantiles() lớn hơn 1 hoặc data chỉ có 1 phần tử hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra**: Là một tập các giá trị số thực (float) chứa giá trị phân vị tính được từ tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm quantiles() nhận tham số đầu vào data = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100], n = 4, method = exclusive. Hàm sẽ chia dữ liệu đầu vào data thành 4 khoảng và tính giá trị phân vị cho các khoảng đó.
* Kết quả là: [27.5, 55.0, 82.5] tương ứng với phân vị 25%, 50% và 75%



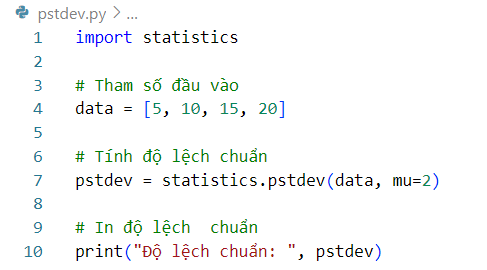


* + 1. ***Đo* lường về sự phân tán**
* **Hàm pstdev():**
* **Công dụng:** Hàm pstdev() dùng để tính độ lệch chuẩn mẫu của tham số đầu vào. Thường được dùng để đo mức độ phân tán và biến thiên của dữ liệu xung quanh giá trị trung bình. Ta có thể tính độ lệch chuẩn bằng cách căn bậc 2 của giá trị phương sai.
* **Cú pháp sử dụng hàm:** statistics.pstdev(data, mu).
* **Tham số đầu vào:**
* **data:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một tập rỗng thì hàm sẽ trả về lỗi StatisticsError.
* **mu**: Là giá trị trung bình của tổng quần thể. Giá trị mặc định là None. Nếu giá trị mu là None thì mu sẽ được gán bằng giá trị trung bình (mean) của tham số đầu vào.
* **Tham số đầu ra:** Là một giá trị số thực (float) biểu diễn độ lệch chuẩn của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm pstdev() nhậm tham số đầu vào data = [5, 10, 15, 20, 25]. Trong trường hợp này tham số mu được bỏ trống sẽ được mặc định là None.
* Kết quả của giá trị độ lệch chuẩn sẽ là: 5.5901699437494745



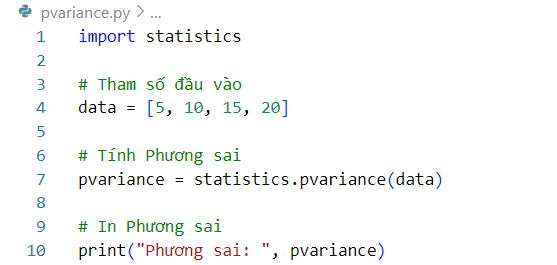


* Ví dụ 2: Hàm pstdev() nhậm tham số đầu vào data = [5, 10, 15, 20, 25], mu = 2.
* Kết quả của giá độ lệch sẽ là: 11.895377253370318



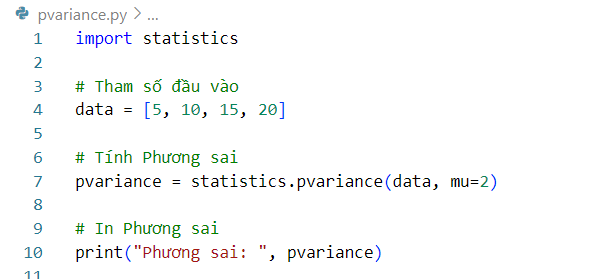


* **Hàm pvariance():**
* **Công dụng:** Hàm pvariance() dùng để tính phương sai mẫu của tham số đầu vào. Thường được dùng để đo mức độ phân tán và biến thiên của dữ liệu xung quanh giá trị trung bình. Ta có thể tính phương sai bằng cách bình phương giá trị độ lệch chuẩn.
* **Cú pháp sử dụng hàm:** statistics.pvariance(data, mu).
* **Tham số đầu vào:**
* **data** : Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một tập rỗng thì hàm sẽ trả về lỗi StatisticsError.
* **mu**: Thường là giá trị trung bình (mean) của tổng quần thể. Giá trị mặc định là None. Nếu giá trị mu là None thì mu sẽ được gán bằng giá trị trung bình (mean) của tham số đầu vào.
* **Tham số đầu ra:** Là một giá trị số thực (float) biểu diễn độ lệch chuẩn của tham số đầu vào data.
* **Ví dụ:**
* Ví dụ 1: Hàm pvariance() nhậm tham số đầu vào data = [5, 10, 15, 20, 25]. Trong trường hợp này tham số mu được bỏ trống sẽ được mặc định là None.
* Kết quả của giá trị phương sai sẽ là: 31.25





* Ví dụ 2: Hàm pvariance() nhậm tham số đầu vào data = [5, 10, 15, 20, 25], mu = 2.
* Kết quả của giá trị phương sai sẽ là: 141.5



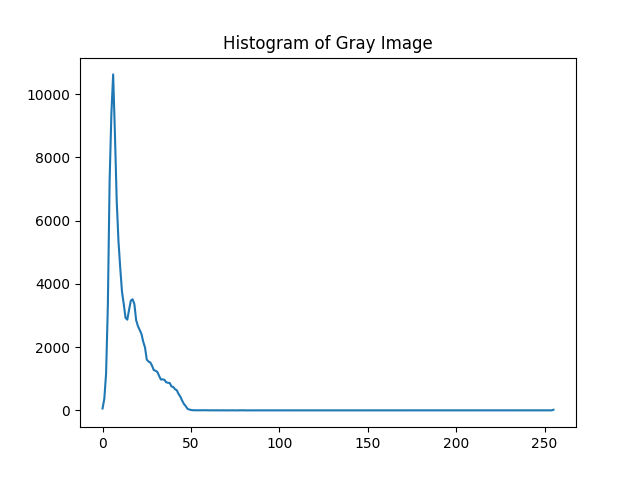


* **Hàm stdev():**
* **Công dụng:** Hàm stdev() có chức năng như hàm pstdev(). Hàm stdev tính độ lệch chuẩn dựa trên tổng thể các giá trị thay vì dựa trên một mẫu như hàm pstdev().Ta có thể tính độ lệch chuẩn bằng cách căn bậc 2 của giá trị phương sai.
* **Cú pháp sử dụng hàm:** statistics.stdev(data, xbar)
* **Tham số đầu vào:**
* **data:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data là một tập rỗng thì hàm sẽ trả về lỗi StatisticsError.
* **xbar:** Thường là giá trị trung bình của tổng quần thể. Giá trị mặc định là None.
* **Ví dụ:**
* **Hàm variance():**
* **Công dụng:** Hàm variance() có chức năng như hàm pvariance(). Hàm variance tính độ lệch chuẩn dựa trên tổng thể các giá trị thay vì dựa trên một mẫu như hàm pvariance().Ta có thể tính phương sai bằng cách bình phương giá trị độ lệch chuẩn.
* **Cú pháp sử dụng hàm:** statistics.variance(data, xbar).
* **Tham số đầu vào:**
* **data:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Tham số này là bắt buộc và không thể bỏ trống. Nếu tham số đầu vào data à một tập rỗng thì hàm sẽ trả về lỗi StatisticsError.
* **xbar:** Thường là giá trị trung bình (mean) của tổng quần thể. Giá trị mặc định là None. Nếu giá trị mu là None thì mu sẽ được gán bằng giá trị trung bình (mean) của tham số đầu vào.
* **Ví dụ:**
  + 1. *Thống kê quan hệ cho hai dữ liệu đầu vào*
* **Hàm covariance():**
* **Công dụng:** Dùng để tính hiệp phương sai (covariance) của 2 tham số đầu vào. Thường dùng để đo độ biến đổi đồng thời của 2 tham số đầu vào.
* **Cú pháp sử dụng hàm:** statistics.stdev(x, y)
* **Tham số đầu vào: x và y** là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Nếu tham 2 tham số đầu vào không cùng độ dài thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) thể hiện giá trị hiệp phương sai của 2 tham số đầu vào.
* **Ví dụ:**
* **Hàm correlation():**
* **Công dụng:** Dùng để tính hệ số tương quan (correlation coefficient) giữa 2 tham số đầu vào.
* **Cú pháp sử dụng hàm:** statistics.scorrelation(x, y)
* **Tham số đầu vào: x và y**à một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Nếu tham 2 tham số đầu vào không cùng độ dài thì hàm sẽ trả về lỗi StatisticsError.
* **Tham số đầu ra:** Là một giá trị số thực (float) thể hiện giá trị tương quan giữa 2 tham số đầu vào x và y chạy trong khoảng từ -1 đến 1:
* Nếu gần bằng 1 thì thể hiện chúng có mối quan hệ tuyến tính mạnh và biến đổi cùng chiều.
* Nếu gần bằng -1 thì thể hiện chúng có mối quan hệ tuyến tính yếu và biến đổi ngược chiều.
* Nếu gần bằng 0 thì không có mối quan hệ tuyến tính giữa 2 tham số đầu vào.
* **Ví dụ:**
* **Hàm linear\_regression**():
* **Công dụng:** Dùng để phân tích hồi quy tuyến tính của 2 tham số đầu vào.
* **Cú pháp sử dụng hàm:** statistics.linear\_regression(x, y, proportional).
* **Tham số đầu vào:**
* **x và y:** Là một tập dữ liệu số (numberic) như: mảng (array), danh sách (list), tuple. Nếu tham 2 tham số đầu vào không cùng độ dài thì hàm sẽ trả về lỗi StatisticsError.
* **proportional:** Là một tham số tùy chọn có kiểu dữ liệu boolen. Nếu proportional = True hàm sẽ thực hiện hồi quy tuyến tính theo tỷ lệ bằng cách tìm đường thẳng đi qua góc tọa độ (0, 0). Nếu proportional = False hàm sẽ tìm đường thẳng tối ưu nhất đi qua toàn bộ tham số đầu vào.
* **Tham số đầu ra:** Hàm trả về một tuple trong đó chứa 2 tham số hệ số góc (slope) và điểm cắt trục y của đường hồi quy (intercept).
* **Ví dụ:**

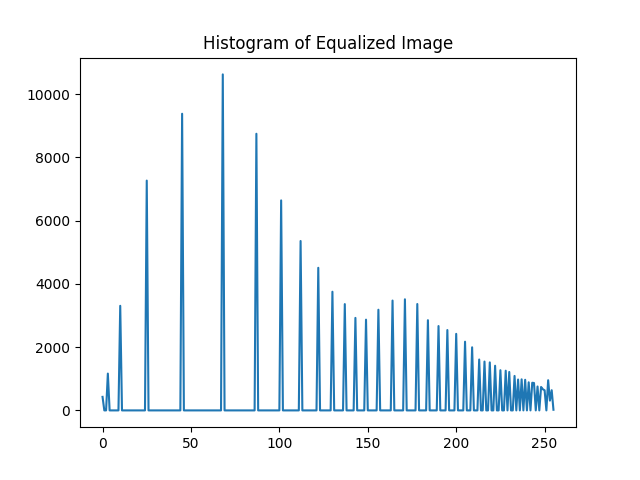
CHƯƠNG 2 – GIẢI THUẬT CÂN BẰNG LƯỢC ĐỒ (HISTOGRAM EQUALIZATION)

* 1. Các vấn đề, ràng buộc/điều kiện, phương pháp/thuật toán.
     1. *Các vấn đề*
* Bài toán: Histogram Equalization là một kỹ thuật xử lí hình ảnh bằng cách cân bằng lại độ tương phản và phân bố cường độ sáng của ảnh.
* Mục tiêu: Làm cân đối độ sáng trên toàn ảnh, giúp cải thiện chất lượng ảnh và làm rõ các chi tiết trên ảnh.
  + 1. *Ràng buộc/ điều kiện*
* Ràng buộc: Phương pháp Historam chỉ nhận ảnh đầu vào là ảnh xám (grayscale image), không tiếp nhận ảnh màu.
* Điều kiện: Ảnh cần có đủ số lượng pixel và mức độ khác nhau để cân bằng Histogram. Ảnh quá nhỏ sẽ không đủ dữ liệu.
  + 1. *Phương pháp/ thuật toán*
* Bước 1: Tính histogram của ảnh gốc bằng cách đếm số pixel tương ứng với mỗi mức xám.
* Bước 2: Tính hàm phân phối tích lũy (CDF) từ histogram bằng công thức CDF = tổng số các pixel ở từng mức xàm từ (0-255).
* Bước 3: Chuẩn hóa CDF để đảm bảo rằng giá trị lớn nhất trong CDF nằm trong khoảng từ (0-255).
* Bước 4: Sử dụng CDF đã chuẩn hóa để ánh xạ lại mức sáng của từng pixel trong ảnh đầu vào để tạo ra ảnh cân bằng histogram.
  + 1. *Nhận xét, phân tích, đánh giá.*
* Phương pháp Histogram Equalization là một phương pháp đơn giản, dễ thực hiện và hiệu quả trong việc cải thiện chất lượng hình ảnh.
* Tuy nhiên trong một số trường hợp phương pháp này có thể gây mất thông tin về độ sáng tổng thể của ảnh, tăng nhiều và làm mờ một số chi tiết của ảnh.
* Cần xem xét kĩ lưỡng và chọn phương pháp phù hợp cho từng loại ảnh khác nhau.
  1. Hình ảnh minh họa.

Ảnh và lược đồ Histogram của ảnh trước khi cân bằng



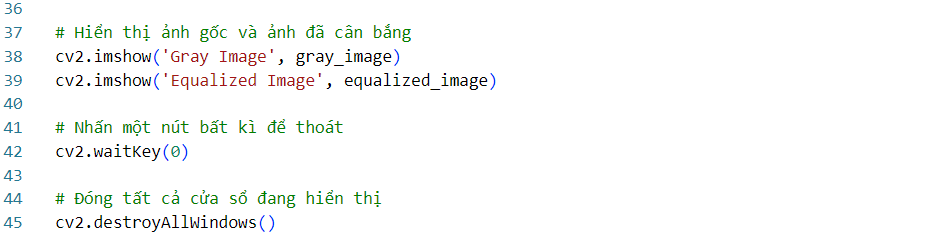
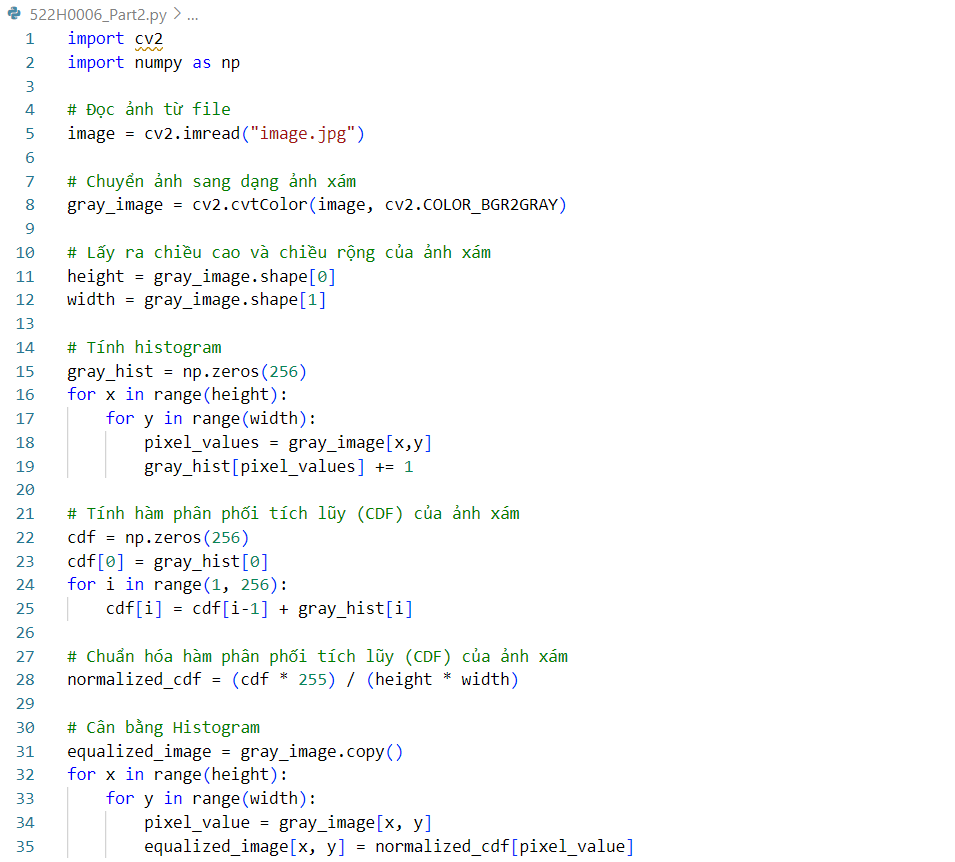
Ảnh và lược đồ Histogram của ảnh sau khi cân bằng



CHƯƠNG 3 – GIẢI THUẬT CÂN BẰNG LƯỢC ĐỒ (HISTOGRAM MATCHING)

* 1. Các vấn đề ràng buộc/điều kiện, phương pháp/thuật toán.
     1. *Các vấn đề*
* Bài toán: Histogram Matching là một kỹ thuật xử lí hình ảnh bằng cách cập nhật độ tương phản và phân bố cường độ sáng của ảnh theo một ảnh tham chiếu.
* Mục tiêu: Cân bằng chất lượng hình ảnh của một ảnh theo ảnh tham chiếu.
  + 1. *Ràng buộc/điều kiện*
* Chỉ áp dụng cho ảnh có cùng kích thước và mức xám
* Cần xác định ngưỡng cho sai khác giữa 2 histo
  + 1. *Phương pháp/thuật toán*
    2. *Nhận xét, phân tích, đánh giá*

CHƯƠNG 4 – THỰC HIỆN CHƯƠNG TRÌNH

* 1. Thuật toán cân bằng lược đồ (Histogram Equalization)
     1. ***Code*
     2. *Giải thích code*
* Dòng 1-2: Import thư viện OpenCV và thư viện Numpy(np).
* Dòng 5: Đọc ảnh từ file “image.jpg” và lưu vào biến image.
* Dòng 8: Chuyển đổi ảnh từ không gian màu BGR sang ảnh xám (grayscale).
* Dòng 10-17: Định nghĩa hàm compute\_histogram() để tính histogram của một ảnh bằng phương pháp đếm số pixel tương ứng với mỗi mức xám:
* Dòng 20-25: Định nghĩa hàm compute\_CDF() để tính hàm phân phối tích lũy (CDF) từ một mảng histogram đầu vào bằng cách cộng tổng các pixel ở từng mức xám tương ứng từ (0-255).
* Dòng 28-30: Định nghĩa hàm normalize\_CDF() chuẩn hóa một sơ đồ phân phối tích lũy (CDF) đầu vào bằng công thức (CDF \* 255) / (H\*W). Với H là chiều cao của ảnh và W là chiều rộng của ảnh.
* Dòng 33-43: Định nghĩa hàm equalize\_image() để cân bằng một ảnh đầu vào bằng cách lần lượt gọi các hàm compute\_histogram() để tính histogram của ảnh đầu vào. Sau đó gọi hàm compute\_CDF() để tính hàm phân phối tích lũy của histogram vừa tính được. Kế tiếp gọi hàm normalize\_CDF() để chuẩn hóa CDF vừa tính được. Cuối cùng ta duyệt qua từng pixel của ảnh và ánh xạ dựa trên CDF đã chuẩn hóa để có được ảnh cân bằng.
* Dòng 45: Gọi hàm equalize\_image() để cân bằng cách gray\_image sau đó lưu vào biến equalized\_image.
* Dòng 48-49: Hiển thị ảnh gốc và ảnh đã cân bằng bằng cửa sổ hiển thị của thư viện OpenCV. Dòng lệnh sẽ hiện 2 ảnh song song.
* Dòng 52: Đợi nhấn một nút bất kì để thoát khỏi chương trình.
* Dòng 55: Đóng tất cả cửa sổ đang mở.
  + 1. *Hướng dẫn xây dựng và chạy chương trình*
* Bước 1: Đảm bảo đã cài đặt thư viện OpenCV và NumPy trên máy tính.
* Bước 2: Tạo một file python mới để lưu đoạn code mẫu trên vào.
* Bước 3: Chuyển file ảnh bạn muốn cân bằng vào cùng thư mục với mã code python và đặt tên file ảnh là “image.jpg”
* Bước 4: Lưu chạy mã code python.
* Bước 5: Nhấn một phím bất kì để đóng các cửa sổ sau khi xem kết quả thực thi.
  + 1. *Thực nghiệm và kết luận*
* Qua thực nghiệm, có thể dễ dàng nhận thấy rằng ảnh đã cân bằng lược đồ có độ tương phản cao hơn và các chi tiết nổi bật hơn ảnh gốc. Phương pháp cân bằng lược đồ là một phương pháp hiệu quả trong việc cải thiện chất lượng ảnh và làm nổi bật các đặc trưng của ảnh.
  1. Thuật toán cân bằng lược đồ theo một mức xám định sẵn (Histogram Matching)
     1. *Code*
     2. *Giải thích code*
     3. *Hướng dẫn xây dựng và chạy chương trình*
     4. *Thực nghiệm và kết luận*
     5. *Kết quả*

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

1. Quách Ngọc Ân (1992), “Nhìn lại hai năm phát triển lúa lai”, *Di tuyền học ứng dụng*, 98(1), tr. 10-16.
2. Bộ nông nghiệp & PTNT (1996), *Báo cáo tổng kết 5 năm (1992-1996) phát triển lúa lai,* Hà Nội.
3. Nguyễn Hữu Đống, Đào Thanh Bằng, Lâm Quang Dụ, Phan Đức Trực (1997), *Đột biến –* *Cơ sở lý luận và ứng dụng,* Nhà xuất bản nông nghiệp, Viện khoa học kỹ thuật nông nghiệp Việt Nam, Hà Nội.
4. Nguyễn Thị Gấm (1996), *Phát hiện và đánh giá một số dòng bất dục đực cảm ứng nhiệt* *độ,* Luận văn thạc sĩ khoa học nông nghiệp, Viện khoa học kỹ thuật nông nghiệp Việt Nam, Hà Nội.

……….

1. Võ Thị Kim Huệ (2000), *Nghiên cứu chẩn đoán và điều trị bệnh…,* Luận án Tiến sĩ y khoa, Trường đại học y Hà Nội, Hà Nội.

**Tiếng Anh**

1. Anderson J.E. (1985), The Relative Inefficiency of Quota, The Cheese Case, *American* *Economic Review*, 75(1), pp. 178-90.
2. Borkakati R. P.,Virmani S. S. (1997), Genetics of thermosensitive genic male sterility in Rice, *Euphytica* 88, pp. 1-7.
3. Boulding K.E. (1955), *Economics Analysis*, Hamish Hamilton, London.
4. Burton G. W. (1988), “Cytoplasmic male-sterility in pearl millet (penni-setum glaucum L.)”, *Agronomic Journal* 50, pp. 230-231.
5. Central Statistical Oraganisation (1995), *Statistical Year Book*, Beijing.
6. FAO (1971), *Agricultural Commodity Projections (1970-1980)*, Vol. II. Rome.

**PHỤ LỤC**

Phần này bao gồm những nội dung cần thiết nhằm minh họa hoặc hỗ trợ cho nội dung luận văn như số liệu, biểu mẫu, tranh ảnh. . . . nếu sử dụng những câu trả lời cho một *bảng câu hỏi thì bảng câu hỏi mẫu này phải được đưa vào phần Phụ lục ở dạng nguyên bản* đã dùng để điều tra, thăm dò ý kiến; **không được tóm tắt hoặc sửa đổi**. Các tính toán mẫu trình bày tóm tắt trong các biểu mẫu cũng cần nêu trong Phụ lục của luận văn. Phụ lục không được dày hơn phần chính của luận văn