TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**Tiểu luận Quá trình 2 môn Xác suất thống kê**

*Người hướng dẫn*: **NGUYỄN LÂM**

*Người thực hiện*: **TRẦN HỮU ĐẠT – 522H0081**

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**



**Tiểu luận Quá trình 2 môn Xác suất thống kê**

*Người hướng dẫn:* **NGUYỄN LÂM**

*Người thực hiện:* **TRẦN HỮU ĐẠT**

Lớp **: 22H50201**

Khoá  **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

# LỜI CẢM ƠN

Tôi xin gửi lời cảm ơn sâu sắc đến Thầy Nguyễn Lâm đã hỗ trợ và giúp đỡ tôi trong suốt quá trình nghiên cứu và hoàn thành bài tiểu luận môn Xác suất thống kê này. Thầy đã tận tình hướng dẫn và giúp đỡ tôi trong quá trình thực hiện đề tài trên. Những lời khuyên và nhận xét của thầy đã giúp tôi hiểu rõ hơn về những khó khăn hạn chế của bài tiểu luận này. Cuối cùng, Tôi cũng xin cảm ơn tất cả các thầy cô trong khoa Công Nghệ Thông Tin của Trường Đại học Tôn Đức Thắng, đã cung cấp cho tôi những điều kiện tốt nhất để thực hiện bài tiểu luận trên.

# 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)

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc141789419)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN ii](#_Toc141789420)

[MỤC LỤC 1](#_Toc141789421)

[Chương 1 : Trung bình và các đo lường trung tâm 2](#_Toc141789422)

[I. Tổng quan về các hàm: 2](#_Toc141789423)

[II. Giải thích ý nghĩa của từng hàm 2](#_Toc141789424)

[Chương 2 : Đo lường về sự phân tán của dữ liệu 10](#_Toc141789425)

[I. Tổng quan về các hàm: 10](#_Toc141789426)

[II. Giải thích ý nghĩa của từng hàm 11](#_Toc141789427)

[Chương 3 : Thống kê cho mối quan hệ giữa hai đầu vào 14](#_Toc141789428)

[I. Tổng quan về các hàm: 14](#_Toc141789429)

[II. Giải thích ý nghĩa của từng hàm 14](#_Toc141789430)

[Chương 4: Lớp statistics.NormalDist và các phép tính liên quan 17](#_Toc141789431)

[I. Tổng quát: 17](#_Toc141789432)

[II. Ví dụ code minh họa của lớp statistics.NormalDist : 18](#_Toc141789433)

# 

# Chương 1 : Trung bình và các đo lường trung tâm

## I. Tổng quan về các hàm:

1. statistics.mean(): Trung bình cộng của dữ liệu.

2. statistics.fmean(): Trung bình cộng với số thập phân, tính nhanh hơn, có thể trọng số tùy chọn

3. statistics.geometric\_mean(): Trung bình hình học của dữ liệu.

4. statistics.harmonic\_mean(): Trung bình điều hoà của dữ liệu.

5. statistics.median(): Trung vị (giá trị ở giữa) của dữ liệu.

6. statistics.median\_low(): Trung vị thấp của dữ liệu.

7. statistics.median\_high(): Trung vị cao của dữ liệu.

8. statistics.median\_grouped(): Trung vị, hoặc phân vị 50, của dữ liệu được nhóm lại.

9. statistics.mode(): Mode đơn lẻ (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu biệt danh.

10. statistics.multimode(): Danh sách các mode (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu biệt danh.

11. statistics.quantiles(): Chia dữ liệu thành các khoảng có xác suất bằng nhau.

## II. Giải thích ý nghĩa của từng hàm

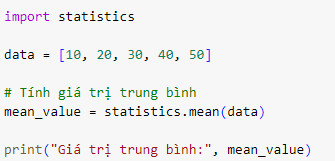
1. **statistics.mean(): Trung bình cộng của dữ liệu.**

- Hàm mean() dùng để tính giá trị trung bình của các số trong một mảng hoặc một danh sách. Cách truyền tham số và ý nghĩa từng tham số truyền vào như sau:

- Cú pháp của hàm là: statistics.mean(data)

- Tham số data: mảng hoặc danh sách chứa các số cần tính trung bình. Tham số này là bắt buộc và không thể bỏ trống. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

- Ví dụ, hàm mean() tính giá trị trung bình của dữ liệu [10, 20, 30, 40, 50], và kết quả là 30.0.



1. **statistics.fmean(): Trung bình cộng với số thập phân, tính nhanh hơn, có thể trọng số tùy chọn**

- Hàm statistics.fmean() dùng để tính trung bình số học nhanh chóng (fast mean) của dữ liệu. Nó là một phiên bản nhanh hơn của hàm mean()

- Cú pháp của hàm là: statistics.fmean(data, weights)

- Cách truyền tham số và ý nghĩa các tham số trong hàm statistics.fmean() như sau:

+ Tham số đầu tiên là `data`, là một sequence hoặc iterable chứa các số cần tính trung bình. Tham số này là bắt buộc và không thể bỏ trống. Nếu bỏ trống, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ Tham số thứ hai là `weights`, là một sequence hoặc iterable chứa các trọng số tương ứng với từng phần tử trong `data`. Tham số này là tùy chọn và mặc định là `None`. Nếu không được chỉ định, tất cả các phần tử trong `data` sẽ có trọng số bằng nhau và trung bình sẽ được tính dựa trên tần số xuất hiện của từng phần tử.

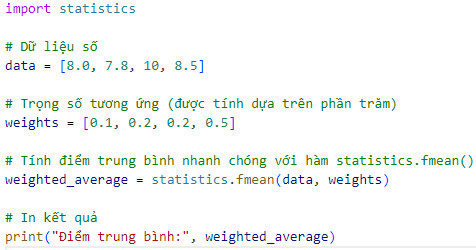
- Ví dụ tính điểm trung bình môn học:  
. 8.0 là điểm quá trình 10%

. 7.8 là điểm quá trình 20%

. 10 là điểm quá trình 20% (điểm giữa kì)

. 8.5 là điểm quá trình 50% (điểm cuối kì)

=>Vậy điểm trung bình là 8.61.

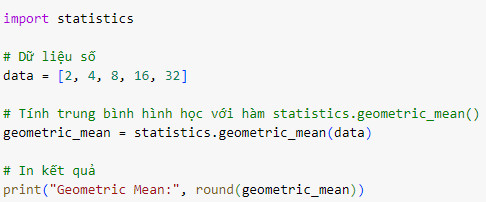


**3.statistics.geometric\_mean(): Trung bình hình học của dữ liệu.**

- Hàm statistics.geometric\_mean() được sử dụng để tính trung bình hình học của một tập hợp các số. Trung bình hình học là căn bậc thứ hai của tích của tất cả các số trong tập hợp. Hàm được sử dụng khi tính trung bình của các giá trị có tính chất nhân tích.

- Cú pháp của hàm là: statistics.geometric\_mean (data)

- Tham số data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị cần tính trung bình hình học. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).



- Trong ví dụ này, chúng ta tính trung bình hình học của các số [2, 4, 8, 16, 32], và kết quả là 8.0, tức là căn bậc hai của tích của tất cả các số trong tập hợp.

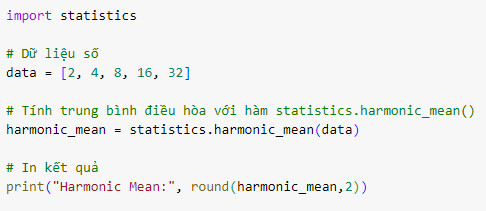
**4. statistics.harmonic\_mean(): Trung bình điều hoà của dữ liệu.**

- Hàm statistics.harmonic\_mean() được sử dụng để tính trung bình điều hòa (harmonic mean) của một tập hợp các số. Trung bình điều hòa là nghịch đảo của trung bình các nghịch đảo của các số trong tập hợp. Hàm này thường được sử dụng trong các tình huống đo lường tốc độ trung bình của một quá trình khi tốc độ thay đổi không đều.

- Cú pháp của hàm là: statistics.harmonic\_mean (data)

- Tham số data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị cần tính trung bình điều hòa. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

- Trong ví dụ này, chúng ta tính trung bình điều hòa của các số [2, 4, 8, 16, 32], và kết quả là 5.16, là nghịch đảo của trung bình các nghịch đảo của các số trong tập hợp.



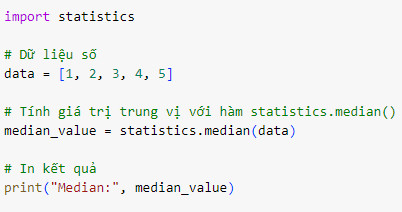
**5. statistics.median(): Trung vị (giá trị ở giữa) của dữ liệu.**

- Hàm statistics.median() dùng để tính giá trị trung vị (median) của một tập hợp các số. Trung vị là giá trị ở giữa của tập hợp số sau khi được sắp xếp theo thứ tự tăng dần. Nó là một phép đo trung tâm chính xác hơn so với trung bình (mean), đặc biệt hữu ích khi dữ liệu có nhiễu hoặc có các giá trị ngoại lệ (outliers).

- Cú pháp của hàm là: statistics.median(data)

- Tham số data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị cần tính trung vị. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

-Nếu số lượng phần tử trong data là lẻ, giá trị trung vị sẽ là phần tử ở vị trí giữa. Nếu số lượng phần tử data là chẵn, giá trị trung vị sẽ là trung bình của hai phần tử ở giữa.



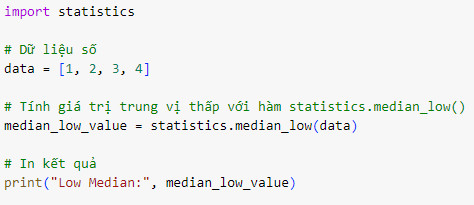
- Trong ví dụ này, chúng ta tính giá trị trung vị của các số [1, 2, 3, 4, 5], kết quả là 3, vì số lượng phần tử là lẻ và phần tử ở giữa là số 3.

**6. statistics.median\_low(): Trung vị thấp của dữ liệu.**

- Hàm statistics.median\_low() dùng để tính giá trị trung vị thấp (low median) của một tập hợp các số. Trung vị thấp là giá trị ở giữa của tập hợp số sau khi được sắp xếp theo thứ tự tăng dần. Nếu số lượng phần tử là chẵn, giá trị trung vị thấp sẽ là giá trị ở vị trí trước giữa của tập hợp.

- Cú pháp của hàm là: statistics.median\_low(data)

- Tham số data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị cần tính trung vị thấp. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).



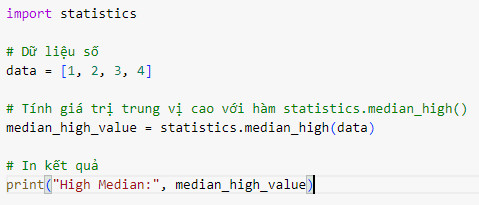
-Trong ví dụ này, chúng ta tính giá trị trung vị thấp của các số [1, 2, 3, 4], và kết quả là 2, vì số lượng phần tử là chẵn và giá trị ở vị trí trước giữa là số 2.

**7. statistics.median\_high(): Trung vị cao của dữ liệu.**

- Hàm statistics.median\_high() dùng để tính giá trị trung vị cao (high median) của một tập hợp các số. Trung vị cao là giá trị ở giữa của tập hợp số sau khi được sắp xếp theo thứ tự tăng dần. Nếu số lượng phần tử là chẵn, giá trị trung vị cao sẽ là giá trị ở vị trí sau giữa của tập hợp.

- Cú pháp của hàm là: statistics.median\_high(data)

- Tham số data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị cần tính trung vị cao. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).



- Trong ví dụ này, chúng ta tính giá trị trung vị cao của các số [1, 2, 3, 4], và kết quả là 3, vì số lượng phần tử là chẵn và giá trị ở vị trí sau giữa là số 3.

**8. statistics.median\_grouped(): Trung vị, hoặc phân vị 50, của dữ liệu được nhóm lại.**

- Hàm statistics.median\_grouped() dùng để tính giá trị trung vị (median) của một tập hợp các số khi dữ liệu đã được nhóm lại thành các khoảng (intervals). Phương pháp này thường được sử dụng khi dữ liệu liên tục và không đơn lẻ.

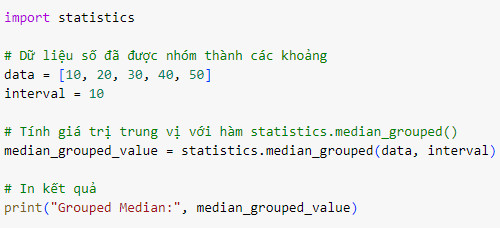
- Cú pháp của hàm là: statistics.median\_grouped(data, interval)

- Cách truyền và ý nghĩa các tham số trong hàm statistics.median\_grouped ( ) như sau:

+ data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị.. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ interval: Là khoảng (interval) giữa các giá trị liên tiếp. Đây là một giá trị số nguyên hoặc số thực dương.

- Trong ví dụ này, chúng ta tính giá trị trung vị của các số [10, 20, 30, 40, 50] đã được nhóm thành các khoảng 10, và kết quả là 30.0.



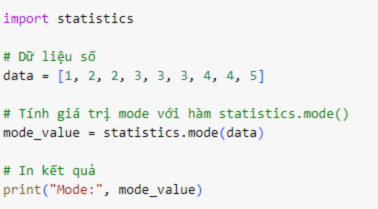
**9. statistics.mode(): Mode đơn lẻ (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu biệt danh.**

-Hàm statistics.mode() dùng để tính giá trị mode của một tập hợp dữ liệu. Mode là giá trị xuất hiện nhiều nhất trong dữ liệu và thường được sử dụng để xác định giá trị phổ biến nhất trong tập dữ liệu.

- Cú pháp của hàm là: statistics.mode (data)

+ data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị.. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

- Hàm statistics.mode() sẽ trả về giá trị mode của các số trong data. Nếu có nhiều giá trị có tần suất xuất hiện cao nhất và bằng nhau, thì hàm sẽ trả về giá trị đầu tiên trong danh sách các mode.



- Trong ví dụ này, chúng ta tính giá trị mode của các số [1, 2, 2, 3, 3, 3, 4, 4, 5], và kết quả là 3, vì giá trị 3 xuất hiện nhiều nhất trong dữ liệu.

**10. statistics.multimode(): Danh sách các mode (giá trị phổ biến nhất) của dữ liệu rời rạc hoặc dữ liệu biệt danh.**

- Hàm statistics.multimode() trong Python dùng để tính danh sách các giá trị mode của một tập hợp dữ liệu. Mode là giá trị xuất hiện nhiều nhất trong dữ liệu và thường được sử dụng để xác định giá trị phổ biến nhất trong tập dữ liệu.

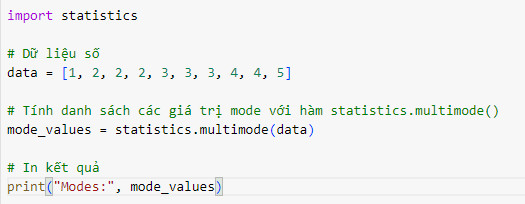
- Cú pháp của hàm là: statistics.multimode (data)

+ data: Là một tập hợp các số, có thể là một danh sách, tuple hoặc mảng chứa các giá trị.. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

- Hàm statistics.multimode() sẽ trả về một danh sách các giá trị mode của các số trong data. Nếu có nhiều giá trị có tần suất xuất hiện cao nhất và bằng nhau, thì hàm sẽ trả về danh sách các giá trị mode đó.

Trong ví dụ này, chúng ta tính danh sách các giá trị mode của các số

[1, 2, 2, 2, 3, 3, 3, 4, 4, 5], và kết quả là danh sách [2, 3], vì giá trị 2 và 3 đều xuất hiện nhiều nhất trong dữ liệu.



**11. statistics.quantiles(): Chia dữ liệu thành các khoảng có xác suất bằng nhau.**

- Hàm statistics.quantiles() trong Python dùng để chia dữ liệu thành các khoảng (quantiles) có cùng tần số (số lượng giá trị) như nhau. Quantiles được sử dụng để phân chia dữ liệu thành các phân vị (percentiles) hoặc tỷ lệ phần trăm.

- Cú pháp của hàm là: statistics.quantiles(data, n=4, method='exclusive')

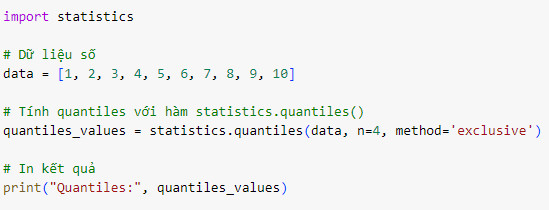
- Cách truyền và ý nghĩa các tham số trong hàm statistics.quantiles ( ) như sau:

+ data: Là một tập hợp các số hoặc giá trị, có thể là một danh sách, tuple hoặc mảng. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ 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'.

-Trong ví dụ này, chúng ta chia dữ liệu [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] thành 4 khoảng, và tính giá trị quantiles cho các khoảng đó. Kết quả là [2.75, 5.5, 8.25], tương ứng với phân vị 25%, 50%, 75%.



# Chương 2 : Đo lường về sự phân tán của dữ liệu

## I. Tổng quan về các hàm:

1.pstdev(): Đây là hàm tính độ lệch chuẩn của dữ liệu cho toàn bộ quần thể (population).

2.pvariance(): Đây là hàm tính phương sai của dữ liệu cho toàn bộ quần thể (population).

3. stdev(): Đây là hàm tính độ lệch chuẩn của dữ liệu cho mẫu (sample).

4. variance(): Đây là hàm tính phương sai của dữ liệu cho mẫu (sample).

## II. Giải thích ý nghĩa của từng hàm

1. **pstdev(): Đây là hàm tính độ lệch chuẩn của dữ liệu cho toàn bộ quần thể (population).**

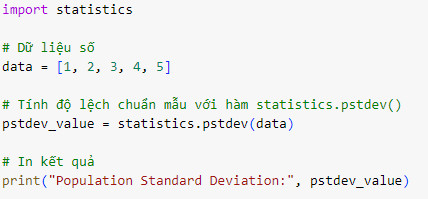
- Hàm pstdev() dùng để tính độ lệch chuẩn mẫu (population standard deviation) của dữ liệu. Độ lệch chuẩn mẫu là một phép đo độ biến đổi (dispersion) của dữ liệu, nó cho biết sự phân tán của các giá trị trong một tập hợp dữ liệu so với giá trị trung bình.

- Cú pháp của hàm là: statistics.pstdev(data, mu=None)

- Cách truyền và ý nghĩa các tham số trong hàm statistics.pstdev() như sau:

+ data: Là một tập hợp các số hoặc giá trị, có thể là một danh sách, tuple hoặc mảng. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ mu: Là giá trị trung bình của dữ liệu (tùy chọn). Nếu không được chỉ định, hàm sẽ tính giá trị trung bình của dữ liệu một cách tự động.





- Trong ví dụ này, chúng ta tính độ lệch chuẩn mẫu của dữ liệu [1, 2, 3, 4, 5], và kết quả là 1.4142135623730951.

1. **pvariance(): Đây là hàm tính phương sai của dữ liệu cho toàn bộ quần thể (population).**

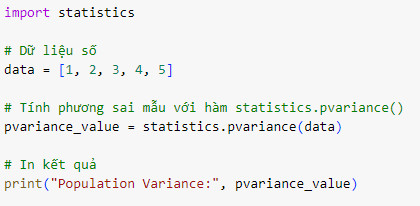
- Hàm pvariance() dùng để tính phương sai mẫu (population variance) của dữ liệu. Phương sai mẫu là một phép đo độ biến đổi (dispersion) của dữ liệu, nó cho biết sự phân tán của các giá trị trong một tập hợp dữ liệu so với giá trị trung bình.

- Cú pháp của hàm là: statistics.pvariance(data, mu=None)

- Cách truyền và ý nghĩa các tham số trong hàm statistics.pvariance() như sau:

+ data: Là một tập hợp các số hoặc giá trị, có thể là một danh sách, tuple hoặc mảng. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ mu: Là giá trị trung bình của dữ liệu (tùy chọn). Nếu không được chỉ định, hàm sẽ tính giá trị trung bình của dữ liệu một cách tự động.



Trong ví dụ này, chúng ta tính phương sai mẫu của dữ liệu [1, 2, 3, 4, 5], kết quả là 2.

1. **stdev(): Đây là hàm tính độ lệch chuẩn của dữ liệu cho mẫu (sample).**

Hàm stdev() dùng để tính độ lệch chuẩn mẫu (sample standard deviation) của dữ liệu.

Độ lệch chuẩn mẫu là một phép đo độ biến đổi (dispersion) của dữ liệu, nó cho biết sự phân tán của các giá trị trong một tập hợp dữ liệu so với giá trị trung bình.

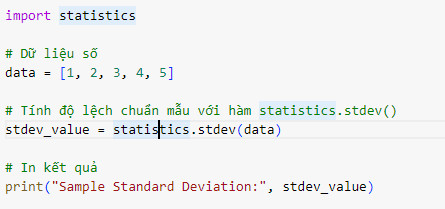
- Cú pháp của hàm là: statistics.stdev(data, xbar=None)

- Cách truyền và ý nghĩa các tham số trong hàm statistics.ptdev() như sau:

+ data: Là một tập hợp các số hoặc giá trị, có thể là một danh sách, tuple hoặc mảng. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ xbar: Là giá trị trung bình của dữ liệu (tùy chọn). Nếu không được chỉ định, hàm sẽ tính giá trị trung bình của dữ liệu một cách tự động.

- Trong ví dụ này, chúng ta tính độ lệch chuẩn mẫu của dữ liệu [1, 2, 3, 4, 5], và kết quả là 1.5811388300841898.


1. **variance(): Đây là hàm tính phương sai của dữ liệu cho mẫu (sample).**

- Hàm variance()dùng để tính phương sai mẫu (sample variance) của dữ liệu. Phương sai mẫu là một phép đo độ biến đổi (dispersion) của dữ liệu, nó đo lường sự phân tán của các giá trị trong một tập hợp dữ liệu so với giá trị trung bình.

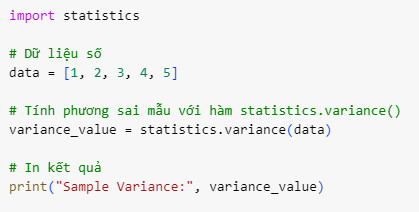
- Cú pháp của hàm là: statistics.stdev(data, xbar=None)

- Cách truyền và ý nghĩa các tham số trong hàm statistics.ptdev() như sau:

+ data: Là một tập hợp các số hoặc giá trị, có thể là một danh sách, tuple hoặc mảng. Nếu bỏ trống tham số này, sẽ ném ra ngoại lệ [StatisticsError](https://docs.python.org/3/library/statistics.html#statistics.StatisticsError).

+ xbar: Là giá trị trung bình của dữ liệu (tùy chọn). Nếu không được chỉ định, hàm sẽ tính giá trị trung bình của dữ liệu một cách tự động.

-Trong ví dụ này, chúng ta tính phương sai mẫu của dữ liệu [1, 2, 3, 4, 5], và kết quả là 2.5.



# Chương 3 : Thống kê cho mối quan hệ giữa hai đầu vào

## I. Tổng quan về các hàm:

1. covariance() - Hiệp phương sai mẫu cho hai biến số.

2. correlation() - Hệ số tương quan Pearson cho hai biến số.

3. linear\_regression() - Góc và điểm cắt cho phương trình đường thẳng tuyến tính đơn giản.

## II. Giải thích ý nghĩa của từng hàm

**1. covariance() - Hiệp phương sai mẫu cho hai biến số.**

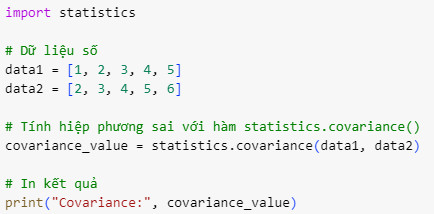
-Hàm covariance(data1, data2) trong Python dùng để tính hiệp phương sai (covariance) giữa hai tập hợp dữ liệu data1 và data2. Hiệp phương sai đo lường mức độ biến đổi đồng thời của hai tập dữ liệu và xác định xem chúng có xu hướng biến đổi cùng chiều hay trái chiều. Kết quả của hiệp phương sai là một giá trị số thực.

- Cú pháp của hàm là: statistics.covariance(data1, data2)

+ data1 và data2: Là hai tập hợp các số hoặc giá trị, có thể là các danh sách, tuple hoặc mảng. Hai tập hợp này phải có cùng số lượng phần tử.

+ Điều kiện để tính hiệp phương sai là hai dãy số data1 và data2 phải có cùng độ dài và không được ít hơn hai phần tử. Nếu không thỏa mãn điều kiện này, sẽ gây lỗi StatisticsError.

- Hàm covariance(data1, data2) trả về giá trị hiệp phương sai giữa hai tập dữ liệu data1 và data2.



- Trong ví dụ này, chúng ta tính hiệp phương sai giữa hai tập dữ liệu

data1 = [1, 2, 3, 4, 5] và data2 = [2, 3, 4, 5, 6], và kết quả là 2.5.

**2. correlation() - Hệ số tương quan Pearson cho hai biến số.**

Hàm statistics.correlation(x, y) trong Python dùng để tính hệ số tương quan (correlation coefficient) giữa hai tập hợp dữ liệu x và y. Hệ số tương quan đo lường mức độ tương quan tuyến tính giữa hai biến, và nằm trong khoảng từ -1 đến 1.

- Cú pháp của hàm là: statistics.correlation(x, y)

+ x và y: Là hai tập hợp các số hoặc giá trị, có thể là các danh sách, tuple hoặc mảng. Hai tập hợp này phải có cùng số lượng phần tử.

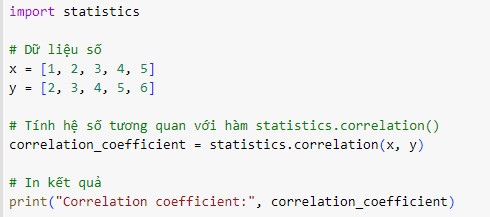
+ Điều kiện để tính hiệp phương sai là hai dãy số x và y phải có cùng độ dài và không được ít hơn hai phần tử. Nếu không thỏa mãn điều kiện này, sẽ gây lỗi StatisticsError.

- Hàm statistics.correlation(x, y) trả về giá trị hệ số tương quan giữa hai tập dữ liệu x và y. Hệ số tương quan đó có các ý nghĩa như sau:

+ Nếu hệ số tương quan gần bằng 1, tức là có một mối quan hệ tuyến tính mạnh giữa hai biến, và chúng có xu hướng biến đổi cùng chiều.

+ Nếu hệ số tương quan gần bằng -1, tức là có một mối quan hệ tuyến tính mạnh giữa hai biến, và chúng có xu hướng biến đổi trái chiều.

+ Nếu hệ số tương quan gần bằng 0, tức là không có mối quan hệ tuyến tính giữa hai biến.



Trong ví dụ này, chúng ta tính hệ số tương quan giữa hai tập dữ liệu x = [1, 2, 3, 4, 5] và y = [2, 3, 4, 5, 6], và kết quả là 1.0, cho thấy có một mối quan hệ tuyến tính mạnh giữa hai biến và chúng có xu hướng biến đổi cùng chiều.

**3. linear\_regression() - Góc và điểm cắt cho phương trình đường thẳng tuyến tính đơn giản.**

- Hàm statistics.linear\_regression(x, y, proportional=False) trong Python dùng để thực hiện phân tích hồi quy tuyến tính giữa hai tập hợp dữ liệu x và y. Phân tích hồi quy tuyến tính tìm ra một đường thẳng gần nhất biểu diễn mối quan hệ tuyến tính giữa hai biến.

- Cú pháp của hàm là: statistics.linear\_regression(x, y, proportional=False)

+ x và y: Là hai tập hợp các số hoặc giá trị, có thể là các danh sách, tuple hoặc mảng. Hai tập hợp này phải có cùng số lượng phần tử.

+ Điều kiện để tính hiệp phương sai là hai dãy số x và y phải có cùng độ dài và không được ít hơn hai phần tử. Nếu không thỏa mãn điều kiện này, sẽ gây lỗi StatisticsError.

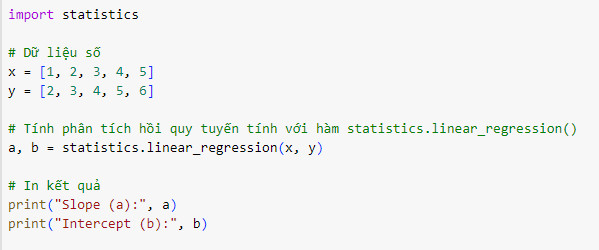
+ proportional: Tham số tùy chọn, kiểu dữ liệu boolean. Nếu proportional=True, hàm sẽ thực hiện hồi quy tuyến tính tỷ lệ, nghĩa là chỉ tìm đường thẳng đi qua gốc tọa độ (0,0). Nếu proportional=False (giá trị mặc định), hàm sẽ tìm đường thẳng tối ưu nhất đi qua toàn bộ tập hợp dữ liệu.

- Hàm statistics.linear\_regression(x, y) trả về kết quả của phân tích hồi quy tuyến tính là một tuple (a, b), trong đó:

a: Hệ số góc của đường thẳng hồi quy (slope).

b: Hệ số giao của đường thẳng hồi quy với trục y (intercept).

Trong ví dụ này, chúng ta thực hiện phân tích hồi quy tuyến tính giữa hai tập dữ liệu x = [1, 2, 3, 4, 5] và y = [2, 3, 4, 5, 6], và kết quả là a = 1.0 và b = 1.0, cho thấy đường thẳng hồi quy cần tìm là y = x + 1.



# Chương 4: Lớp statistics.NormalDist và các phép tính liên quan

## I. Tổng quát:

**1. Phân phối chuẩn**

- Phân phối chuẩn (normal distribution) là công cụ dùng để tạo và điều chỉnh phân phối chuẩn của biến ngẫu nhiên. Đây là một lớp (class) xử lý trung bình cộng và độ lệch chuẩn của các đo lường dữ liệu như một thực thể duy nhất.

- Phân phối chuẩn xuất hiện từ Định lý Trung tâm và có rất nhiều ứng dụng trong thống kê.

- Phương thức khởi tạo: statistics.NormalDist(mu=0.0, sigma=1.0)

+ Trả về một đối tượng NormalDist mới với mu là trung bình cộng, và sigma là độ lệch chuẩn.

+ Nếu sigma là số âm, sẽ gây ra lỗi StatisticsError.

**2. Các thuộc tính của lớp statistics.NormalDist:**

+ mean: Trả về giá trị trung bình của phân phối chuẩn (không thể thay đổi).

+ median: Trả về giá trị trung vị của phân phối chuẩn (không thể thay đổi).

+ mode: Trả về giá trị mode của phân phối chuẩn (không thể thay đổi).

+ stdev: Trả về giá trị độ lệch chuẩn của phân phối chuẩn (không thể thay đổi).

+ variance: Trả về giá trị phương sai của phân phối chuẩn, bằng bình phương của độ lệch chuẩn (không thể thay đổi).

**3. Phương thức từ mẫu dữ liệu:**

+ from\_samples(data): Tạo một đối tượng phân phối chuẩn với các tham số mu và sigma được ước tính từ dữ liệu sử dụng hàm fmean() và stdev().

+ samples(n, \*, seed=None): Tạo n mẫu ngẫu nhiên cho giá trị trung bình và độ lệch chuẩn đã cho. Trả về danh sách các giá trị float.

**4. Các phương thức xác suất:**

+ pdf(x): Sử dụng hàm mật độ xác suất (pdf), tính xác suất có liên quan đến giá trị x của biến ngẫu nhiên X. Xác suất liên quan là tỷ lệ giữa xác suất một mẫu xuất hiện trong khoảng nhỏ và chiều rộng của khoảng đó.

+ cdf(x): Sử dụng hàm phân phối tích lũy (cdf), tính xác suất mà một biến ngẫu nhiên X sẽ nhỏ hơn hoặc bằng giá trị x.

+ inv\_cdf(p): Tính hàm phân phối tích lũy nghịch đảo, còn được gọi là hàm phân vị hoặc hàm điểm phần trăm. Tìm giá trị x của biến ngẫu nhiên X sao cho xác suất của biến đó nhỏ hơn hoặc bằng giá trị p.

**5. Các phương thức thống kê:**

+ overlap(other): Đo lường mức độ trùng lắp giữa hai phân phối xác suất chuẩn. Trả về một giá trị trong khoảng từ 0.0 đến 1.0 thể hiện diện tích trùng lắp cho hai hàm mật độ xác suất.

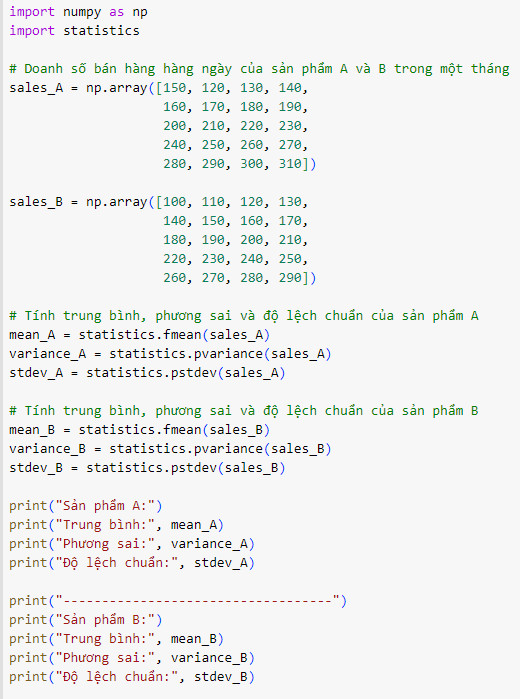
+ quantiles(n=4): Chia phân phối chuẩn thành n khoảng liên tục với xác suất bằng nhau. Trả về danh sách gồm (n - 1) điểm chia cách nhau.

+ zscore(x): Tính Điểm chuẩn mô tả giá trị x dựa trên số độ lệch chuẩn so

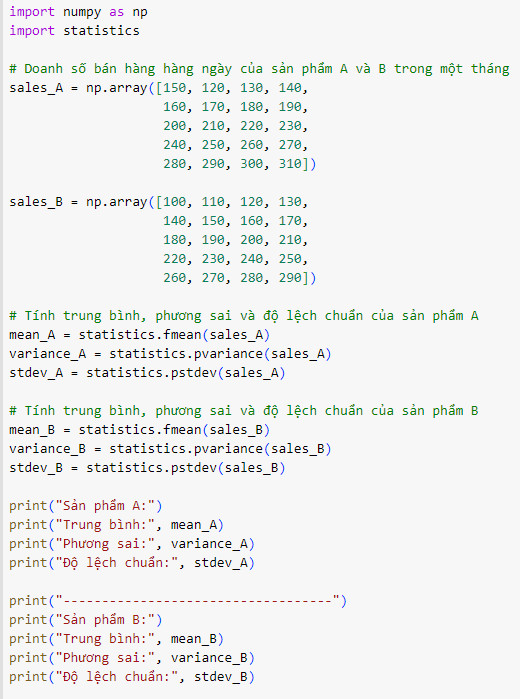
với giá trị trung bình của phân phối chuẩn: (x - mean) / stdev.

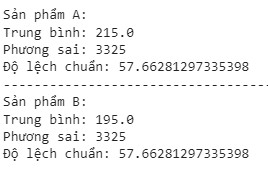
## II. Ví dụ code minh họa của lớp statistics.NormalDist :

Giả sử bạn là một nhà kinh doanh bán hàng trực tuyến và bạn muốn đánh giá hiệu suất của hai sản phẩm A và B. Bạn đã thu thập dữ liệu về doanh số bán hàng hàng ngày của hai sản phẩm trong một tháng. Dữ liệu được biểu diễn bởi hai mảng numpy như sau:



Bây giờ, bạn muốn tính toán trung bình, phương sai và độ lệch chuẩn của hai sản phẩm để so sánh hiệu suất bán hàng của chúng. Hãy sử dụng NormalDist để giải quyết bài toán này:





\* Dựa vào kết quả trung bình, phương sai và độ lệch chuẩn của hai sản phẩm A và B, chúng ta có thể so sánh hiệu suất bán hàng của chúng như sau:

- Trung bình (Mean): Trung bình của doanh số bán hàng cho sản phẩm A là 215.0, trong khi trung bình của doanh số bán hàng cho sản phẩm B là 195.0. Điều này cho thấy sản phẩm A có doanh số bán hàng trung bình cao hơn so với sản phẩm B.

- Phương sai (Variance) và Độ lệch chuẩn (Standard Deviation): Phương sai và độ lệch chuẩn đo lường mức độ biến động của doanh số bán hàng. Cả hai sản phẩm A và B đều có cùng một giá trị phương sai và độ lệch chuẩn, tức là mức độ biến động của doanh số bán hàng của hai sản phẩm là giống nhau.

-> Từ các thông số trên, chúng ta có thể kết luận rằng sản phẩm A có doanh số bán hàng trung bình cao hơn so với sản phẩm B, nhưng mức độ biến động của doanh số bán hàng của cả hai sản phẩm là tương tự.