Stats:

1. Cách Import?

from scipy import stats

1. Hàm Ngược Của CDF Của Phân Phối Student?

<Hoành Độ> = stats.t.ppf(<Xác Suất Tích Lũy>, <Độ Tự Do>)

* P(X ≤ <Hoành Độ>) = <Xác Suất Tích Lũy>, X ~ St(<Độ tự do>)

1. Hàm Ngược Của CDF Của Phân Phối Chuẩn Tắc?

<Hoành Độ> = stats.norm.ppf(<Xác Suất Tích Lũy>)

* P(X ≤ <Hoành Độ>) = <Xác Suất Tích Lũy>, X ~ N(0, 1)

1. CDF Của Phân Phối Chuẩn?

<Xác Suất Tích Lũy> = stats.norm.cdf(<Hoành Độ>, <Mean>, <Độ Lệch Chuẩn>)

* P(X ≤ <Hoành Độ>) = <Xác Suất Tích Lũy>, X ~ N(<Mean>, <Độ Lệch Chuẩn>2)

1. Hàm Ngược Của CDF Của Phân Phối Chi Bình Phương?

<Hoành Độ> = stats.chi2.ppf(<Xác Suất Tích Lũy>, <Độ Tự Do>)

* P(X ≤ <Hoành Độ>) = <Xác Suất Tích Lũy>, X ~ χ2(<Độ Tự Do>)

Signal:

1. Cách Import?

from scipy import signal

1. Trả Về Tương Quan Chéo?

signal.correlate(<Signal>, <Kernel>, <Mode>)

* <Signal> và <Kernel> đều là mảng có cùng số chiều
* <Mode> mặc định là "full"
* <Kernel> lúc đầu sẽ đặt ở vị trí sao cho phần tử cuối cùng của nó trùng với phần tử đầu tiên của <Signal>, sau đó trượt trên Signal từ trái sang phải, từ xa tới gần, từ trên xuống dưới, … từng phần tử một, trượt tới đâu, nó sẽ tính tích vô hướng của nó với phần <Signal> bị nó đè lên, ngoài biên của <Signal> mặc định = 0
* Ví dụ

a = [

[1, 2, 3, 4, 5],

[6, 7, 8, 9, 10],

[11, 12, 13, 14, 15]

]

b = [

[1, 2],

[3, 4]

]

foo = signal.correlate(a, b)

“foo” sẽ có giá trị là

[

[4, 11, 18, 25, 32, 15],

[26, 51, 61, 71, 81, 35],

[56, 101, 111, 121, 131, 55],

[22, 35, 38, 41, 44, 15]

]

* Nếu không muốn <Kernel> vượt biên <Signal> thì đặt <Mode> = "valid"
* Ví dụ

foo = signal.correlate(a, b, "valid")

* “foo” sẽ có giá trị là

[

[51, 61, 71, 81],

[101, 111, 121, 131]

]