

13.3_Integracio_Sympy

December 2, 2014



Figure 1: BY-SA

*Authors : Sonia Estradé
José M. Gómez
Ricardo Graciani
Manuel López
Xavier Luri
Josep Sabater*

1 Integració amb sympy

1.1 Integrals indefinides

Sympy permet fer integració simbòlica de funcions (integrals indefinides) mitjançant la funció `integrate()`, que reb com a argument una expressió simbòlica.

```
In [1]: import sympy as sp
        sp.init_printing()

        sp.var("x")

        print( "Integrand sin(x)*exp(x)" )
        sp.integrate(sp.sin(x)*sp.exp(x),x)
```

Integrand sin(x)*exp(x)

Out[1]:

$$\frac{e^x}{2} \sin(x) - \frac{e^x}{2} \cos(x)$$

Aquesta mateixa funció permet fer integrals múltiples:

```
In [2]: import sympy as sp
        sp.init_printing()

        sp.var("x")
        sp.var("y")
```

```
print( "Integrand exp(-x**2-y**2)" )
sp.integrate(sp.exp(-x**2 - y**2),x,y)
```

Integrand exp(-x**2-y**2)

Out[2]:

$$\frac{\pi}{4} \operatorname{erf}(x) \operatorname{erf}(y)$$

1.2 Integrals definides

La mateixa funció `integrate()` permet també calcular integrals definides indicant els límits d'integració. Per exemple la integral

$$\int_0^\pi \sin(x) dx = -\cos(x)|_0^\pi = 2$$

s'implementa com:

```
In [3]: import sympy as sp
        sp.init_printing()

        sp.var("x")

        sp.integrate(sp.sin(x), (x, 0, sp.pi))
```

Out[3]:

$$2$$

Es pot indicar un límit d'integració infinit amb el símbol `sympy.oo`. L'exemple següent implementa la integral:

$$\int_0^\infty e^{-x} dx = e^{-x}|_0^\infty = 1$$

```
In [4]: import sympy as sp
        sp.init_printing()

        sp.var("x")

        sp.integrate(sp.exp(-x), (x, 0, sp.oo))
```

Out[4]:

$$1$$

De forma similar es poden calcular integrals dobles definides. Per exemple, la integral:

$$\int_0^\infty \int_0^\infty e^{-x^2-y^2} dx dy = \pi$$

```
In [5]: import sympy as sp
        sp.init_printing()

        sp.var("x")
        sp.var("y")

        sp.integrate(sp.exp(-x**2 - y**2), (x, -sp.oo, sp.oo), (y, -sp.oo, sp.oo))
```

Out[5]:

$$\pi$$

Els límits d'integració poden ser símbols, i el resultat s'expressa en funció d'ells. Per exemple

$$\int_0^a \sin(x) dx = -\cos(a) + 1$$

```
In [6]: import sympy as sp
        sp.init_printing()

        sp.var("x")
        sp.var("a")

        sp.integrate(sp.sin(x), (x, 0, a))
```

Out[6]:

$$-\cos(a) + 1$$

En l'exemple següent el resultat indica dues possibilitats per què la integral no convergeix si la part real de a no és > 1

```
In [7]: import sympy as sp
        sp.init_printing()

        sp.var("x")
        sp.var("a")

        sp.integrate(x**a*sp.exp(-x), (x, 0, sp.oo))
```

Out[7]:

$$\begin{cases} \Gamma(a+1) & \text{for } -\Re a < 1 \\ \int_0^\infty x^a e^{-x} dx & \text{otherwise} \end{cases}$$

1.3 Integrals com a símbols

En cas que la integral no es vulgui avaluar, obtenint la funció o el valor numèric corresponent, es pot usar `Integral()` que retorna la integral com una expressió simbòlica de *sympy*. Posteriorment la integral es pot avaluar usant `doit()`

Per exemple, en el cas d'integració definida:

```
In [8]: import sympy as sp
        sp.init_printing()

        sp.var("x")

        resultat = sp.Integral(sp.exp(-x**2 - y**2), (x, -sp.oo, sp.oo), (y, -sp.oo, sp.oo))
        resultat, resultat.doit()
```

Out[8]:

$$\left(\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-x^2 - y^2} dx dy, \pi \right)$$

I un exemple en el cas d'integració simbòlica:

```
In [9]: import sympy as sp
        sp.init_printing()

        sp.var("x")

        resultat = sp.Integral((x**4 + x**2*sp.exp(x) - x**2 - 2*x*sp.exp(x) - 2*x - sp.exp(x))*sp.exp(x),
                                (x, 0, sp.oo))
        resultat, resultat.doit()
```

Out [9]:

$$\left(\int \frac{(x^4 + x^2 e^x - x^2 - 2x e^x - 2x - e^x) e^x}{(x-1)^2 (x+1)^2 (e^x + 1)} dx, \log(e^x + 1) + \frac{e^x}{x^2 - 1} \right)$$

També podem usar *evalf()* per obtenir un resultat numèric amb precisió arbitrària

```
In [10]: import sympy as sp
         sp.init_printing()
```

```
         sp.var("x")
```

```
         resultat = sp.Integral(sp.exp(-x**2), (x, -sp.oo, sp.oo))
```

```
         #resultat, resultat.doit()
```

```
         print( resultat.doit().evalf(100)**2 )
```

```
         print( sp.pi.evalf(100) )
```

```
         resultat = sp.Integral(sp.exp(-x**2), (x, -1, 1))
```

```
         print( resultat.doit(), resultat.doit().evalf(100) )
```

```
3.141592653589793238462643383279502884197169399375105820974944592307816406286208998628034825342117068
3.141592653589793238462643383279502884197169399375105820974944592307816406286208998628034825342117068
sqrt(pi)*erf(1) 1.4936482656248540507989348722637060107089993736252126580553089979172106551235456629956
```