How to get the Gradient and Hessian | Sympy

Asked 3 years ago Active 3 years ago Viewed 5k times

7

1

Here is the situation: I have a symbolic function **lamb** which is function of the elements of the variable z and the functions elements of the variable h. Here is an image of the lamb symbolic function

$$\frac{1}{2\sigma^2}\left((z_1 - \mathrm{h}_1(\xi,\eta))^2 + (z_2 - \mathrm{h}_2(\xi,\eta))^2 + (z_3 - \mathrm{h}_3(\xi,\eta))^2\right).$$

Now I would like the compute the Gradient and Hessian of this function with respect to the variables eta and xi. Of course I googled for it but I could not find a straight way for doing this. What I found is here, but as I said it does not seen to be the best approach for this situation. Any idea? Bellow, the source code. Thanks.

```
from sympy import Symbol, Matrix, Function, simplify

eta = Symbol('eta')
xi = Symbol('xi')

x = Matrix([[xi],[eta]])

h = [Function('h_'+str(i+1))(x[0],x[1]) for i in range(3)]
z = [Symbol('z_'+str(i+1)) for i in range(3)]

lamb = 0
for i in range(3):
    lamb += 1/(2*sigma**2)*(z[i]-h[i])**2
simplify(lamb)
```

python-3.x sympy Edit tags

asked Sep 18 '16 at 14:09

Randerson

338 2 14

2 Answers

9

You could either use the very Pythonic way suggested by Stelios, or use some recently added features to SymPy:

```
In [14]: from sympy.tensor.array import derive_by_array

In [15]: derive_by_array(lamb, (eta, xi))
Out[15]:
```

```
[-(z_1 - h_1(xi, eta))*Derivative(h_1(xi, eta), eta)/sigma**2 - (z_2 - h_2(xi,
 eta))*Derivative(h_2(xi, eta), eta)/sigma**2 - (z_3 - h_3(xi, eta))*Derivativ
e(h_3(xi, eta), eta)/sigma**2, -(z_1 - h_1(xi, eta))*Derivative(h_1(xi, eta),
xi)/sigma**2 - (z_2 - h_2(xi, eta))*Derivative(h_2(xi, eta), xi)/sigma**2 - (z
_3 - h_3(xi, eta))*Derivative(h_3(xi, eta), xi)/sigma**2]
```

Unfortunately the printer is still missing for N-dim arrays, you can visualize by converting them to a list (or, alternatively, using *.tomatrix()*):

```
In [16]: list(derive_by_array(lamb, (eta, xi)))
Out[16]:
⎡                      ∂                                ∂
⎢  (z₁ - h₁(ξ, η))⋅──(h₁(ξ, η))   (z₂ - h₂(ξ, η))⋅──(h₂(ξ, η))   (z₃ - h₃(ξ, η
⎢                      ∂η                               ∂η
⎢- ──────────────────────────── - ──────────────────────────── - ─────────────
⎢                  2                              2
⎣                 σ                              σ

   ∂                                 ∂                                 ∂
))⋅──(h₃(ξ, η))   (z₁ - h₁(ξ, η))⋅──(h₁(ξ, η))   (z₂ - h₂(ξ, η))⋅──(h₂(ξ, η))
   ∂η                                ∂ξ                                ∂ξ
───────────────, - ──────────────────────────── - ────────────────────────────
 2                               2                               2
σ                               σ                               σ

                       ∂              ⎤
   (z₃ - h₃(ξ, η))⋅──(h₃(ξ, η))⎥
                       ∂ξ             ⎥
 - ────────────────────────────⎥
                  2                   ⎥
                 σ                    ⎦
```

For the Hessian, just repeat the procedure twice:

```
In [18]: list(derive_by_array(derive_by_array(lamb, (eta, xi)), (eta, xi)))
Out[18]:
⎡                        2                                 2
⎢                       ∂                                 ∂
⎢  (z₁ - h₁(ξ, η))⋅───(h₁(ξ, η))   (z₂ - h₂(ξ, η))⋅───(h₂(ξ, η))   (z₃ - h₃(ξ,
⎢                        2                                 2
⎢                      ∂η                                ∂η
⎢- ───────────────────────────── - ───────────────────────────── - ───────────
⎢                  2                                 2
⎣                 σ                                 σ

        2
       ∂                 ⎛∂             ⎞2   ⎛∂             ⎞2   ⎛∂             ⎞2
 η))⋅───(h₃(ξ, η))   ⎜──(h₁(ξ, η))⎟   ⎜──(h₂(ξ, η))⎟   ⎜──(h₃(ξ, η))⎟    (z
        2            ⎝∂η            ⎠   ⎝∂η            ⎠   ⎝∂η            ⎠
      ∂η
────────────────── + ─────────────── + ─────────────── + ───────────────, - ──
    2                       2                  2                  2
   σ                       σ                  σ                  σ

                 2                                  2
                ∂                                  ∂
```

```
₁ - h₁(ξ, η))⋅───────(h₁(ξ, η))   (z₂ - h₂(ξ, η))⋅───────(h₂(ξ, η))   (z₃ - h₃(ξ,
              ∂ξ ∂η                               ∂ξ ∂η
──────────────────────────────── - ────────────────────────────────── - ────────────
               2                                  2
              σ                                  σ


        2
       ∂                      ∂                ∂                 ∂                ∂
η))⋅───────(h₃(ξ, η))   ──(h₁(ξ, η))⋅──(h₁(ξ, η))   ──(h₂(ξ, η))⋅──(h₂(ξ, η))
     ∂ξ ∂η              ∂η               ∂ξ               ∂η               ∂ξ
───────────────────── + ──────────────────────────── + ──────────────────────────── +
    2                               2                              2
   σ                               σ                              σ


                                                       2
∂              ∂                                      ∂
──(h₃(ξ, η))⋅──(h₃(ξ, η))    (z₁ - h₁(ξ, η))⋅───────(h₁(ξ, η))   (z₂ - h₂(ξ, η))
∂η             ∂ξ                               ∂ξ ∂η
────────────────────────────, - ────────────────────────────────── - ────────────────
             2                                  2
            σ                                  σ


    2                                   2
   ∂                                   ∂                       ∂               ∂
⋅───────(h₂(ξ, η))   (z₃ - h₃(ξ, η))⋅───────(h₃(ξ, η))   ──(h₁(ξ, η))⋅──(h₁(ξ, η))
 ∂ξ ∂η                                  ∂ξ ∂η              ∂η               ∂ξ
────────────────── - ────────────────────────────────── + ────────────────────────────
 2                                  2                                 2
σ                                  σ                                 σ


                                                                                       ∂
   ∂             ∂                ∂               ∂                 (z₁ - h₁(ξ, η))⋅──
   ──(h₂(ξ, η))⋅──(h₂(ξ, η))    ──(h₃(ξ, η))⋅──(h₃(ξ, η))
   ∂η              ∂ξ              ∂η              ∂ξ                               ∂ξ
 + ──────────────────────────── + ────────────────────────────, - ────────────────────
               2                              2                                  2
              σ                              σ                                  σ

2                                   2                                   2
                                   ∂                                   ∂
─(h₁(ξ, η))   (z₂ - h₂(ξ, η))⋅───(h₂(ξ, η))   (z₃ - h₃(ξ, η))⋅───(h₃(ξ, η))
2                                   2                                   2
                                  ∂ξ                                  ∂ξ
──────────── - ───────────────────────────────── - ───────────────────────────────── +
                                2                                   2
                               σ                                   σ

                                                          ⎤
⎛               ⎞2   ⎛               ⎞2   ⎛               ⎞2⎥
⎜∂              ⎟    ⎜∂              ⎟    ⎜∂              ⎟ ⎥
⎜──(h₁(ξ, η))⎟    ⎜──(h₂(ξ, η))⎟    ⎜──(h₃(ξ, η))⎟ ⎥
⎝∂ξ             ⎠    ⎝∂ξ             ⎠    ⎝∂ξ             ⎠ ⎥
───────────────── + ───────────────── + ─────────────────⎥
        2                    2                    2        ⎥
       σ                    σ                    σ         ⎦
```

answered Sep 19 '16 at 21:18
Francesco Bonazzi
1,647 5 8

10

You can simply compute the gradient vector "manually" (assuming that the variables are ordered as `(z1, z2, z3, eta)`):

```
[lamb.diff(x) for x in z+[eta]]
```

Similarly, for the Hessian matrix:

```
[[lamb.diff(x).diff(y) for x in z+[eta]] for y in z+[eta]]
```

answered Sep 18 '16 at 14:47
Stelios
3,726 1 9 21