ScientificPython

February 19, 2020

[136]:

```
import numpy as np
from scipy import interpolate
import matplotlib.pyplot as plt
```

[137]:

```
def noise(nnodes, amplitude):
        x = np.linspace(0, 1, nnodes) # numero di nodi da interpolare spaziati␣
 ↪lineari
        y = x
        z = amplitude * np.random.uniform(-1, 1, (nnodes, nnodes)) # y cas -1␣
 ↪mare 0 terra 1 montagna

        return interpolate.RectBivariateSpline(x, y, z)
```

[138]:

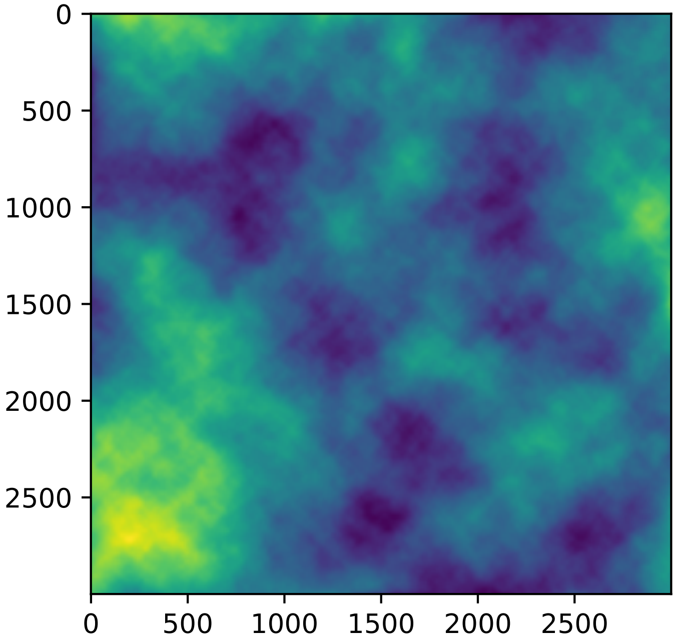
```
class Noise(object):

    def __init__(self, size, details):
        self._f = [noise(2**(size + n), 1./2**n) for n in range(details)]

    def __call__(self, x, y):
        return sum([f(x,y) for f in self._f])
```

[146]:

```
f = Noise(2, 6)
```

[148]:

```
x = np.linspace(0, 1, 3000)
y = x
plt.imshow(f(x, y))
plt.savefig("noise.png")
```


0
500
1000
1500
2000
2500
0
500
1000
1500
2000
2500

[ ]: