lec4_step1_NumPy_Samples_TFarray_Function

November 30, 2022

[ ]:
```
## Python basics for novice data scientists, supported by Wagatsuma Lab@Kyutech
#
# The MIT License (MIT): Copyright (c) 2020 Hiroaki Wagatsuma and Wagatsuma␣
 ↪Lab@Kyutech
#
# Permission is hereby granted, free of charge, to any person obtaining a copy␣
 ↪of this software and associated documentation files (the "Software"), to␣
 ↪deal in the Software without restriction, including without limitation the␣
 ↪rights to use, copy, modify, merge, publish, distribute, sublicense, and/or␣
 ↪sell copies of the Software, and to permit persons to whom the Software is␣
 ↪furnished to do so, subject to the following conditions:
# The above copyright notice and this permission notice shall be included in␣
 ↪all copies or substantial portions of the Software.
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR␣
 ↪IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,␣
 ↪FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE␣
 ↪AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER␣
 ↪LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING␣
 ↪FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS␣
 ↪IN THE SOFTWARE. */
#
# # @Time    : 2020-11-30
# # @Author  : Hiroaki Wagatsuma
# # @Site    : https://github.com/hirowgit/2A1_python_intermediate_course
# # @IDE     : Python 3.9.14 (main, Sep  6 2022, 23:29:09) [Clang 13.1.6␣
 ↪(clang-1316.0.21.2.5)] on darwin
# # @File    : lec4_step1_NumPy_Samples_TFarray_Function.py
```

[139]:
```
import numpy as np
#prFill=[90    60    50    50    50    90    40    30    80    40  20 ]/100;
prFill=np.array([90, 60, 50, 50, 50, 90, 40, 30, 80, 40, 20])
prFill=prFill/100
print(prFill)
```

```
[0.9 0.6 0.5 0.5 0.5 0.9 0.4 0.3 0.8 0.4 0.2]
```

[140]:
```
# fillLine=boolean(ones(1,size(prFill,2)));
# 1
fillLine=np.full(len(prFill),True)
print(fillLine)
```

```
[ True  True  True  True  True  True  True  True  True  True  True]
```

[88]:
```
# fillLine=boolean(ones(1,size(prFill,2)));
# 1    :       x
fillLine=np.full(( 1,len(prFill)),True)
print(fillLine)
```

```
[[ True  True  True  True  True  True  True  True  True  True  True]]
```

[23]:
```
#
fillLine2=np.empty(len(prFill), dtype = bool)
fillLine2[:]=True  #
print(fillLine2)
```

```
[ True  True  True  True  True  True  True  True  True  True  True]
```

[41]:
```
a=np.zeros((2,3))
print(a)
print(len(a))
print(a.size)
print(a.ndim)
print(a.shape)
```

```
[[0. 0. 0.]
 [0. 0. 0.]]
2
6
2
(2, 3)
```

[145]:
```
b=np.argwhere(prFill>0.8)
print(b)
print(b.shape)
```

```
[[0]
 [5]]
(2, 1)
```

[143]:
```
prFill
```

[143]:
```
array([0.9, 0.6, 0.5, 0.5, 0.5, 0.9, 0.4, 0.3, 0.8, 0.4, 0.2])
```

[152]:
```
c=np.where(prFill>0.8)
print(c)
print(np.shape(c))
```

```
(array([0, 5]),)
(1, 2)
```

[47]:
```
np.argwhere((prFill>0.8) & fillLine)
```

[47]:
```
array([[0, 0],
       [0, 5]])
```

[156]:
```
fillLine[5]=False
np.where((prFill>0.8) & fillLine)
```

[156]:
```
(array([0]),)
```

[158]:
```
fillLine
```

[158]:
```
array([ True,  True,  True,  True,  True, False,  True,  True,  True,
        True,  True])
```

[159]:
```
prFill
```

[159]:
```
array([0.9, 0.6, 0.5, 0.5, 0.5, 0.9, 0.4, 0.3, 0.8, 0.4, 0.2])
```

[54]:
```
prFill[(prFill>0.8) ]
```

[54]:
```
array([0.9, 0.9])
```

[55]:
```
np.where((prFill>0.8) & (prFill>=0.8))
```

[55]:
```
(array([0, 5]),)
```

[90]:
```
b=np.where(np.logical_and(prFill>0.8,fillLine))
print(b)
print(len(b))
```

```
(array([0, 5]),)
1
```

[69]:
```
b2=np.unique(b)
print(b2)
```

```
[0 5]
```

[71]:
```
b2[0]
```

[71]:
```
0
```

[91]:
```
print(fillLine[3:-1])
```

```
[ True  True  True  True  True  True  True]
```

[135]:
```
i=3
np.where((prFill[i+1:-1]>0.8) & fillLine[i+1:-1])
```

[135]:
```
(array([1]),)
```

[137]:
```
i=3
remF=0.5
IDrem=np.where((prFill[i+1:-1]>remF) & fillLine[i+1:-1])
print(IDrem)
```

```
(array([1, 4]),)
```

[134]:
```
a=np.empty(0)
print(a)
a.append([0,1])
print(a)
np.append(a,np.array([4])
print(a)
np.append(a,np.array([1,2,3])
```

```
  File "<ipython-input-134-26a9b3048070>", line 6
    print(a)
    ^
SyntaxError: invalid syntax
```

[130]:
```
a=[]
print(a)
a.append([0,1])
print(a)
a.append([1,2,3])
print(a)
```

```
[]
[[0, 1]]
[[0, 1], [1, 2, 3]]
```

[129]:
```
gg=[[1,2],[0,1,2],[5]]
print(gg)
print(gg[0])
gg.append([1,2,1,1,3])
```

```
print(gg)
```

```
[[1, 2], [0, 1, 2], [5]]
[1, 2]
[[1, 2], [0, 1, 2], [5], [1, 2, 1, 1, 3]]
```

[ ]: