lec4_step8_BarStack_Aligned_Stage7

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_step8_BarStack_Aligned_Stage7.py
```

[1]:
```
import numpy as np
import matplotlib.pyplot as plt

#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
fillLine=np.full(len(prFill),True)
LineT=[]
tmp=[]
```

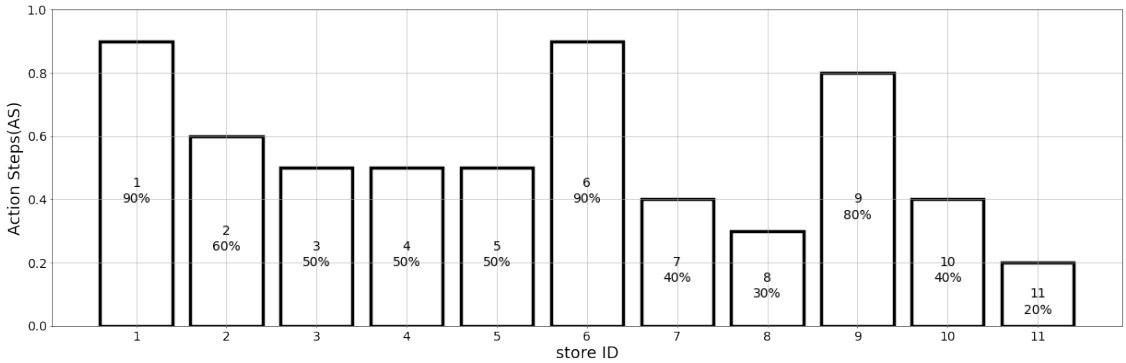
```
k=0
for i in range(len(prFill)):
#for i in range(5):
#for i in range(5):
    if fillLine[i]:
        remF=1-prFill[i]
        IDrem=np.where((prFill[i+1:len(prFill)]<=remF) & fillLine[i+1:
 ↪len(prFill)])
        tmp=i
        fID=i
        #j=0
        while IDrem[0].size > 0:
            fID=IDrem[0][0]+fID+1
            tmp=np.append(tmp,fID)
            remF=remF-prFill[fID]
            IDrem=np.where((prFill[fID+1:len(prFill)]<=remF) & fillLine[fID+1:
 ↪len(prFill)])
        LineT.append(tmp)
        fillLine[tmp]=False
#         print("k;",k)
#         print("LineT;",LineT)
        k=k+1
#         print("k;",k)

# int  'numpy.ndarray'
f_LineT = [np.array(i) if type(i)==int else i for i in LineT]
#lenLineT=cell2mat(cellfun(@(x) length(x),LineT,'UniformOutput',false));
lenLineT = [i.size for i in f_LineT]
#stackBarD=zeros(size(LineT,2),max(lenLineT));
stackBarD = np.zeros((np.shape(f_LineT)[0],max(lenLineT)))
#stackBarD     prFill
for i in range(len(f_LineT)):
    tmp = f_LineT[i]
    stackBarD[i,0:lenLineT[i]] = prFill[tmp]
print(stackBarD)

y_data_stack = []
y_data_stack = tuple([np.append(y_data_stack, i)  for i in stackBarD])

x_label = [i+1 for i in range(len(prFill))]
x_stack_label = [i+1 for i in range(len(stackBarD))]
y_label = np.arange(0, 12, 2)
y_label =[i/10 for i in y_label]

#2
fig = plt.figure(figsize=(30,20), dpi=50)
init_fig = fig.add_subplot(2 , 1, 1)
```

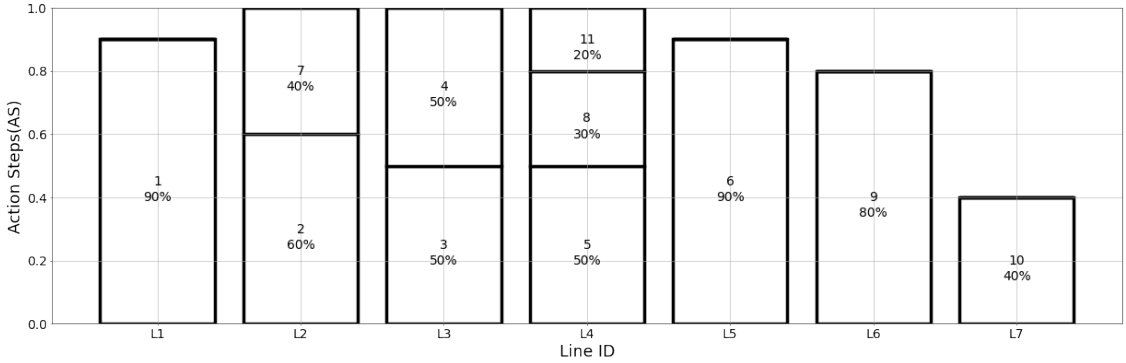
```
stack_fig = fig.add_subplot(2, 1, 2)

#
#  https://www.yutaka-note.com/entry/matplotlib_axis

init_fig.set_xlabel("store ID", size = 25)
init_fig.set_xticks(x_label)
init_fig.set_xticklabels(x_label, size=20)

init_fig.set_ylabel("Action Steps(AS)", size = 25)
init_fig.set_yticks(y_label)
init_fig.set_yticklabels(y_label, size=20)
init_fig.set_ylim(0 , 1)
init_fig.grid(True)
#
bar = init_fig.bar(x_label, prFill, color = 'w', edgecolor ='black', linewidth␣
 ↪= '5')

#   BAR

for i in range(len(bar)):
        cx = bar[i].get_x() + bar[i].get_width() / 2
        cy = bar[i].get_y() + bar[i].get_height() / 2
        init_fig.text(cx, cy, x_label[i], size= 20,  color="k", ha="center",␣
 ↪va="center")
        init_fig.text(cx, cy-0.05, str(f'{prFill[i]*100:.0f}') +'%',size= 20, ␣
 ↪color="k", ha="center", va="center")

#
#  https://www.yutaka-note.com/entry/matplotlib_axis

stack_fig.set_xlabel("Line ID", size = 25)
stack_fig.set_xticks(x_stack_label)
stack_fig.set_xticklabels(list(map(lambda label:'L' + str(label),␣
 ↪x_stack_label)), size=20)

stack_fig.set_ylabel("Action Steps(AS)", size = 25)
stack_fig.set_yticks(y_label)
stack_fig.set_yticklabels(y_label, size=20)
stack_fig.set_ylim(0 , 1)
stack_fig.grid(True)
#
bottom = np.zeros(stackBarD.T.shape[1])

for i in range(stackBarD.T.shape[0]):
    if i ==0:
```

```
        s_bar = stack_fig.bar(x_stack_label, stackBarD.T[i], color = 'w',␣
↪edgecolor ='black', linewidth = '5')
    else:
        s_bar = stack_fig.bar(x_stack_label, stackBarD.T[i], bottom= bottom,␣
↪color = 'w', edgecolor ='black', linewidth = '5')
    bottom = np.add(bottom, stackBarD.T[i])

for i in range(stackBarD.shape[0]):
    baseY=0
    for j in range(stackBarD.shape[1]):
        if stackBarD[i][j]>0:
            if f_LineT[i].size ==1:
                key = f_LineT[i]
                tmp = prFill[key]
            else:
                key = f_LineT[i][j]
                tmp = prFill[key]
            ypos = tmp/2
            stack_fig.text(s_bar[i].get_x() + s_bar[i].get_width() / 2, baseY+␣
↪ypos, str(key+1),size= 20,  color="k", ha="center", va="center")
            stack_fig.text(s_bar[i].get_x() + s_bar[i].get_width() / 2, baseY+␣
↪ypos - 0.05, str(f'{tmp*100:.0f}') +'%',size= 20,  color="k", ha="center",␣
↪va="center")
            baseY = baseY+tmp
```

```
/usr/local/lib/python3.9/site-packages/numpy/core/fromnumeric.py:1970:
VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences
(which is a list-or-tuple of lists-or-tuples-or ndarrays with different lengths
or shapes) is deprecated. If you meant to do this, you must specify
'dtype=object' when creating the ndarray.
  result = asarray(a).shape
```

```
[[0.9 0.  0. ]
 [0.6 0.4 0. ]
 [0.5 0.5 0. ]
 [0.5 0.3 0.2]
 [0.9 0.  0. ]
 [0.8 0.  0. ]
 [0.4 0.  0. ]]
```


Action Steps(AS)
store ID
1
90%
2
60%
3
50%
4
50%
5
50%
6
90%
7
40%
8
30%
9
80%
10
40%
11
20%


Action Steps(AS)
Line ID
L1
L2
L3
L4
L5
L6
L7
1
90%
7
40%
2
60%
4
50%
3
50%
11
20%
8
30%
5
50%
6
90%
9
80%
10
40%

[ ]: