How can I add textures to my bars and wedges?

I'm drawing several bar and pie charts using `matplotlib.pyplot.bar()` and `matplotlib.pyplot.pie()`. In both functions, I can change the colors of the bars and wedges.

However, I need to print these charts in black and white. It would be much more useful to be able to put textures on the bars and wedges, similar to the `Line2D` marker property which is available for drawing lines. Can I maybe fill the bars and wedges with these markers in a consistent way? Or is there any other way to achieve something like that?

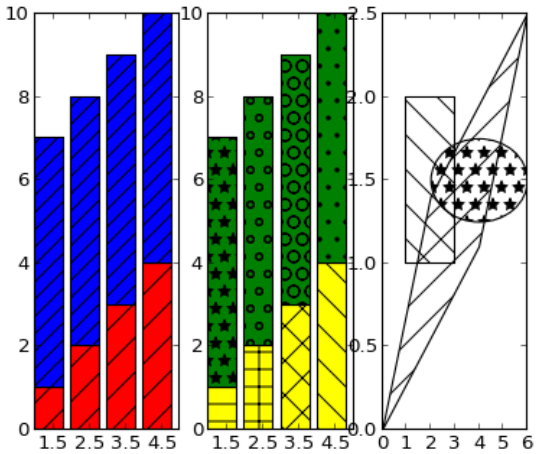
python matplotlib

asked Jan 11 '13 at 13:37

pemistahl
2,568 10 43

2 Answers

With `bar()`, you can directly use hatches (with some backends):
http://matplotlib.org/examples/pylab_examples/hatch_demo.html:



It works by adding the `hatch` argument to your call to `bar()`.

As for `pie()`, it does not have a `hatch` keyword. You can instead get the individual pie chart patches and add hatches to them: you get the patches with:

```
patches = pie(…)[0]  # The first element of the returned tuple are the pie slices
```

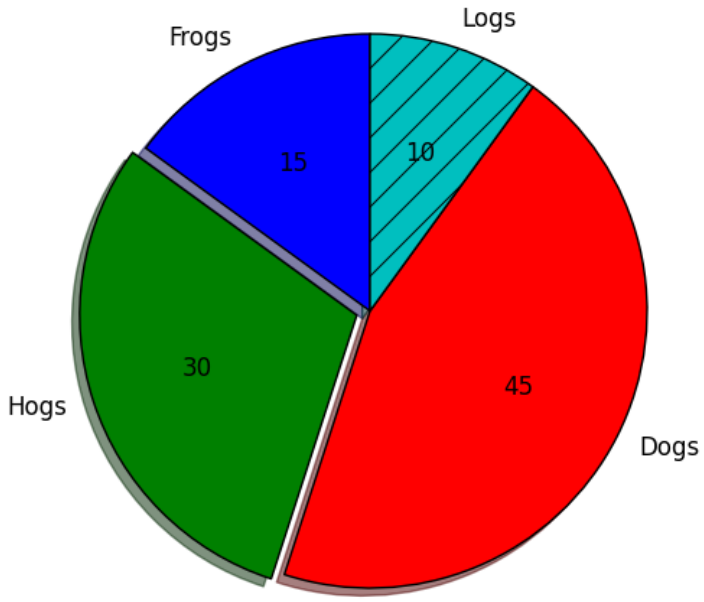
then you apply the hatches to each slice (patch):

```
patches[0].set_hatch('/')  # Pie slice #0 hatched.
```

(hatches list at http://matplotlib.org/api/artist_api.html#matplotlib.patches.Patch.set_hatch).

And you apply the changes with:

```
pyplot.draw()
```



edited Jan 12 '13 at 8:33

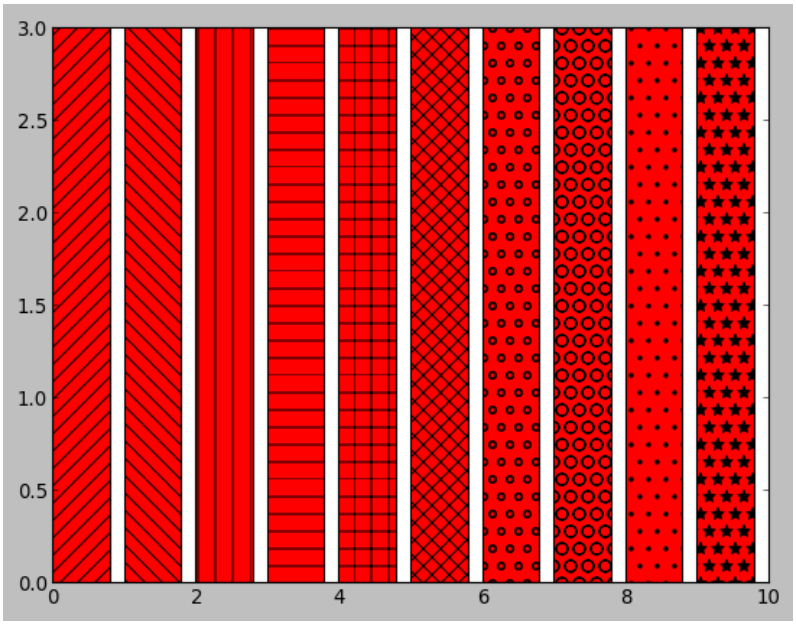
answered Jan 11 '13 at 13:48

EOL
28.2k 15 78 130

1 Awesome! I would have never found that out so fast on my own. Thanks a lot! :) – pemistahl Jan 11 '13 at 14:04

```
import matplotlib.pyplot as plt

fig = plt.figure()

patterns = [ "/" , "\\" , "|" , "-" , "+" , "x", "o", "O", ".", "*" ]
```

```
ax1 = fig.add_subplot(111)
for i in range(len(patterns)):
    ax1.bar(i, 3, color='red', edgecolor='black', hatch=patterns[i])


plt.show()
```



It's in the documentation here:

http://matplotlib.org/api/artist_api.html#matplotlib.patches.Patch.set_hatch

Okay - so to texture a piechart, you need to do this:

if you look here:

```
Return value:
If autopct is None, return the tuple (patches, texts):

patches is a sequence of matplotlib.patches.Wedge instances
texts is a list of the label matplotlib.text.Text instances.
```

so then we look at the Wedges page, and see that it has a set_hatch() method.

so we just need to add a few lines to the piechart demo and...

""" import matplotlib.pyplot as plt

fig = plt.figure()

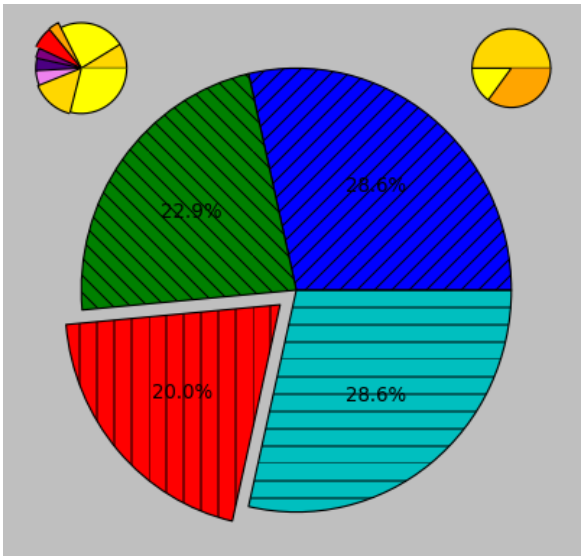
patterns = ["/" , "\" , "|" , "-" , "+" , "x", "o", "O", ".", "*"]

ax1 = fig.add_subplot(111) for i in range(len(patterns)): ax1.bar(i, 3, color='red', edgecolor='black', hatch=patterns[i])

plt.show()"""

```
"""
Make a pie chart - see
http://matplotlib.sf.net/matplotlib.pylab.html#-pie for the docstring.

This example shows a basic pie chart with labels optional features,
like autolabeling the percentage, offsetting a slice with "explode",
adding a shadow, and changing the starting angle.

"""

from pylab import *
import math
import numpy as np

patterns = [ "/" , "\\" , "|" , "-" , "+" , "x", "o", "O", ".", "*" ]


def little_pie(breakdown,location,size):
    breakdown = [0] + list(np.cumsum(breakdown)* 1.0 / sum(breakdown))
```

```
    for i in xrange(len(breakdown)-1):
        x = [0] + np.cos(np.linspace(2 * math.pi * breakdown[i], 2 * math.pi *
                        breakdown[i+1], 20)).tolist()
        y = [0] + np.sin(np.linspace(2 * math.pi * breakdown[i], 2 * math.pi *
                        breakdown[i+1], 20)).tolist()
        xy = zip(x,y)
        scatter( location[0], location[1], marker=(xy,0), s=size, facecolor=
               ['gold','yellow', 'orange', 'red','purple','indigo','violet'][i%7])

figure(1, figsize=(6,6))

little_pie([10,3,7],(1,1),600)
little_pie([10,27,4,8,4,5,6,17,33],(-1,1),800)

fracs = [10, 8, 7, 10]
explode=(0, 0, 0.1, 0)

**piechart =** pie(fracs, explode=explode, autopct='%1.1f%%')
**for i in range(len(piechart[0])):
    piechart[0][i].set_hatch(patterns[(i)%len(patterns)])**


show()
```



edited Jan 11 '13 at 14:11

answered Jan 11 '13 at 13:55

will

3,556 11 25

Thanks. :) And how do I do this for pie wedges? – pemistahl Jan 11 '13 at 13:58

@PeterStahl - Sorry, i've never used pie charts. set_hatch is a method for the artist though, so you should be able to set it get getting the artist for each slice. – will Jan 11 '13 at 14:04

@PeterStahl okay, worked a way to do it. – will Jan 11 '13 at 14:12

1 Thank you will. But since @EOL was the first one providing the solution, I accepted his answer. I gave you an upvote though. Thanks! :) – pemistahl Jan 11 '13 at 14:18

Yah i noticed. Pipped me at the post – will Jan 11 '13 at 14:20