Pandas, Matplotlib and Numpy

We use the modules pandas and matplotlib to import a dataset and create a nice visualization. Pandas assumes that your data file has rows separated by newlines, and columns separated by an expression that you specify. It also assumes that the first row contains the names of your columns. We start with the dataset on LSD and math scores[1], from the page `stat.ufl.edu/winner/datasets.html`.

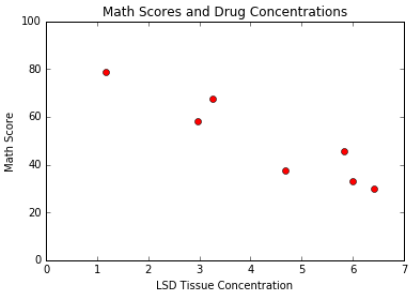
```
drug math
1.17 78.93
2.97 58.20
3.26 67.47
4.69 37.47
5.83 45.65
6.00 32.92
6.41 29.97
```

I added the column names and saved this as a .txt file. We then import this into pandas as follows:

```
import pandas as pd
lsd=pd.read_table('lsd.txt','\t')
```

Pandas lets us call the separate columns by their name using commands `lsd['math']` and `lsd['drug']`. We can use these directly as lists to input into pyplot and create a scatter plot.

```
import matplotlib.pyplot as plt
plt.plot(lsd['drug'],lsd['math'],'o')
plt.axis([0, 7, 0, 100])
plt.ylabel('Math Score')
plt.xlabel('LSD Tissue Concentration')
plt.title('Math Scores and Drug Concentrations')
plt.show()
```

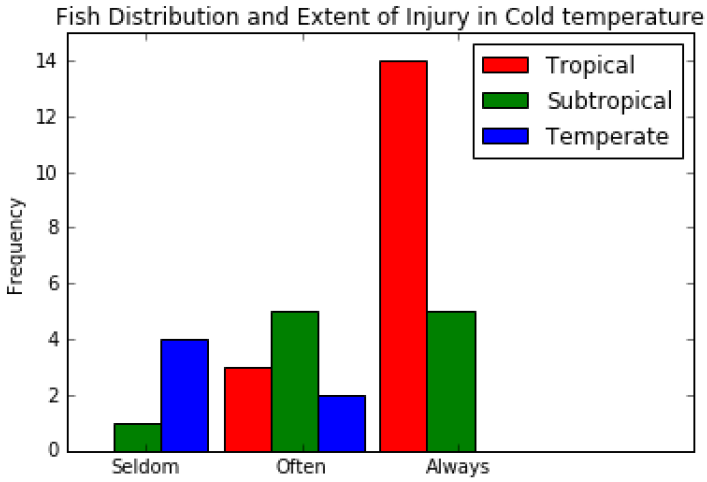


We did the same for a data set on different types of fish and injuries[2]. Notice how we split the dataframe up into categories first, and use the familiar `len()` function to obtain the number of observations in each category.

[1]Dataset: lsd.dat Source: Wagner, Agahajanian, and Bing (1968). Correlation of Performance Test Scores with Tissue Concentration of Lysergic Acid Diethylamide in Human Subjects. Clinical Pharmacology and Therapeutics, Vol.9 pp635-638. Description: Group of volunteers was given LSD, their mean scores on math exam and tissue concentrations of LSD were obtained at n=7 time points. Variables/Columns: Tissue Concentration 1-4 Math Score 8-12

[2]Dataset: coldfish.dat Source: Margaret Storey (1937). "The Relation Between Normal Range and Mortality of Fishes Due to Cold at Sanibel Island, Florida," Ecology, Vol. 18, #1, pp. 10-26 Description: Distribution of fish species and extent of injury due to cold temperatures at Sanibel Island, Florida. Variables/Columns: Species Distribution 8 /* 1=Tropical, 2=Subtropical, 3=Temperate */ Extent of Injury 16 /* 1=Seldom, 2=Often, 3=Always */ 0

```
import numpy as np
coldfish=pd.read_table('coldfish.txt','\s+')
tropical=coldfish['species']==1
subtropical=coldfish['species']==2
temperate=coldfish['species']==3
seldom=coldfish['injury']==1
often=coldfish['injury']==2
always=coldfish['injury']==3
barwidth=.3
plt.axis([0, 4, 0, 15])
plt.ylabel('Frequency')
plt.title('Fish Distribution and Extent of Injury in Cold temperature')
plt.xticks(np.arange(3)+1/2,['Seldom','Often','Always'])
plt.bar(np.arange(3),[len(coldfish[tropical & seldom].index),
  len(coldfish[tropical & often].index),
  len(coldfish[tropical & always].index)],
  barwidth,color='r',label='Tropical')
plt.bar(np.arange(3)+barwidth,[len(coldfish[subtropical & seldom].index),
  len(coldfish[subtropical & often].index),
  len(coldfish[subtropical & always].index)],
  barwidth,color='g',label='Subtropical')
plt.bar(np.arange(3)+2*barwidth,[len(coldfish[temperate & seldom].index),
  len(coldfish[temperate & often].index),
  len(coldfish[temperate & always].index)],
  barwidth,color='b',label='Temperate')
plt.legend()
plt.show()
```



Networks

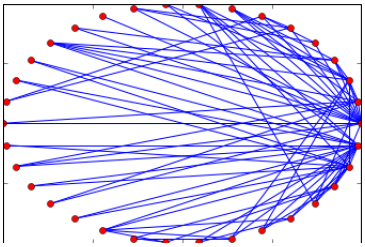
Networks are sets of nodes that may pairwise be connected by links. Links may be directed or weighted, and the network might contain other information such as categories of nodes or links. We

can store a network in a 2-dimensional array (a list of lists) such that the value at index i, j indicates the presence of a link. For example, here is a network with nodes 0, 1, 2, such that node 1 is connected to node 0 and 2:

```
In:  N=[[0,1,0],[1,0,1],[0,1,0]]
In:  N[0][1]
Out:  1
In:  N[0][2]
Out:  0
```

In class, we wrote a function that takes any network in this form, and plots it using matplotlib. It creates the x and y coordinates by placing the nodes equally spaced around a circle:

```
def network_plot_circle(N):
    n=len(N)
    x=[np.cos(2*np.pi*i/n) for i in range(n)]
    y=[np.sin(2*np.pi*i/n) for i in range(n)]
    for i in range(n):
        for j in range(i):
            if N[i][j]==1:
                plt.plot([x[i],x[j]],[y[i],y[j]],'b')
    plt.plot(x,y,'ro')
```



The example network is the Zachary Karate Club social network. You can find the data for this on the materials page on my website.

Exercises

- Create visualizations for several other datasets from the toy dataset page.
- Adapt the network plotting code so that it plots the nodes at uniform randomly chosen coordinates.
- Adapt the network plotting code so that it plots edges of two different colors, which the user can indicate by recording edges as 1s or 2s in their data.
- Adapt the network plotting code so that it takes as input a network and list, which is a subset of the nodes. It plots those nodes in a different color from the rest, and plots them next to each other on the circle.
- Adapt the network plotting code so that it plots edges of different thickness, depending on their value in the data.