NEURALBERTATECH

Presents:

Loops, Pandas & Visualizing

September 24th, 2019
Created by Eden Redman

Install Pandas

```
conda install pandas

pip install pyaudio

pip install wave

conda install scipy
```

Functions Review

```
<var> = function_name(arguement_1,arguement_2,...)
```

```
x = input(“prompt”)
name = input(“enter your name: ”)
```

Output:

```
enter your name:
```

If Review

```
if x < 5:
    print(“x is less than 5”)
else:
    print(“x is greater than or equal to 5”)
```

For Loops

```
y = 0

for x in range(10):

   print(x)

   y += 1

print(y)
```

For Loops (Cont.)

```
list = [1,0,1,1,1,0,1,0,1]

count_0,count_1 = 0,0

for x in range(len(list)):

    if x == 0:

        count_0 += 1

    else:

        count_1 += 1
```

While Loops (Cont.)

```
import random

x = 0

count = 0

while count <= 10

    x += random.randint(0,10)

    count += 1

print(x)

print(count)
```

Your Turn!

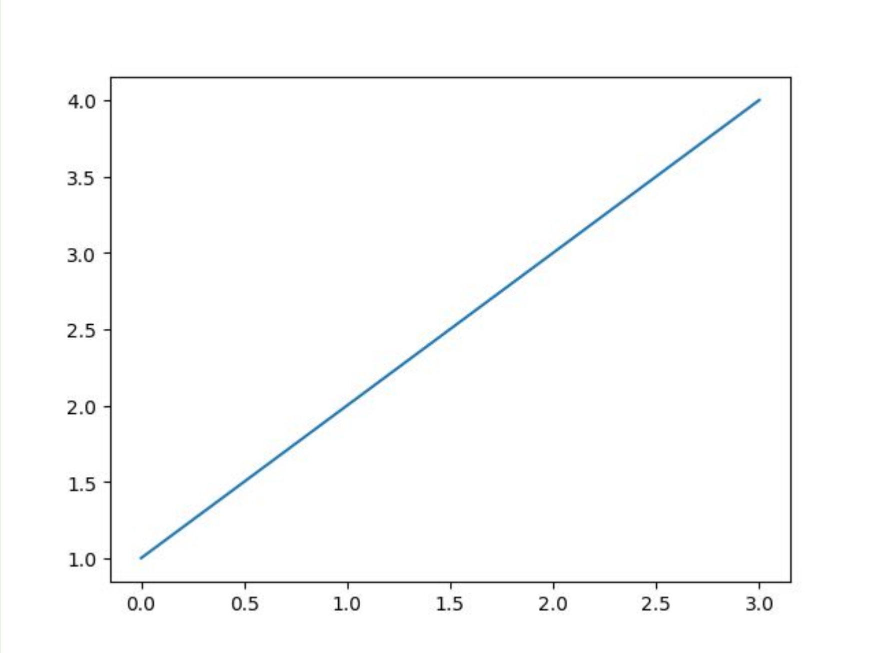
Breaktime, sort of

Say hello to your neighbour and work on the following problem:

How do we construct a loop that moves backwards through our range?

Matplotlib

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4])

plt.show()
```

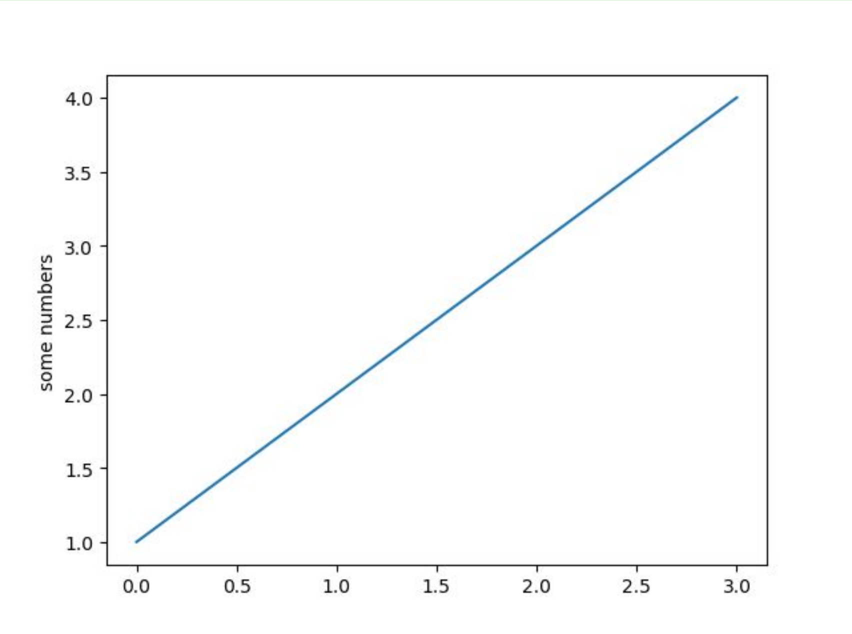


Matplotlib (Cont.)

plt.plot([1, 2, 3, 4])

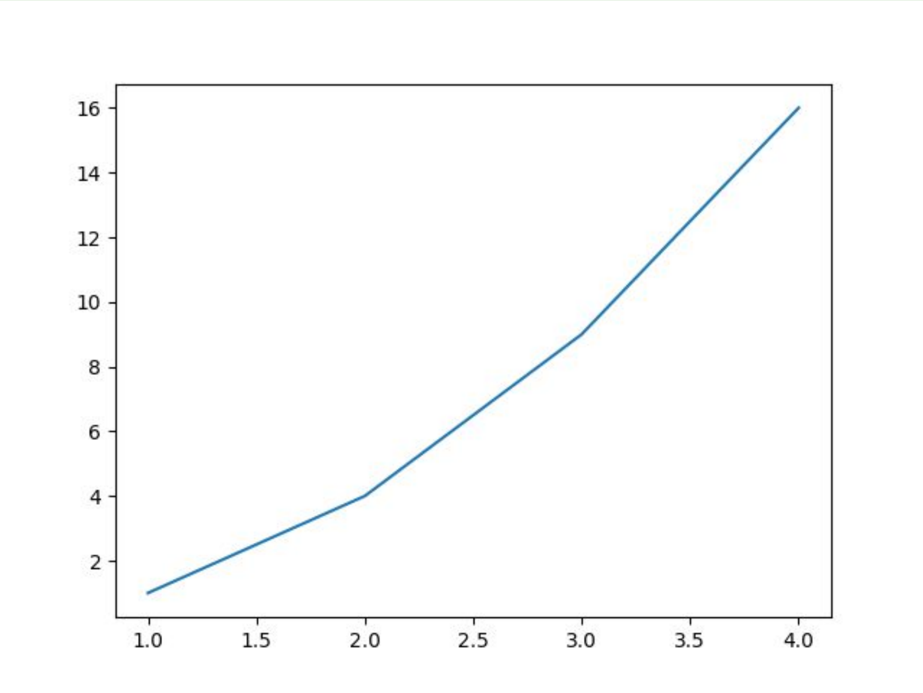
plt.ylabel('some numbers')

plt.show()



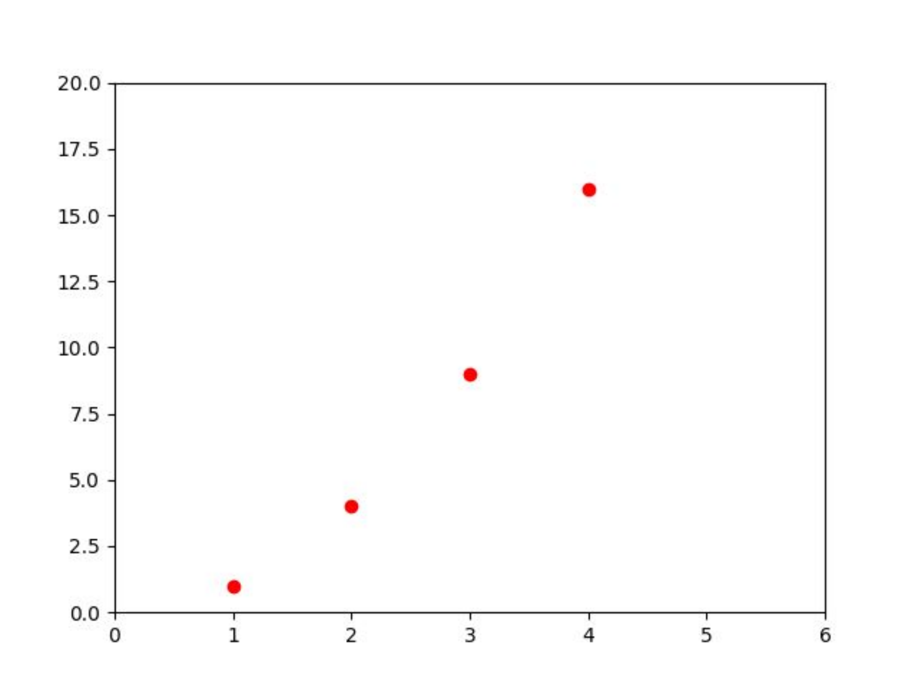
Matplotlib (Cont.)

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])

plt.show()
```



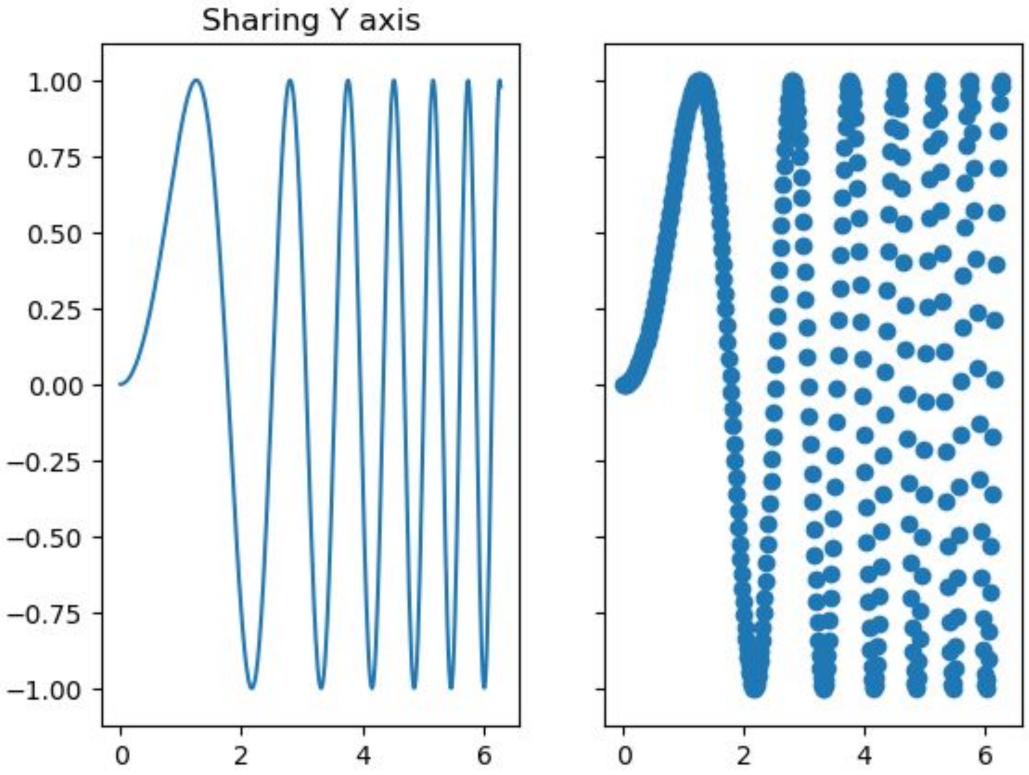
Matplotlib (Cont.)

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16],'ro')

plt.axis([0, 6, 0, 20])

plt.show()
```



Making Multiple Plots!!!

```
x = np.linspace(0, 2*np.pi, 400)

y = np.sin(x**2)



f, (ax1, ax2) = plt.subplots(1, 2, sharey=True)

ax1.plot(x, y)

ax1.set_title('Sharing Y axis')

ax2.scatter(x, y)
```

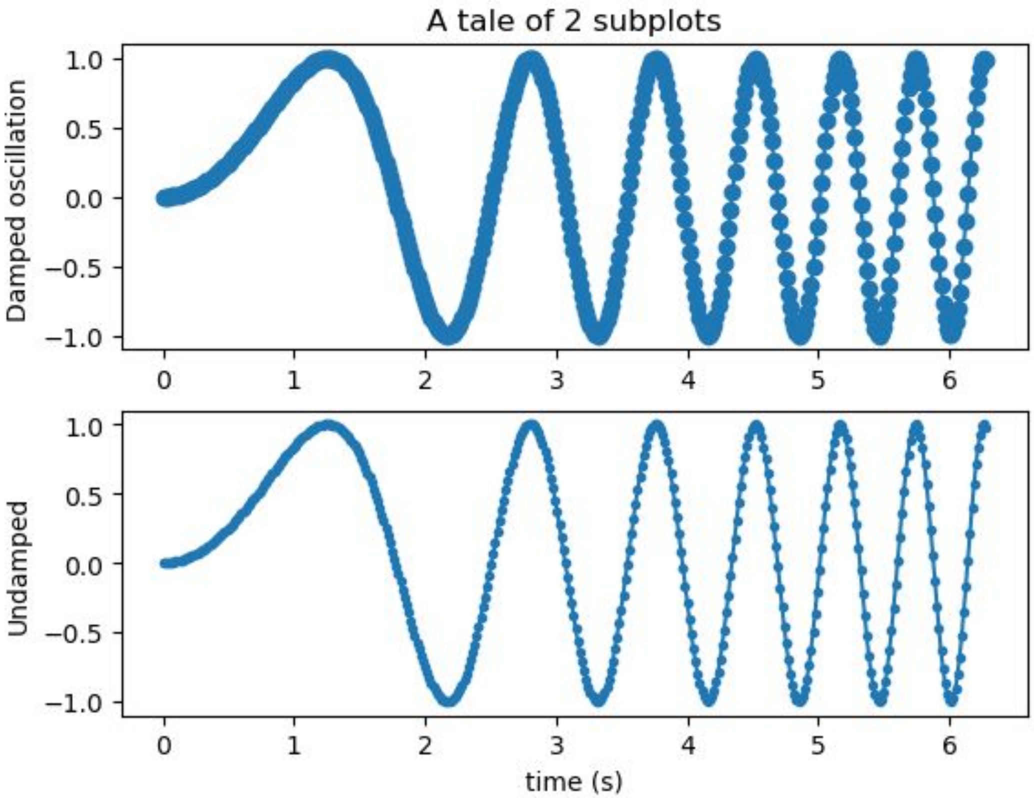

Sharing Y axis
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
0
2
4
6

Making Multiple Plots (Differently)!!!

```
x = np.linspace(0, 2*np.pi, 400)

y = np.sin(x**2)
```

```
plt.subplot(2, 1, 1)

plt.plot(x,y, 'o-')

plt.title('A tale of 2 subplots')

plt.ylabel('Damped oscillation')
```

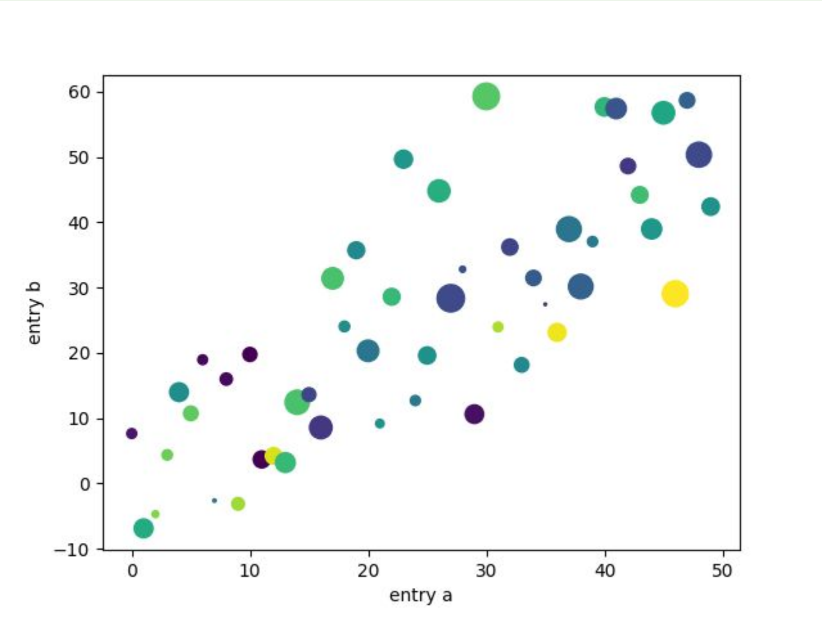
```
plt.subplot(2, 1, 2)

plt.plot(x,y, '.-')

plt.xlabel('time (s)')

plt.ylabel('Undamped')
```



NEURALBERTATECH

Scatter Plot

```
plt.scatter('a', 'b', c='c', s='d', data=data)

plt.xlabel('entry a')

plt.ylabel('entry b')

plt.show()
```

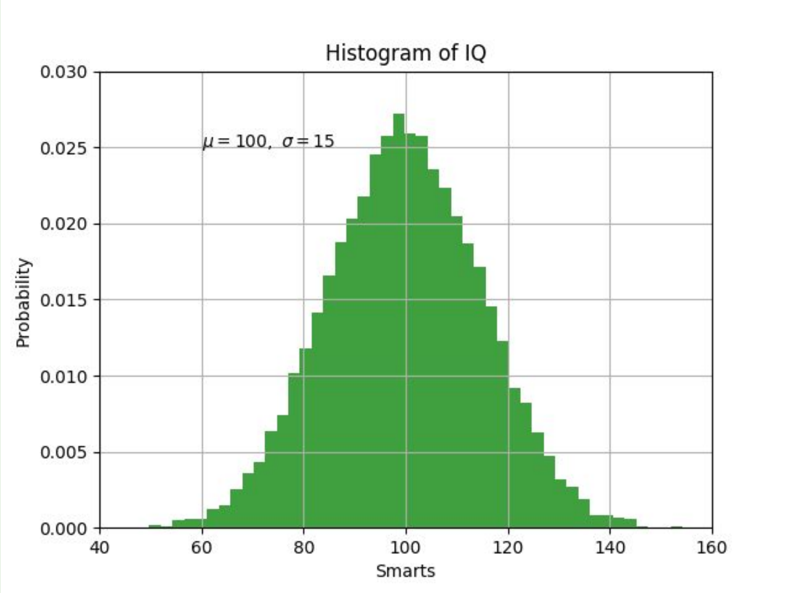


Construct Random-ish Dictionary

```
dict = {‘a’:’value’}



data = {'a': np.arange(50),

        'c': np.random.randint(0, 50, 50),

        'd': np.random.randn(50)}

data['b'] = data['a'] + 10 * np.random.randn(50)

data['d'] = np.abs(data['d']) * 100
```

Histograms - Plot

```
plt.xlabel('Smarts')

plt.ylabel('Probability')

plt.title('Histogram of IQ')

plt.text(60, .025, r'$\mu=100,\ \sigma=15$')

plt.axis([40, 160, 0, 0.03])

plt.grid(True)

plt.show()
```

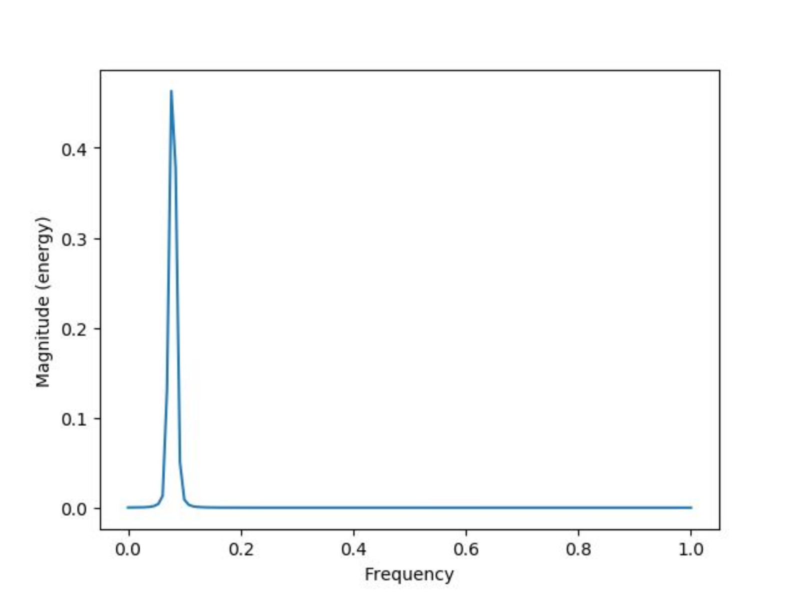


Histograms - Data

```
mu, sigma = 100, 15
x = mu + sigma * np.random.randn(10000)


# the histogram of the data
n, bins, patches = plt.hist(x, 50, density=1, facecolor='g', alpha=0.75)
```

Spectra

```
time   = np.arange(0, 65, .25);

 signalAmplitude   = np.sin(time)



 plot.magnitude_spectrum(signalAmplitude)

 plot.show()
```



Record Sample Audio

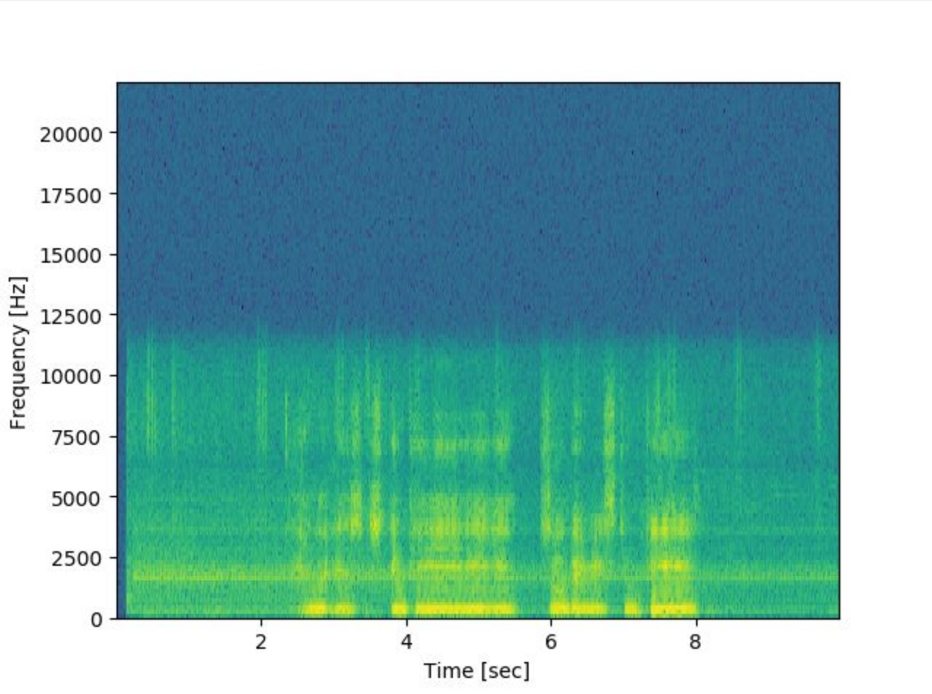
Open Pull_Audio.py

Record a duration of audio in seconds defined by RECORD_SECONDS

Spectragram

$$\mathrm{Spec}_k(\mathbf{x}) \equiv \left|\sum_{s=1}^{t} e^{iks} x_s\right|^2 = \left(\sum_{s=1}^{t} \cos(ks) x_s\right)^2 + \left(\sum_{s=1}^{t} \sin(ks) x_s\right)$$

Open Open_Wav_Spectrogramify.py



Homework

1. Program a game of rock, paper, scissors (you v computer)
   a. Bonus - for each round you update a bar graph that depicts win records
   b. Extra Bonus - subplot that depicts ratio of your wins/ties/losses

2. Read through next 4 slides

   a. Rewrite slide 18-19 using Pandas Dataframe (well documented online)

Pandas Dataframes (Series)

```
import pandas as pd

s = pd.Series([1, 3, 5, np.nan, 6, 8])        # pandas equivalent of a basic array

s
```

Pandas Dataframes (DataFrames)

```
df = pd.DataFrame({'A': 1.,
   ...:                    'B': pd.Timestamp('20130102'),
   ...:                    'C': pd.Series(1, index=list(range(4)), dtype='float32'),
   ...:                    'D': np.array([3] * 4, dtype='int32'),
   ...:                    'E': pd.Categorical(["test", "train", "test", "train"]),
   ...:                    'F': 'foo'})
df.dtypes
```

```
dates = pd.date_range('20130101', periods=6)

df2 = pd.DataFrame(np.random.randn(6, 4), index=dates, columns=list('ABCD'))

df2
```

Some Pandas functions

```
df2.head()

df2.tail(3)

df2.T

df2.to_numpy()

df2.describe()

df2.sort_values(by='B')

df2.sort_index(axis=1, ascending=False)
```

Pandas Extras

```
# Index Indexing
df2['A']

df2[0:3]


# Label Indexing
df2.loc[dates[0]]

df.loc['20130102', ['A', 'B']]
```

```
# Positional Indexing
df2.iloc[3]

df2.iloc[[1, 2, 4], [0, 2]]
```