

```
In [50]: import matplotlib.ticker as mticker

import matplotlib.pyplot as plt
import numpy as np
import urllib
import matplotlib.dates as mdates
import datetime as dt
%matplotlib inline
from matplotlib import style
style.use('ggplot')
```

```

In [53]: #converting the timestamps from the Yahoo finance API to times that Matplotlib
         understands
         def bytespdate2num(fmt, encoding='utf-8'):
             strconverter = mdates.strpdate2num(fmt)

             def bytesconverter(b):
                 s = b.decode(encoding)
                 return strconverter(s)

             return bytesconverter

         def graph_data():

             fig = plt.figure()
             ax1 = plt.subplot2grid((1,1), (0,0))
             # Unfortunately, Yahoo's API is no longer available
             # feel free to adapt the code to another source, or use this drop-in replacement.
             stock_price_url = 'https://pythonprogramming.net/yahoo_finance_replacement'

             source_code = urllib.request.urlopen(stock_price_url).read().decode()
             stock_data = []
             split_source = source_code.split('\n')
             for line in split_source[2:]:
                 split_line = line.split(',')
                 if len(split_line) == 7:
                     if 'values' not in line and 'labels' not in line:
                         stock_data.append(line)

             # pprint.pprint(stock_data)
             date, openp, highp, lowp, closep, adj_closep, volume = np.loadtxt(stock_data,
                                                                                   delimiter=
r=',',
                                                                                   unpack=T
rue,
                                                                                   # %Y = f
ull year. 2015
                                                                                   # %y = p
artial year 15
                                                                                   # %m = n
umber month

```

```

# %d = n
umber day
# %H = h
ours
# %M = m
inutes
# %S = s
econds
# 12-06-
2014
# %m-%d-
%Y
converte

rs={0: bytesdate2num('%Y-%m-%d')}}

x = 0
y = len(date)
ohlc = []

while x < y:
    append_me = date[x], openp[x], highp[x], lowp[x], closep[x], volume[x]
    ohlc.append(append_me)
    x+=1

#candlestick_ohlc(ax1, ohlc, width=0.4, colorup='#77d879', colordown='#db3
f3f')
ax1.plot(date,closep)
ax1.plot(date,openp)

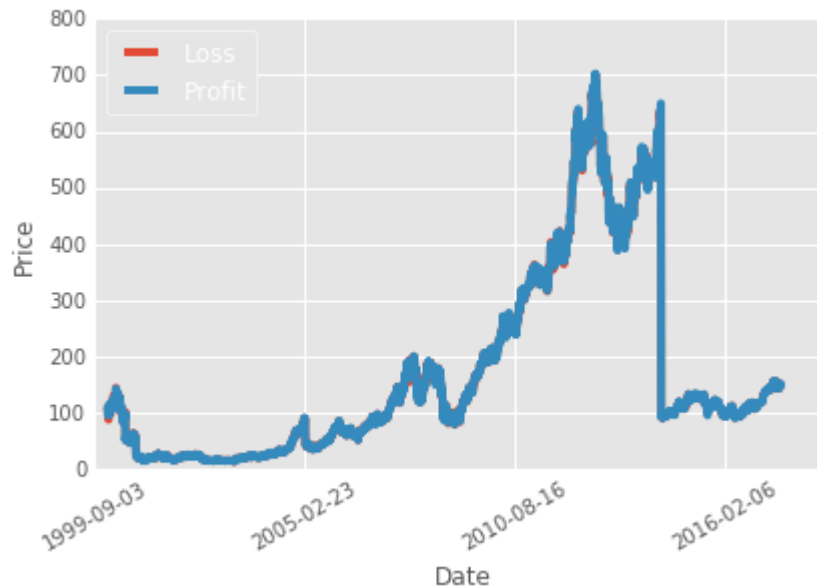
for label in ax1.xaxis.get_ticklabels():
    label.set_rotation(30)

ax1.xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m-%d'))
ax1.xaxis.set_major_locator(mticker.MaxNLocator(4))
ax1.grid(True)

plt.xlabel('Date')
plt.ylabel('Price')
plt.title('Market Stock')
plt.legend(['Loss', 'Profit'], loc='upper left')
plt.subplots_adjust(left=0.09, bottom=0.20, right=0.94, top=0.98,
wspace=0.2, hspace=0)
plt.show()

graph_data()

```



```
In [40]: print(plt.style.available)
```

```
['grayscale', 'classic', 'seaborn-paper', 'seaborn-muted', 'seaborn-pastel',  
 'seaborn-poster', 'dark_background', 'seaborn-dark-palette', 'seaborn-notebo  
ok', 'ggplot', 'seaborn-bright', 'seaborn-darkgrid', 'seaborn-colorblind', 's  
eaborn-whitegrid', 'bmh', 'seaborn-white', 'seaborn-dark', 'seaborn-deep', 's  
eaborn-talk', 'fivethirtyeight', 'seaborn-ticks']
```

```
In [ ]:
```