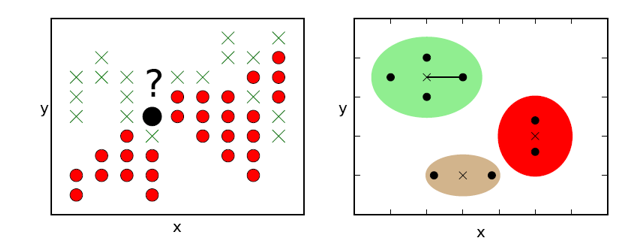
Teaching Assistant: Pritti Agrawal

Pythontutor.com

Lecture 1:

Focus: Prediction and Classification



Tasks:

Get historical stock data (5 years Jan 1, 2013 – Dec 31, 2018 - table 1,250 rows

2 ways to this: use Python Script:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

from pandas\_datareader import data as web

import os

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

def get\_stock(ticker, start\_date, end\_date, s\_window, l\_window):

try:

df = web.get\_data\_yahoo(ticker, start=start\_date, end=end\_date)

df['Return'] = df['Adj Close'].pct\_change()

df['Return'].fillna(0, inplace = True)

df['Date'] = df.index

df['Date'] = pd.to\_datetime(df['Date'])

df['Month'] = df['Date'].dt.month

df['Year'] = df['Date'].dt.year

df['Day'] = df['Date'].dt.day

df['Weekday'] = df['Date'].dt.weekday\_name

df['Short\_MA'] = df['Adj Close'].rolling(window=s\_window, min\_periods=1).mean()

df['Long\_MA'] = df['Adj Close'].rolling(window=l\_window, min\_periods=1).mean()

col\_list = ['Date', 'Year', 'Month', 'Day', 'Weekday',

'High', 'Low', 'Close', 'Volume', 'Adj Close',

'Return', 'Short\_MA', 'Long\_MA']

df = df[col\_list]

return df

except Exception as error:

print(error)

return None

ticker='MSFT'

start\_date='2017-01-01'

end\_date='2017-12-31'

s\_window = 14

l\_window = 50

input\_dir = r'C:\Users\epinsky\bu\python\data\_science\_with\_Python\datasets'

output\_file = os.path.join(input\_dir, ticker + '.csv')

df = get\_stock(ticker, start\_date, end\_date, s\_window, l\_window)

df.to\_csv(output\_file, index=False)

with open(output\_file) as f:

lines = f.read().splitlines()

df[['Short\_MA', 'Long\_MA', 'Adj Close']].plot()

#df\_2 = pd.read\_csv(output\_file)

We set NaN values to 0

(problem of missing data)

Q:

2,2,2,3,5,6,7,89,0,3,45,6,7,89,…., 80, …….

Imputing data (i.e. fill in missing data)

If Adj Close > L\_MA then buy

If Adj\_close < L\_MA then sell (if you have a stock)

Homework Assignment #1:

Step 1: run the script with your stock ticker and save the results into csv file

Step 2: read stock data into list of lists (lines)

Step 2: all computations will be done in Python

What do you need to do:

Implement some trading strategies and get some statistics (use “Adj Close” in computation of your returns)

Note: nyse.com and look up tickers

1. If you decide to hold a stock for 1 day, what is the best day of the week to do so –

For each day of the week, compute average, min and max of daily returns

Result:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Day of the week | min | max | average | median |
| Monday | -5.29% | 7.85 | 4.59 | 2.87 |

1. If you decide to buy a stock for one month, what would be the best/worst month?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Month | min | max | average | median |
| January | -5.29% | 7.85 | 4.59 | 2.87 |

Trading Strategies:

1. If your stock is down W days in a row, buy on day W and sell next day: put $100 at adj closing price

Example:

Day 1 -3%

Day 2 -2%

Day 3 -1% adj Price = $10

You buy 10 shares

Day 4 sell you position

For W = 1,2,3,4,5

After W consecutive declines, buy on day W ($100), sell on W+1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| W | Trades | #proftable  trades | Profit/  Per  profitable trade | #losing  trades | Average  Loss  Per  Losing  trade |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Last strategy:

If adj close > s\_ma then buy

Else sell

Same statistics as before:

(breakdown by losing and winning trades)

Optional Assignments:

Assume you know tomorrow, start with a $100

Q: how much money would you have on 12/31/2018?