Signal_Backtest

November 20, 2017

In [ ]:
```
mainland_bank = ['939']
mainland_insur = ['2318','2628']
mainland_realestate = ['688']
gambling = ['27','1928']
energy = ['883','386','857']
telecommunications = ['728','941']
Network = ['700','992']
local_insur = ['1299']
local_bank = ['5']
local_realestate = ['16']
Finance = ['388']
public_service = ['902']
car = ['2333']
congol = ['1']

strategy_list = ['0939.HK', '2318.HK', '2628.HK', '0688.HK', '0027.HK',
                 '1928.HK', '0883.HK', '0386.HK', '0857.HK', '0728.HK',
                 '0941.HK', '0700.HK', '0992.HK', '1299.HK', '0005.HK',
                 '0016.HK', '0388.HK', '0902.HK', '2333.HK', '0001.HK']

namelist = []

for i in strategy_list:
    namelist.append(i.split('.')[0])
```

In [ ]:
```
import pandas as pd
import bisect
from statistics import median

localpath = '/Users/Lwmformula/Downloads/option_trade/strategy_trade/{}.HK.csv'

def percentChange(startPoint,currentPoint):
    return ((float(currentPoint-startPoint))/abs(startPoint))*100

Backtest = pd.DataFrame(columns=['Num','BigYang','BigYin','Goodeat',
                                 'Badeat','Bull_harami','Bear_harami',
                                 'Bull_reverse','Bear_reverse','sunrise',
```

```
                                        'darkcloud','shootingstar','hammer'])

        for i in range(len(namelist)):
            Backtest.set_value(i, 'Num', namelist[i])
        Backtest = Backtest.set_index(['Num'])
```

In [ ]:
```
        import pandas as pd
        import bisect
        from statistics import median

        localpath = ('/Users/Lwmformula/Downloads/option_trade/' +
                     'strategy_trade/{}.HK.csv')

        signal = ['BigYang','BigYin','Goodeat','Badeat','Bull_harami',
                  'Bear_harami','Bull_reverse','Bear_reverse','sunrise',
                  'darkcloud','shootingstar','hammer']

        Cols = ['Descript','one_day','two_day','three_day',
                'four_day','five_day']

        Rows = ['up3_median','down3_median','up3_times','down3_times',
                'up3_Prob','down3_Prob','exp_up_%','exp_down_%',
                'occurrence','Rich_index','Rich_vol_index']

        def create_backtest_df():
            Backtest_inv = pd.DataFrame(columns = Cols)
            for i in range(len(Rows)):
                Backtest_inv.set_value(i, 'Descript', Rows[i])
            Backtest_inv = Backtest_inv.set_index(['Descript'])
            return Backtest_inv

        def percentChange(startPoint,currentPoint):
            return ((float(currentPoint-startPoint))/abs(startPoint))*100
```

In [ ]:
```
        def create_signal(df):

            openp = df.loc[:,'Open'].tolist()
            closep = df.loc[:,'Close'].tolist()
            lowp = df.loc[:,'Low'].tolist()
            highp = df.loc[:,'High'].tolist()


            # BigYang and BigYin
            Yang = []
            Yin = []
            change = []
            for i in range(len(df)):
                change.append(percentChange(openp[i],closep[i]))
```

```
    if change[i] >= 4.00 and change[i] <= 5.00:
        Yang.append(1)
    else: Yang.append(0)
    if change[i] <= -4.00 and change[i] >= -5.00:
        Yin.append(1)
    else: Yin.append(0)
df['BigYang'] = Yang
df['BigYin'] = Yin

# Goodeat and Badeat
Goodeat = [0,]
Badeat = [0,]
for i in range(1,len(df)):
    if (closep[i-1] < openp[i-1] and
        closep[i] > openp[i] and
        closep[i] > openp[i-1] and
        closep[i-1] > openp[i]):
        Goodeat.append(1)
    else: Goodeat.append(0)
    if (closep[i-1] > openp[i-1] and
        closep[i] < openp[i] and
        closep[i-1] < openp[i] and
        closep[i] < openp[i-1]):
        Badeat.append(1)
    else: Badeat.append(0)
df['Goodeat'] = Goodeat
df['Badeat'] = Badeat

# Bull_harami and Bear_harami
Bull_harami = [0,]
Bear_harami = [0,]
for i in range(1,len(df)):
    if (closep[i-1] < openp[i-1] and
        closep[i] > openp[i] and
        closep[i-1] < openp[i] and
        closep[i] < openp[i-1]):
        Bull_harami.append(1)
    else: Bull_harami.append(0)
    if (closep[i-1] > openp[i-1] and
        closep[i] < openp[i] and
        closep[i-1] > openp[i] and
        closep[i] > openp[i-1]):
        Bear_harami.append(1)
    else: Bear_harami.append(0)
df['Bull_harami'] = Bull_harami
df['Bear_harami'] = Bear_harami

# Bull_reverse and Bear_reverse
```

```
Bull_reverse = [0,0,]
Bear_reverse = [0,0,]
for i in range(2,len(df)):
    if (((closep[i]-closep[i-1])/closep[i-1]) > 0.03
        and ((closep[i-1]-closep[i-2])/closep[i-2]) < -0.03):
        Bull_reverse.append(1)
    else: Bull_reverse.append(0)
    if (((closep[i]-closep[i-1])/closep[i-1]) < -0.03
        and ((closep[i-1]-closep[i-2])/closep[i-2]) > 0.03):
        Bear_reverse.append(1)
    else: Bear_reverse.append(0)
df['Bull_reverse'] = Bull_reverse
df['Bear_reverse'] = Bear_reverse

# sunrise and darkcloud
sunrise = [0,]
darkcloud = [0,]
for i in range(1,len(df)):
    if (openp[i-1] > closep[i-1] and
        openp[i] < closep[i] and
        openp[i] < closep[i-1] and
        openp[i-1] > closep[i] and
        ((closep[i-1]+openp[i-1])/2) < closep[i]):
        sunrise.append(1)
    else: sunrise.append(0)
    if (openp[i-1] < closep[i-1] and
        openp[i] > closep[i] and
        openp[i-1] < closep[i] and
        openp[i] > closep[i-1] and
        ((closep[i-1]+openp[i-1])/2) > closep[i]):
        darkcloud.append(1)
    else: darkcloud.append(0)
df['sunrise'] = sunrise
df['darkcloud'] = darkcloud

# hammer and shootingstar
hammer = []
shootingstar = []
for i in range(len(df)):
    if (highp[i] - closep[i] <= closep[i] - openp[i] and
        closep[i] > openp[i] and
        (openp[i] - lowp[i]) >= 2*(closep[i]-openp[i]) and
        (closep[i] - openp[i])/openp[i] > 0.005):
        hammer.append(1)
    else: hammer.append(0)
    if (openp[i] > closep[i] and
        (highp[i] - openp[i]) >= (2*(openp[i]-closep[i])) and
        (closep[i]-lowp[i]) <= (openp[i]-closep[i]) and
```

```
                ((openp[i]-closep[i])/openp[i]) > 0.005):
                shootingstar.append(1)
            else: shootingstar.append(0)
        df['hammer'] = hammer
        df['shootingstar'] = shootingstar

        # data until five days before

        for i in range(len(df)-5,len(df)):
            df.set_value(i, 'BigYang', 0)
            df.set_value(i, 'BigYin', 0)

            df.set_value(i, 'Goodeat', 0)
            df.set_value(i, 'Badeat', 0)

            df.set_value(i, 'Bull_harami', 0)
            df.set_value(i, 'Bear_harami', 0)

            df.set_value(i, 'Bull_reverse', 0)
            df.set_value(i, 'Bear_reverse', 0)

            df.set_value(i, 'sunrise', 0)
            df.set_value(i, 'darkcloud', 0)

            df.set_value(i, 'hammer', 0)
            df.set_value(i, 'shootingstar', 0)
```

In [ ]:
```
def create_dict():
    BigYang = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'two_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'three_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'four_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'five_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
```

```
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
             'occurance': len(df) - df['BigYang'].value_counts()[0],
             'Rich_index':'',
             'Rich_vol_index':'',
             'avg_times': round((len(df) -
                                 df['BigYang'].value_counts()[0])/10.7,1),
             'name':'BigYang'}

BigYin = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
          'two_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
          'three_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
          'four_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
          'five_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
          'occurance': len(df) - df['BigYin'].value_counts()[0],
          'Rich_index':'',
          'Rich_vol_index':'',
          'avg_times': round((len(df) -
                              df['BigYin'].value_counts()[0])/10.7,1),
          'name':'BigYin'}

Goodeat = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
           'two_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
           'three_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
```

```
            'four_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
            'five_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
            'occurance': len(df) - df['Goodeat'].value_counts()[0],
            'Rich_index':'',
            'Rich_vol_index':'',
            'avg_times': round((len(df) -
                                df['Goodeat'].value_counts()[0])/10.7,1),
            'name':'Goodeat'}

Badeat = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
          'two_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
          'three_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
          'four_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
          'five_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
          'occurance': len(df) - df['Badeat'].value_counts()[0],
          'Rich_index':'',
          'Rich_vol_index':'',
          'avg_times': round((len(df) -
                              df['Badeat'].value_counts()[0])/10.7,1),
          'name':'Badeat'}

Bull_harami = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'two_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
```

```
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
                'three_day': {'up3':[],'down3':[],'up3_median':'',
                              'down3_median':'','up3_times':'',
                              'down3_times':'','up3_ratio':'','down3_ratio':'',
                              'up_final':'','down_final':''},
                'four_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
                'five_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
                'occurance': len(df) - df['Bull_harami'].value_counts()[0],
                'Rich_index':'',
                'Rich_vol_index':'',
                'avg_times': round((len(df) -
                                    df['Bull_harami'].value_counts()[0])/10.7,1),
                'name':'Bull_harami'}

Bear_harami = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'two_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'three_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
               'four_day': {'up3':[],'down3':[],'up3_median':'',
                            'down3_median':'','up3_times':'',
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
               'five_day': {'up3':[],'down3':[],'up3_median':'',
                            'down3_median':'','up3_times':'',
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
               'occurance': len(df) - df['Bear_harami'].value_counts()[0],
               'Rich_index':'',
               'Rich_vol_index':'',
               'avg_times': round((len(df) -
                                   df['Bear_harami'].value_counts()[0])/10.7,1),
               'name':'Bear_harami'}
```

```
Bull_reverse = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                            'down3_median':'','up3_times':'',
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
               'two_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
               'three_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
               'four_day': {'up3':[],'down3':[],'up3_median':'',
                            'down3_median':'','up3_times':'',
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
               'five_day': {'up3':[],'down3':[],'up3_median':'',
                            'down3_median':'','up3_times':'',
                            'down3_times':'','up3_ratio':'','down3_ratio':'',
                            'up_final':'','down_final':''},
               'occurance': len(df) - df['Bull_reverse'].value_counts()[0],
               'Rich_index':'',
               'Rich_vol_index':'',
               'avg_times': round((len(df) -
                                   df['Bull_reverse'].value_counts()[0])/10.7,1),
               'name':'Bull_reverse'}

Bear_reverse  = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
                 'two_day': {'up3':[],'down3':[],'up3_median':'',
                             'down3_median':'','up3_times':'',
                             'down3_times':'','up3_ratio':'','down3_ratio':'',
                             'up_final':'','down_final':''},
                 'three_day': {'up3':[],'down3':[],'up3_median':'',
                               'down3_median':'','up3_times':'',
                               'down3_times':'','up3_ratio':'','down3_ratio':'',
                               'up_final':'','down_final':''},
                 'four_day': {'up3':[],'down3':[],'up3_median':'',
                              'down3_median':'','up3_times':'',
                              'down3_times':'','up3_ratio':'','down3_ratio':'',
                              'up_final':'','down_final':''},
                 'five_day': {'up3':[],'down3':[],'up3_median':'',
                              'down3_median':'','up3_times':'',
                              'down3_times':'','up3_ratio':'','down3_ratio':'',
                              'up_final':'','down_final':''},
                 'occurance': len(df) - df['Bear_reverse'].value_counts()[0],
```

```
                    'Rich_index':'',
                    'Rich_vol_index':'',
                    'avg_times': round((len(df) -
                                        df['Bear_reverse'].value_counts()[0])/10.7,1)
                    'name':'Bear_reverse'}

sunrise = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                       'down3_median':'','up3_times':'',
                       'down3_times':'','up3_ratio':'','down3_ratio':'',
                       'up_final':'','down_final':''},
           'two_day': {'up3':[],'down3':[],'up3_median':'',
                       'down3_median':'','up3_times':'',
                       'down3_times':'','up3_ratio':'','down3_ratio':'',
                       'up_final':'','down_final':''},
           'three_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
           'four_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
           'five_day': {'up3':[],'down3':[],'up3_median':'',
                        'down3_median':'','up3_times':'',
                        'down3_times':'','up3_ratio':'','down3_ratio':'',
                        'up_final':'','down_final':''},
           'occurance': len(df) - df['sunrise'].value_counts()[0],
           'Rich_index':'',
           'Rich_vol_index':'',
           'avg_times': round((len(df) -
                               df['sunrise'].value_counts()[0])/10.7,1),
           'name':'sunrise'}

darkcloud = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
             'two_day': {'up3':[],'down3':[],'up3_median':'',
                         'down3_median':'','up3_times':'',
                         'down3_times':'','up3_ratio':'','down3_ratio':'',
                         'up_final':'','down_final':''},
             'three_day': {'up3':[],'down3':[],'up3_median':'',
                           'down3_median':'','up3_times':'',
                           'down3_times':'','up3_ratio':'','down3_ratio':'',
                           'up_final':'','down_final':''},
             'four_day': {'up3':[],'down3':[],'up3_median':'',
                          'down3_median':'','up3_times':'',
                          'down3_times':'','up3_ratio':'','down3_ratio':'',
```

```
                                'up_final':'','down_final':''},
                    'five_day': {'up3':[],'down3':[],'up3_median':'',
                                'down3_median':'','up3_times':'',
                                'down3_times':'','up3_ratio':'','down3_ratio':'',
                                'up_final':'','down_final':''},
                    'occurance': len(df) - df['darkcloud'].value_counts()[0],
                    'Rich_index':'',
                    'Rich_vol_index':'',
                    'avg_times': round((len(df) -
                                        df['darkcloud'].value_counts()[0])/10.7,1),
                    'name':'darkcloud'}

    shootingstar  = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                                 'down3_median':'','up3_times':'',
                                 'down3_times':'','up3_ratio':'','down3_ratio':'',
                                 'up_final':'','down_final':''},
                        'two_day': {'up3':[],'down3':[],'up3_median':'',
                                 'down3_median':'','up3_times':'',
                                 'down3_times':'','up3_ratio':'','down3_ratio':'',
                                 'up_final':'','down_final':''},
                        'three_day': {'up3':[],'down3':[],'up3_median':'',
                                  'down3_median':'','up3_times':'',
                                  'down3_times':'','up3_ratio':'','down3_ratio':'',
                                  'up_final':'','down_final':''},
                        'four_day': {'up3':[],'down3':[],'up3_median':'',
                                  'down3_median':'','up3_times':'',
                                  'down3_times':'','up3_ratio':'','down3_ratio':'',
                                  'up_final':'','down_final':''},
                        'five_day': {'up3':[],'down3':[],'up3_median':'',
                                  'down3_median':'','up3_times':'',
                                  'down3_times':'','up3_ratio':'','down3_ratio':'',
                                  'up_final':'','down_final':''},
                        'occurance': len(df) - df['shootingstar'].value_counts()[0],
                        'Rich_index':'',
                        'Rich_vol_index':'',
                        'avg_times': round((len(df) -
                                            df['shootingstar'].value_counts()[0])/10.7,1)
                        'name':'shootingstar'}

    hammer = {'one_day': {'up3':[],'down3':[],'up3_median':'',
                          'down3_median':'','up3_times':'',
                          'down3_times':'','up3_ratio':'','down3_ratio':'',
                          'up_final':'','down_final':''},
                'two_day': {'up3':[],'down3':[],'up3_median':'',
                          'down3_median':'','up3_times':'',
                          'down3_times':'','up3_ratio':'','down3_ratio':'',
                          'up_final':'','down_final':''},
                'three_day': {'up3':[],'down3':[],'up3_median':'',
```

```
                                    'down3_median':'','up3_times':'',
                                    'down3_times':'','up3_ratio':'','down3_ratio':'',
                                    'up_final':'','down_final':''},
                        'four_day': {'up3':[],'down3':[],'up3_median':'',
                                    'down3_median':'','up3_times':'',
                                    'down3_times':'','up3_ratio':'','down3_ratio':'',
                                    'up_final':'','down_final':''},
                        'five_day': {'up3':[],'down3':[],'up3_median':'',
                                    'down3_median':'','up3_times':'',
                                    'down3_times':'','up3_ratio':'','down3_ratio':'',
                                    'up_final':'','down_final':''},
                        'occurance': len(df) - df['hammer'].value_counts()[0],
                        'Rich_index':'',
                        'Rich_vol_index':'',
                        'avg_times': round((len(df) -
                                            df['hammer'].value_counts()[0])/10.7,1),
                        'name':'hammer'}

            whole_dict = {'BigYang':BigYang,'BigYin':BigYin,
                          'Goodeat':Goodeat,'Badeat':Badeat,
                          'Bull_harami':Bull_harami,'Bear_harami':Bear_harami,
                          'Bull_reverse':Bull_reverse,
                          'Bear_reverse':Bear_reverse,'sunrise':sunrise,
                          'darkcloud':darkcloud,
                          'shootingstar':shootingstar,'hammer':hammer,}

            return whole_dict
```

In [ ]:
```
def Rich_index(signal_dict):
    for i in range(len(df)):
        if df.loc[i,signal_dict['name']] == 1:
            #up3
            if percentChange(df.loc[i,'Close'],df.loc[i+1,'High']) > 3:
                bisect.insort(signal_dict['one_day']['up3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+1,'High']))
            if percentChange(df.loc[i,'Close'],df.loc[i+2,'High']) > 3:
                bisect.insort(signal_dict['two_day']['up3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+2,'High']))
            if percentChange(df.loc[i,'Close'],df.loc[i+3,'High']) > 3:
                bisect.insort(signal_dict['three_day']['up3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+3,'High']))
            if percentChange(df.loc[i,'Close'],df.loc[i+4,'High']) > 3:
                bisect.insort(signal_dict['four_day']['up3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+4,'High']))
            if percentChange(df.loc[i,'Close'],df.loc[i+5,'High']) > 3:
                bisect.insort(signal_dict['five_day']['up3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+5,'High']))
```

```
            #down3
            if percentChange(df.loc[i,'Close'],df.loc[i+1,'Low']) < -3:
                bisect.insort(signal_dict['one_day']['down3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+1,'Low']))
            if percentChange(df.loc[i,'Close'],df.loc[i+2,'Low']) < -3:
                bisect.insort(signal_dict['two_day']['down3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+2,'Low']))
            if percentChange(df.loc[i,'Close'],df.loc[i+3,'Low']) < -3:
                bisect.insort(signal_dict['three_day']['down3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+3,'Low']))
            if percentChange(df.loc[i,'Close'],df.loc[i+4,'Low']) < -3:
                bisect.insort(signal_dict['four_day']['down3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+4,'Low']))
            if percentChange(df.loc[i,'Close'],df.loc[i+5,'Low']) < -3:
                bisect.insort(signal_dict['five_day']['down3'],
                              percentChange(df.loc[i,'Close'],df.loc[i+5,'Low']))

    #up3,down3 times and median
    for i in signal_dict:
        if (i == 'occurance' or
            i == 'Rich_index' or
            i == 'name' or
            i == 'avg_times' or
            i == 'Rich_vol_index'):
            continue
        if len(signal_dict[i]['up3']) == 0:
            signal_dict[i].update({'up3_times': 0.0})
            signal_dict[i].update({'up3_median': 0.0})
        else:
            signal_dict[i].update({'up3_times':len(signal_dict[i]['up3'])})
            signal_dict[i].update({'up3_median':median(signal_dict[i]['up3'])})

        if len(signal_dict[i]['down3']) == 0:
            signal_dict[i].update({'down3_times': 0.0})
            signal_dict[i].update({'down3_median': 0.0})
        else:
            signal_dict[i].update({'down3_times':len(signal_dict[i]['down3'])})
            signal_dict[i].update({'down3_median':median(signal_dict[i]['down3'])})

    #up3 and down3 ratio
        signal_dict[i].update({'up3_ratio':(float(signal_dict[i]['up3_times'])/
                                            float(signal_dict['occurance']))*100})
        signal_dict[i].update({'down3_ratio':(float(signal_dict[i]['down3_times'])/
                                              float(signal_dict['occurance']))*100})
    #up and down final
        signal_dict[i].update({'up_final':((float(signal_dict[i]['up3_median'])*
                                            float(signal_dict[i]['up3_ratio']))/
                                            100)})
```

```
            signal_dict[i].update({'down_final':((float(signal_dict[i]['down3_median'])*
                                                  float(signal_dict[i]['down3_ratio']))/
                                                  100)})


        up = (signal_dict['one_day']['up_final'] +
              signal_dict['two_day']['up_final'] +
              signal_dict['three_day']['up_final'] +
              signal_dict['four_day']['up_final'] +
              signal_dict['five_day']['up_final'])

        down = (abs(signal_dict['one_day']['down_final'] +
                    signal_dict['two_day']['down_final'] +
                    signal_dict['three_day']['down_final'] +
                    signal_dict['four_day']['down_final'] +
                    signal_dict['five_day']['down_final']))

        signal_dict.update({'Rich_index':round(((up-down)/100 * 10000)/5,1)})
        signal_dict.update({'Rich_vol_index':round(((up+down)/100 * 10000)/5,1)})
```

In [ ]:
```
for main in namelist:
    df = pd.read_csv(localpath.format(main))
    create_signal(df)
    df_backtest = create_backtest_df()
    whole_dict = create_dict()

    for i in whole_dict:
        Rich_index(whole_dict[i])


    for items in whole_dict:
        for i in Cols[1:]:
            df_backtest.set_value('up3_median',i,
                                  whole_dict[items][i]['up3_median'])
            df_backtest.set_value('down3_median',i,
                                  whole_dict[items][i]['down3_median'])
            df_backtest.set_value('up3_times',i,
                                  whole_dict[items][i]['up3_times'])
            df_backtest.set_value('down3_times',i,
                                  whole_dict[items][i]['down3_times'])
            df_backtest.set_value('up3_Prob',i,
                                  whole_dict[items][i]['up3_ratio'])
            df_backtest.set_value('down3_Prob',i,
                                  whole_dict[items][i]['down3_ratio'])
            df_backtest.set_value('exp_up_%',i,
                                  whole_dict[items][i]['up_final'])
            df_backtest.set_value('exp_down_%',i,
                                  whole_dict[items][i]['down_final'])
```

```
                df_backtest.set_value('occurrence','one_day',
                                      whole_dict[items]['occurance'])
                df_backtest.set_value('Rich_index','one_day',
                                      whole_dict[items]['Rich_index'])
                df_backtest.set_value('Rich_vol_index','one_day',
                                      whole_dict[items]['Rich_vol_index'])

                df_backtest.to_csv('/Users/Lwmformula/Downloads/option_trade/strategy_trade' +
                                   '/Backtest_21082017/{}_{}.csv'.format(main,items),index=True
```

In [ ]:
```
for main in namelist:
    try:
        df = pd.read_csv(localpath.format(main))
        create_signal(df)

        whole_dict = create_dict()
        for i in whole_dict:
            Rich_index(whole_dict[i])

        for i in Backtest:
            rich_index = (str(whole_dict[i]['Rich_vol_index']) +
                          ' (' + str(whole_dict[i]['avg_times']) + ')')
            Backtest.set_value(main, i, rich_index)
    except:
        print main

print Backtest
```

In [ ]:
```
import datetime
ctime = datetime.datetime.now().strftime("%d%m%Y")
Backtest.to_csv('/Users/Lwmformula/Downloads/option_trade/strategy_trade/' +
                'Backtest_21082017/Backtest_Strangle_{}.csv'.format(ctime),
                index=True)
```