A Side View of New Highs and New Lows

By Tom Ku, Faras Holdings LLC

Disclaimer

Faras Holdings LLC ("Company") is not an investment advisory service, nor a registered investment advisor or broker-dealer and does not purport to tell or suggest which securities you should buy or sell for themselves. The analysts and employees or affiliates of Company may hold positions in the stocks or industries discussed here. You understand and acknowledge that there is a very high degree of risk involved in trading securities. The Company, the authors, the publisher, and all affiliates of Company assume no responsibility or liability for your trading and investment results. Factual statements on the Company's website, or in its publications, are made as of the date stated and are subject to change without notice.

Past results of any individual trader or trading system published by Company are not indicative of future returns by that trader or system, and are not indicative of future returns which be realized by you. In addition, the indicators, strategies, columns, articles and all other features of Company's products (collectively, the "Information") are provided for informational and educational purposes only and should not be construed as investment advice. Accordingly, you should not rely solely on the Information in making any investment. Rather, you should use the Information only as a starting point for doing additional independent research in order to allow you to form your own opinion regarding investments. You should always check with your licensed financial advisor and tax advisor to determine the suitability of any investment.

HYPOTHETICAL OR SIMULATED PERFORMANCE RESULTS HAVE CERTAIN INHERENT LIMITATIONS. UNLIKE AN ACTUAL PERFORMANCE RECORD, SIMULATED RESULTS DO NOT REPRESENT ACTUAL TRADING AND MAY NOT BE IMPACTED BY BROKERAGE AND OTHERSLIPPAGE FEES. ALSO, SINCE THE TRADES HAVE NOT ACTUALLY BEEN EXECUTED, THE RESULTS MAY HAVE UNDER- OR OVER-COMPENSATED FOR THE IMPACT, IF ANY, OF CERTAIN MARKET FACTORS, SUCH AS LACK OF LIQUIDITY. SIMULATED TRADING PROGRAMS IN GENERAL ARE ALSO SUBJECT TO THE FACT THAT THEY ARE DESIGNED WITH THE BENEFIT OF HINDSIGHT. NO REPRESENTATION IS BEING MADE THAT ANY ACCOUNT WILL OR IS LIKELY TO ACHIEVE PROFITS OR LOSSES SIMILAR TO THOSE SHOWN.

Agenda

- What's My "A Trade"
- Event Studies on New Highs and New Lows

What's My "A" Trade?

Goal: Trade with a positive **"Statistical Edge".**

Example: A Bet on Coin Flips

"Edge": the Odds of Winning + the Payoff Ratio

- A money-losing strategy may have more wins than losses.
- A bet with a great payoff may be a loser too.

"Statistical Edge": One Flip vs. Many Flips

- The outcome of a single coin flip is highly unpredictable.
- The outcome of many flips is more predictable.

A Trading Strategy is a Recipe for Making a Series of Bets in the Market.

An "A Trade" is a Trading Strategy with a Superior Edge.

- Quantify the edge of a strategy.
- Compare the edges of different strategies.
- Many different quantitative approaches.
- Historical event study is one method.

What's an "Event Study"?

An Example: "Spike Bounce Standby Mode"

- Make a hypothesis: Buying on "Standby" signal is less profitable than buying on "Bounce" signal.
- Define two types of events: "Standby" and "Bounce".
- Compute the historical returns of buying S&P 500 Index on each type of events.
- Compare the returns.

Event Studies

- Spike Bounce Standby Mode
- Spike Bounce Retrigger

Description of the Data Set

- Historical members of S&P 1500 Composite Index
- Daily %NH and %NL based the **close** prices
- S&P 500 Index total return

Set up for Studies

In [2]:

```
from __future__ import division

import numpy as np
import pandas as pd
import study_util as study
import us_eq_data
from datetime import datetime as dt
import matplotlib.pyplot as plt
from collections import OrderedDict

pd.set_option('precision', 3)
```

In [3]:

```
%matplotlib inline
fig_size = (12, 10)
```

In [4]:

```
d_in, spx, spy, vxx = us_eq_data.load_nhnl_data()
```

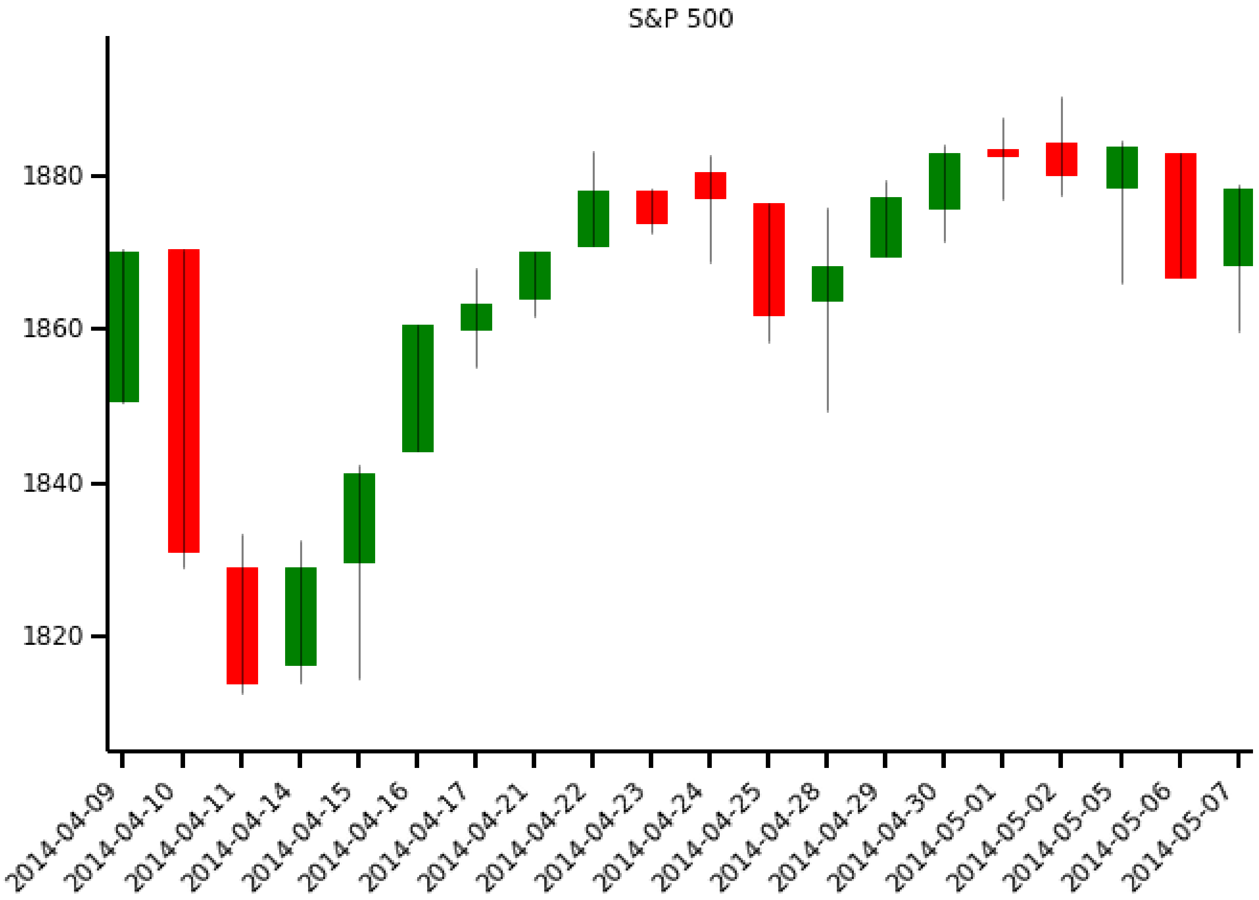
```
In [5]: if 'ts_subject' in globals():
            d_all = study.combine_ret(ts_subject, d_in, hh = np.arange(1, 22))
        else:
            d_all = study.combine_ret(spx.close, d_in, hh = np.arange(1, 22))

        assert(d_all.index.equals(spx.index))
        assert(d_all.index.equals(spy.index))
        assert(d_all.index.equals(vxx.index))
```

```
In [43]: print('The Data set ranges from {} to {}'.format(d_all.index[0].strftime('
         %b %d, %Y'),
                                                     d_all.index[-1].strftime('%b %d,
         %Y')))
```

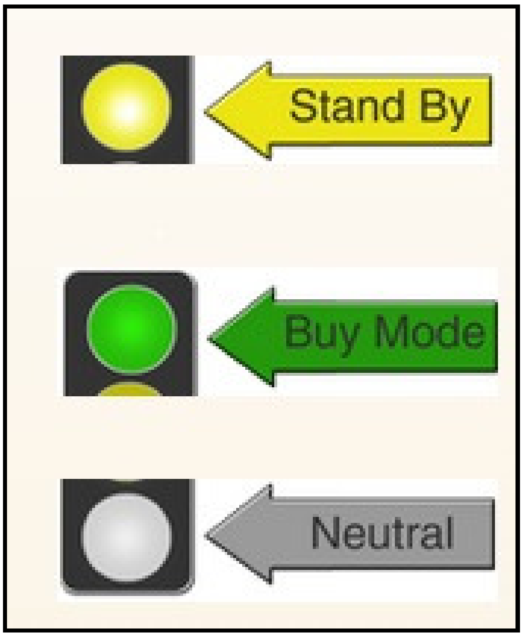
```
         The Data set ranges from Jan 02, 1991 to May 07, 2014
```

```
In [8]: fig = study.chart(spx.tail(20), title = 'S&P 500')
```



Study 1: Spike Bounce Standby Mode

Is Driving on a Yellow Light Dangerous?



Event Definitions

Standby

- %NH - %NL > -20 yesterday
- %NH - %NL <= -20 today

Bounce

- %NH - %NL <= -20 yesterday
- %NH - %NL > -20 today

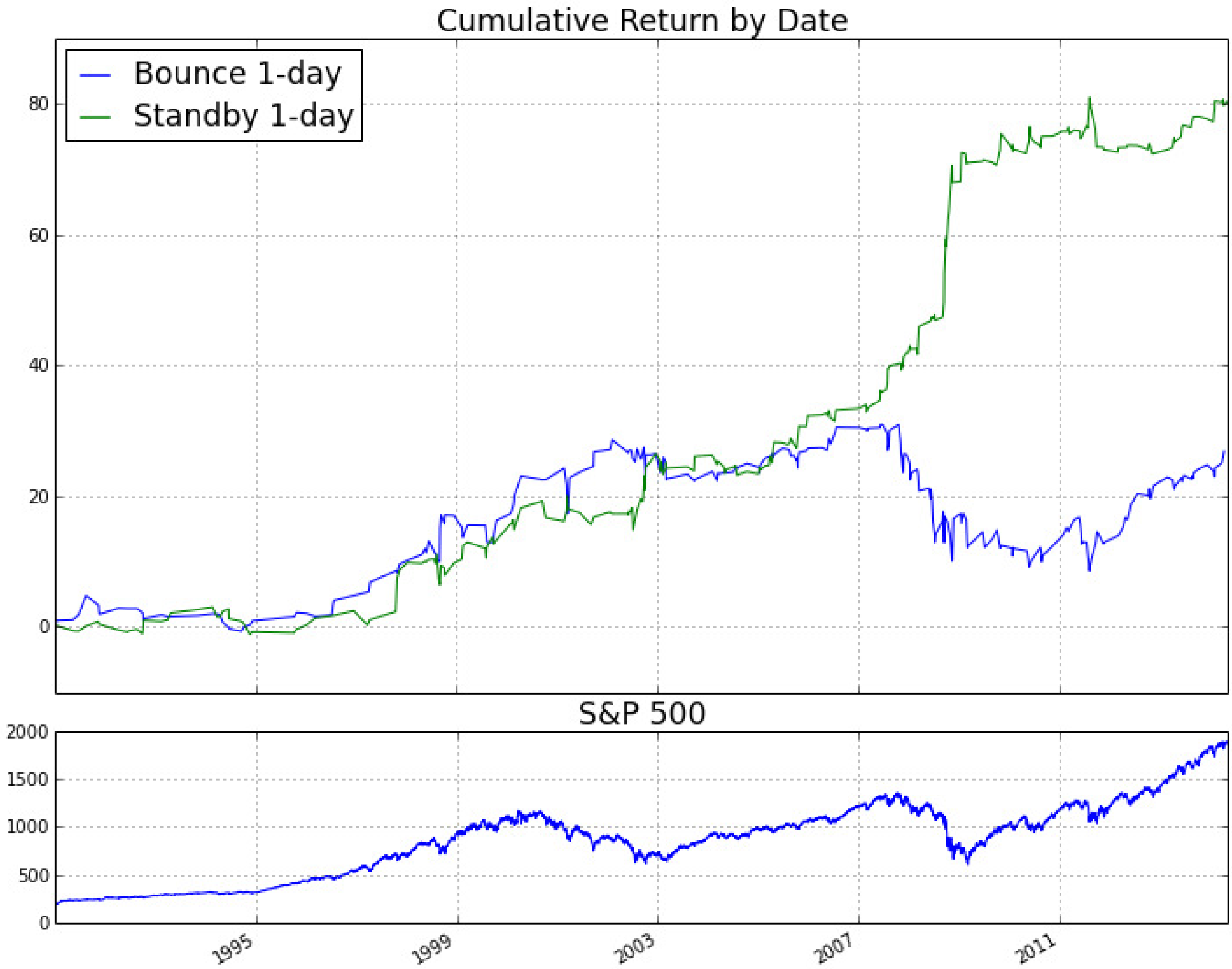
Historical Results By Holding Horizons

```
In [11]: events = OrderedDict()
         events['Bounce'] = (d_all.spike_bounce)
         events['Standby'] = (d_all.spike_standby)
```

Holding Horizon: 1 Trading Day

In [13]:
```
rs, details = study.show_events(d_all, events, [1],
                                fig_size = fig_size,

                                font_size = 'xx-large',
                                by_trade = False)
```



In [14]:
```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0,1, axis=1)
```

Out[14]:

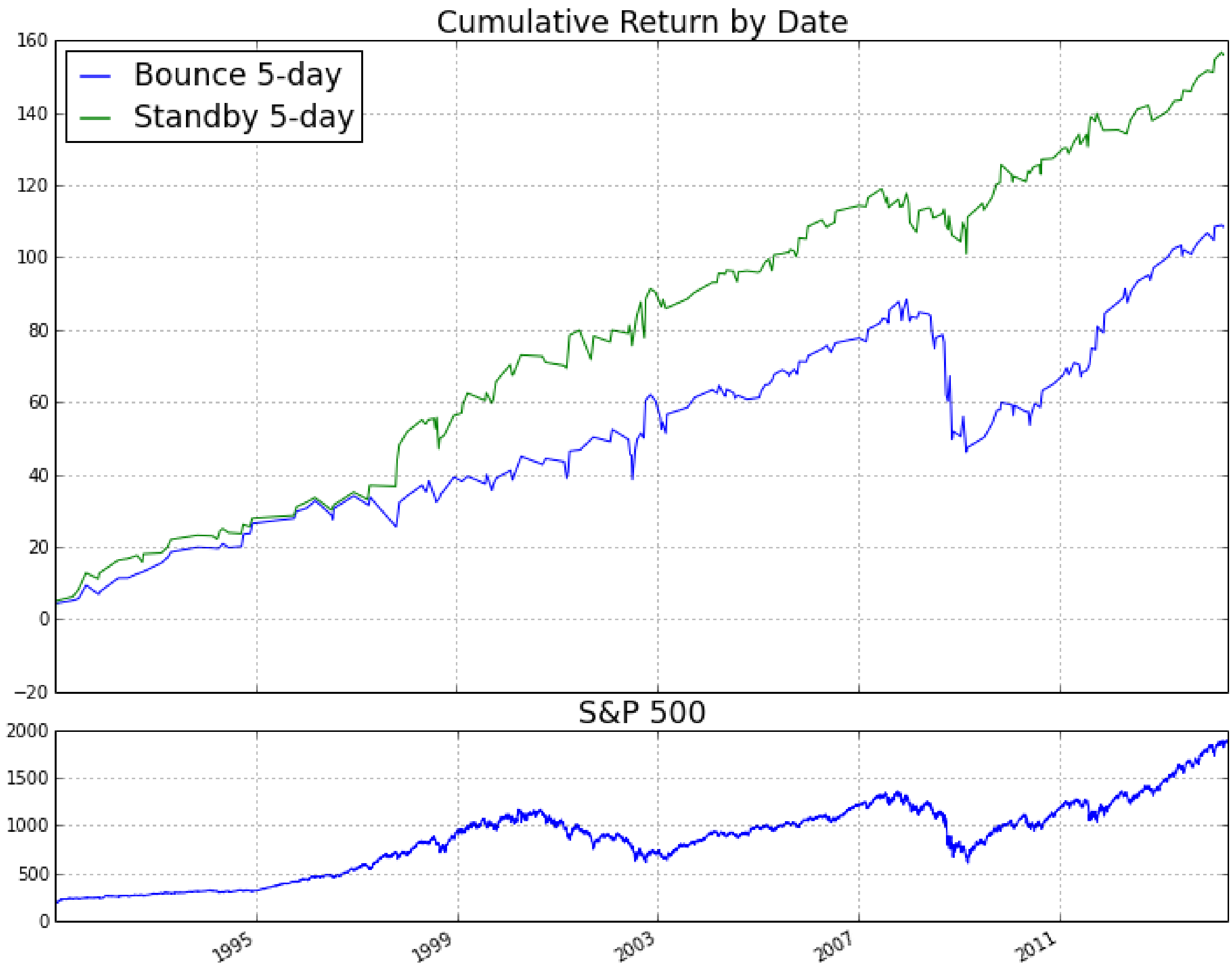
event	Bounce		Standby	
stat	avg	n	avg	n
ret_01	0.1	276	0.29	277

1 rows × 4 columns

Holding Horizon: 5 Trading Days

In [15]:

```
rs, details = study.show_events(d_all, events, [5],
                                fig_size = fig_size,
                                font_size = 'xx-large',
                                by_trade = False)
```



In [16]:

```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

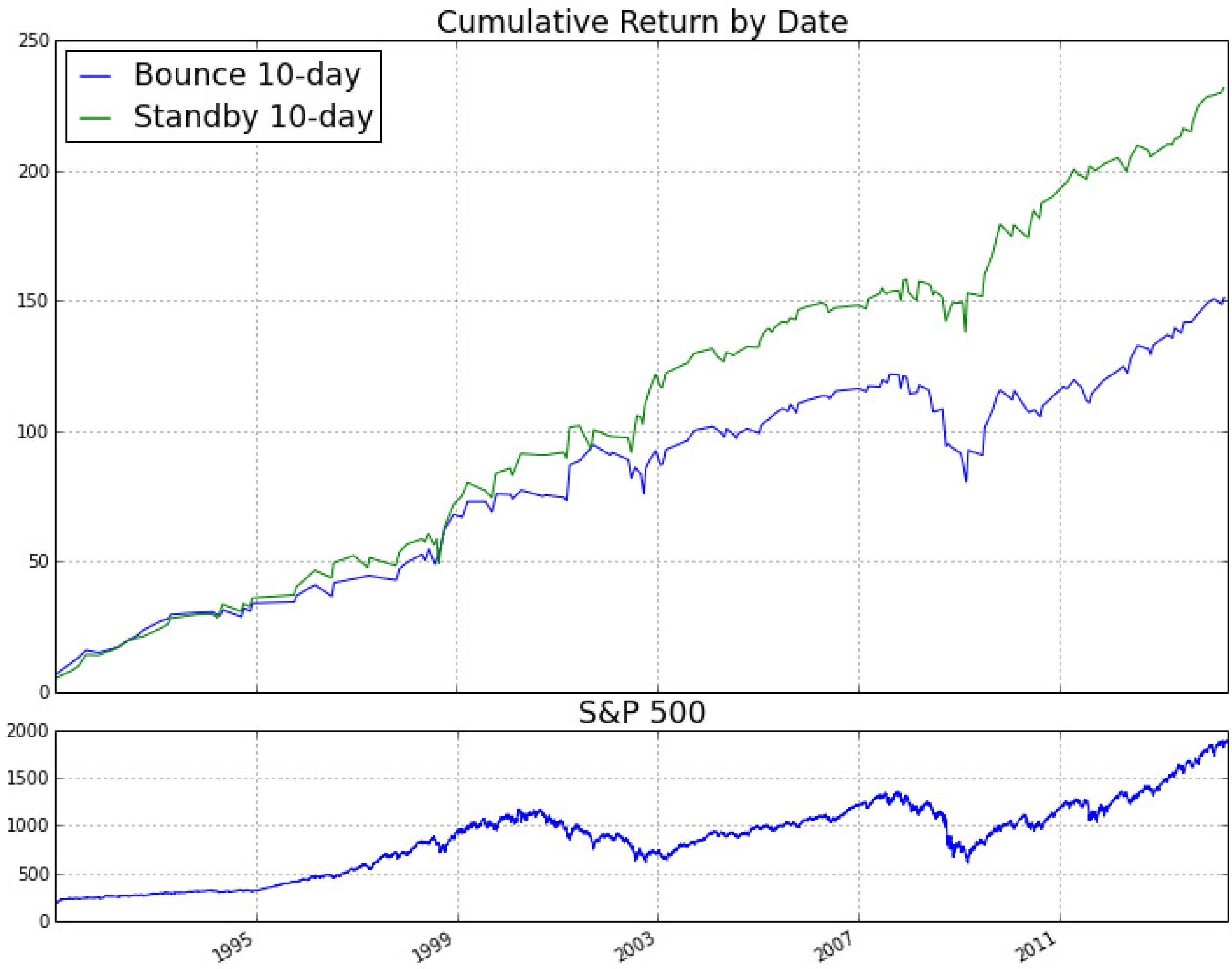
Out[16]:

event	Bounce		Standby	
stat	avg	n	avg	n
ret_05	0.52	210	0.74	210

1 rows × 4 columns

Holding Horizon: 10 Trading Days

In [17]:
```
rs, details = study.show_events(d_all, events, [10],
                                fig_size = fig_size,
                                font_size = 'xx-large',
                                by_trade = False)
```



In [18]:
```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

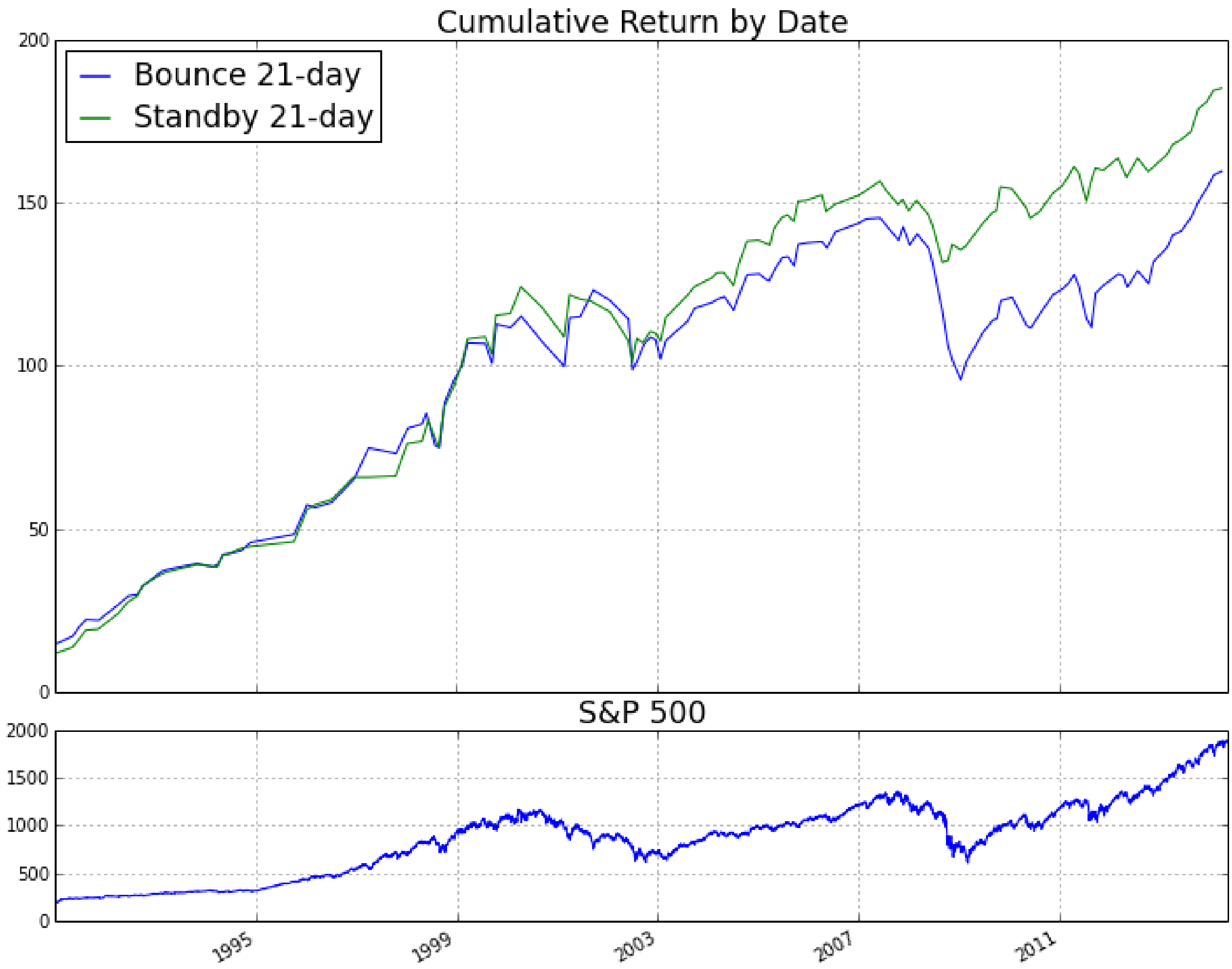
Out[18]:

event	Bounce		Standby	
stat	avg	n	avg	n
ret_10	0.93	163	1.4	166

1 rows × 4 columns

Holding Horizon: 21 Trading Days

```
In [19]: rs, details = study.show_events(d_all, events, [21],
                                         fig_size = fig_size,
                                         font_size = 'xx-large',
                                         by_trade = False)
```



```
In [20]: rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

Out[20]:

event	Bounce		Standby	
stat	avg	n	avg	n
ret_21	1.34	119	1.58	117

1 rows × 4 columns

Conclusion

Buying S&P 500 when the 20-Day NH-NL is oversold is more profitable than buying after it rebounds, historically.

Study 2: Spike Bounce Retrigger

Once Bitten, Twice Shy?

Event Definitions

Fresh Bounce

- Today is a "Spike Bounce" day.
- There was no "Spike Bounce" in the previous 4 days.

Bounce Retrigger

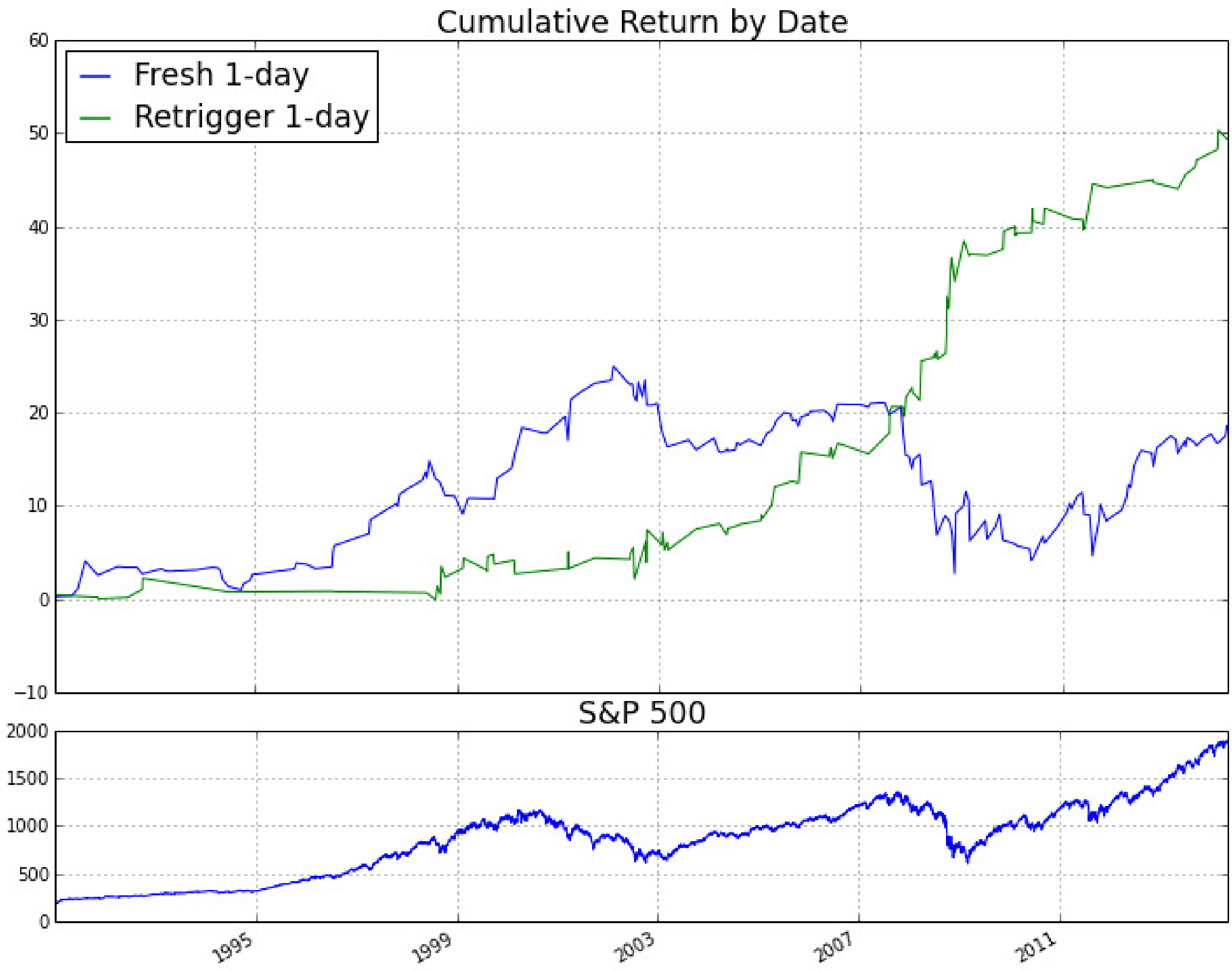
- Today is a "Spike Bounce" day.
- There was at least one "Spike Bounce" in the previous 4 days.

Historical Results By Holding Horizons

Holding Horizon: 1 Trading Day

In [29]:

```
rs, details = study.show_events(d_all, events, [1],
                                fig_size = fig_size,
                                font_size = 'xx-large',
                                by_trade = False)
```



In [30]:

```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

Out[30]:

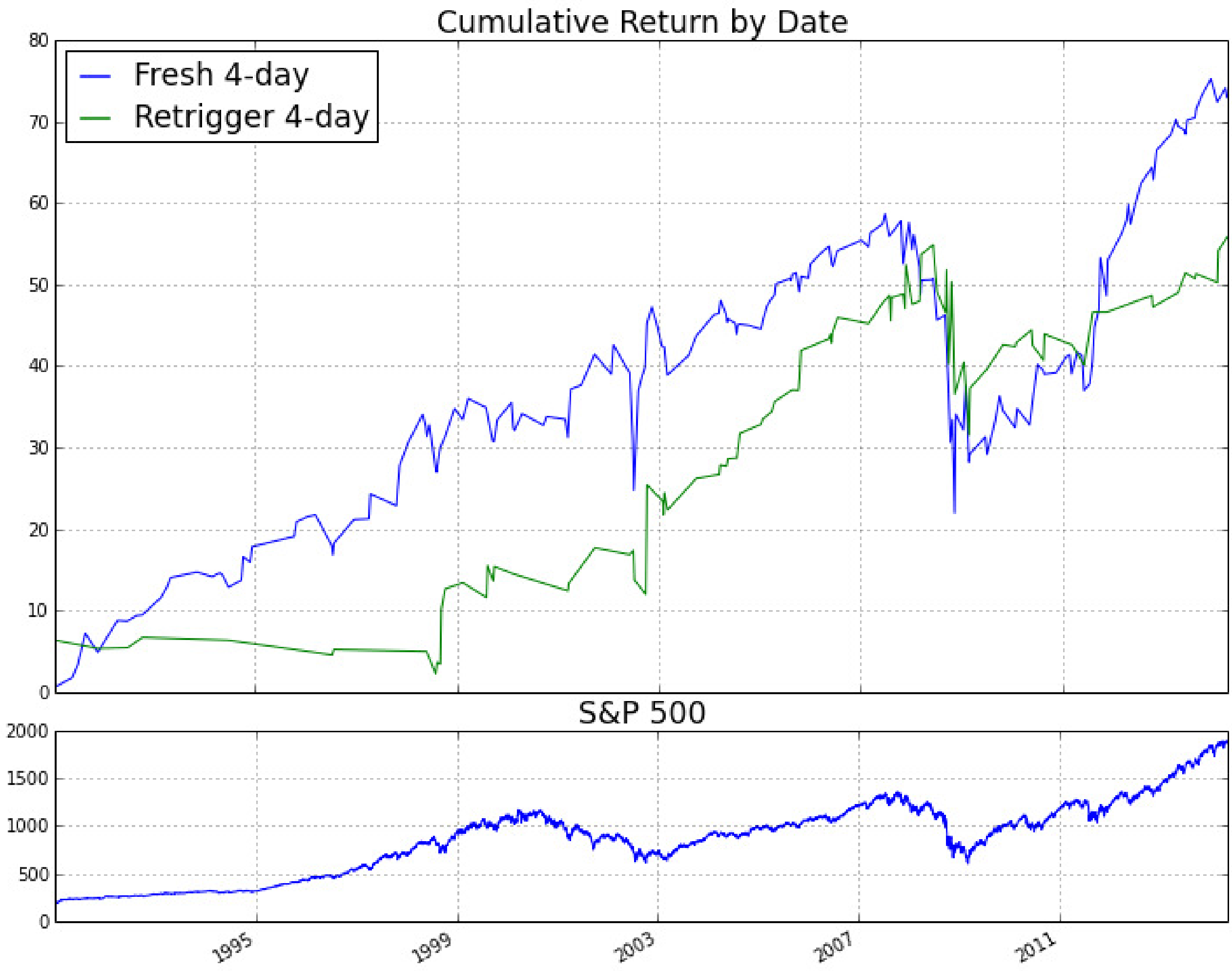
event	**Fresh**		**Retrigger**	
stat	**avg**	**n**	**avg**	**n**
ret_01	0.1	191	0.4	124

1 rows × 4 columns

Holding Horizon: 4 Trading Days

In [31]:

```
rs, details = study.show_events(d_all, events, [4],
                                fig_size = fig_size,
                                font_size = 'xx-large',
                                by_trade = False)
```



In [32]:

```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

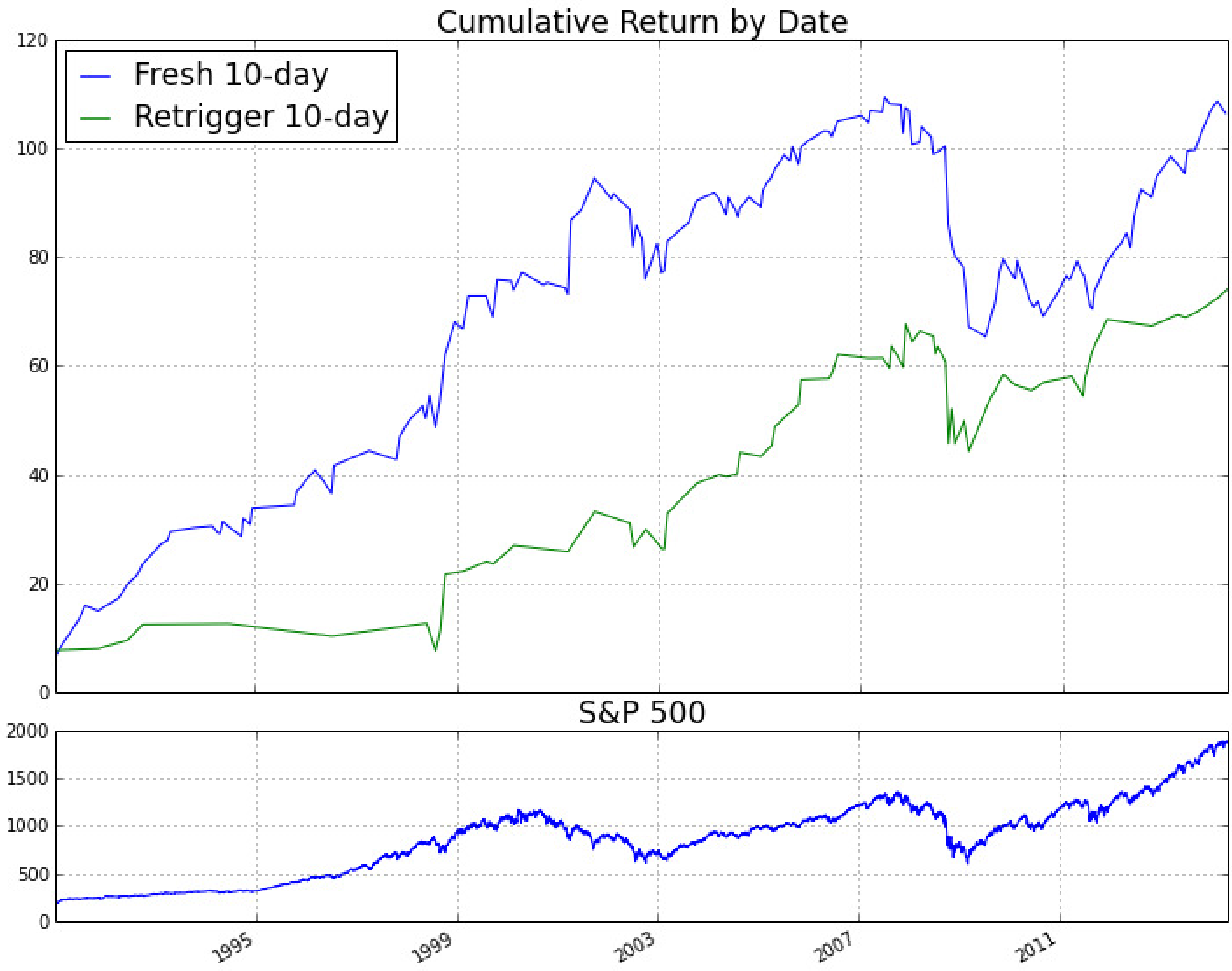
Out[32]:

event	Fresh		Retrigger	
stat	avg	n	avg	n
ret_04	0.38	191	0.57	98

1 rows × 4 columns

Holding Horizon 10 Trading Days

```
In [33]: rs, details = study.show_events(d_all, events, [10],
                                          fig_size = fig_size,
                                          font_size = 'xx-large',
                                          by_trade = False)
```



```
In [34]: rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

Out[34]:

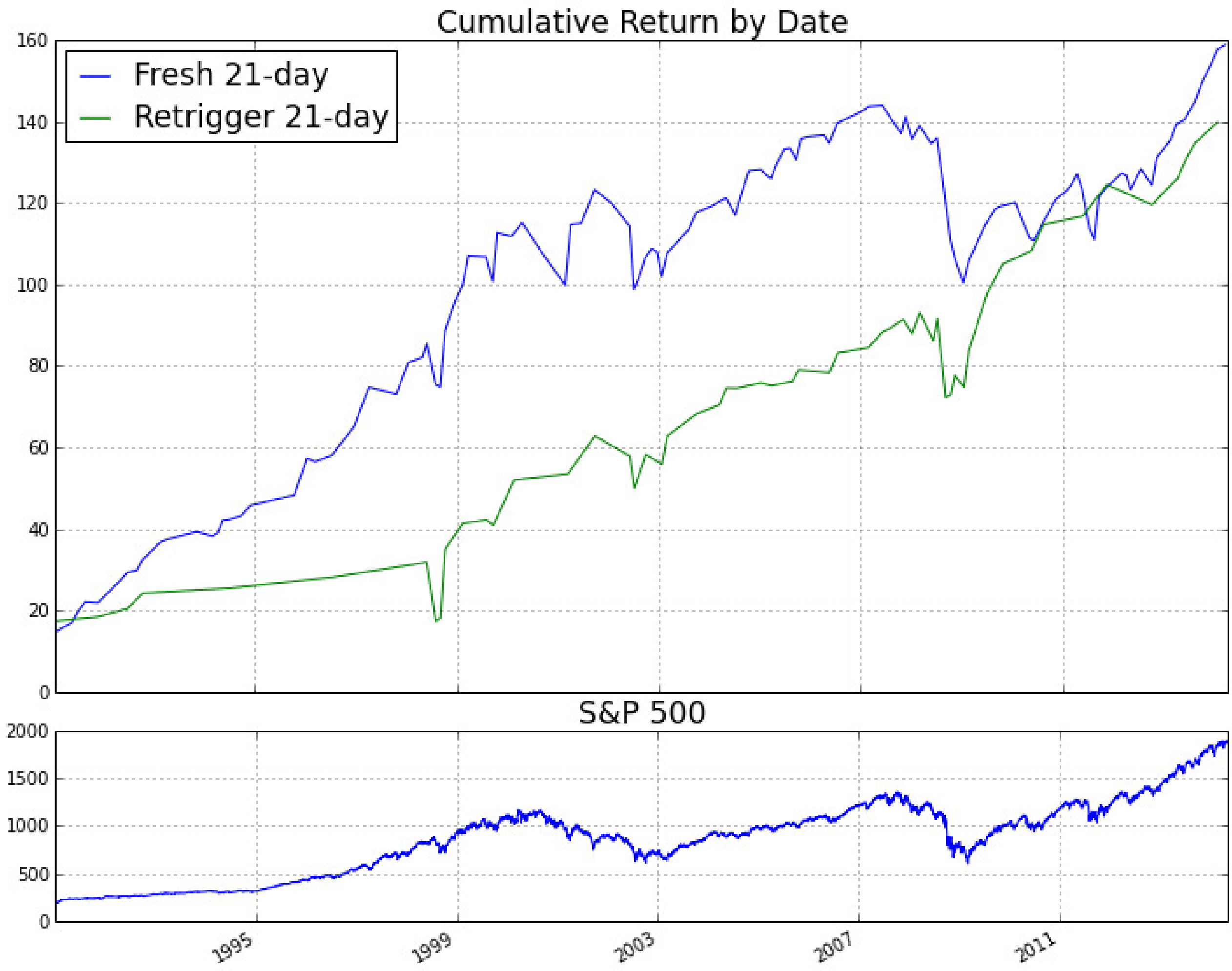
event	Fresh		Retrigger	
stat	avg	n	avg	n
ret_10	0.69	154	1.07	69

1 rows × 4 columns

Holding Horizon 21 Trading Days

In [44]:

```
rs, details = study.show_events(d_all, events, [21],
                                fig_size = fig_size,
                                font_size = 'xx-large',
                                by_trade = False)
```



In [45]:

```
rs.swaplevel(0, 1, axis =1)[['avg', 'n']].swaplevel(0, 1, axis =1)
```

Out[45]:

event	Fresh		Retrigger	
stat	**avg**	**n**	**avg**	**n**
ret_21	1.35	118	2.41	58

1 rows × 4 columns

"Once Bitten, Twice Shy?"

The Next Time a Spike Bounce Retriggers ...

Q&A

This Presentation is Powered By

IP[y]: IPython
Interactive Computing

```
In [ ]:
```