

# Portfolio Sector Tear Sheets

```
In [1]: # Import new libraries
import pandas as pd
import numpy as np
from pandas_datareader import data as web
import empyrical
import pyfolio as pf

import warnings
warnings.filterwarnings(action='ignore')

/opt/anaconda3/envs/pyvizenv/lib/python3.7/site-packages/pyfolio/pos.py:27: User
Warning: Module "zipline.assets" not found; multipliers will not be applied to
position notionalss.
'Module "zipline.assets" not found; multipliers will not be applied' +
```

## Fix Broken Pyfolio Functions

```
In [2]: # Replacement functions for broken pyfolio code
def get_max_drawdown_underwater_f(underwater):
    """
    Determines peak, valley, and recovery dates given an 'underwater'
    DataFrame.

    An underwater DataFrame is a DataFrame that has precomputed
    rolling drawdown.

    Parameters
    -----
    underwater : pd.Series
        Underwater returns (rolling drawdown) of a strategy.

    Returns
    -----
    peak : datetime
        The maximum drawdown's peak.
    valley : datetime
        The maximum drawdown's valley.
    recovery : datetime
        The maximum drawdown's recovery.
    """

    #valley = np.argmin(underwater) # end of the period
    valley = underwater.index[np.argmin(underwater)] # end of the period

    # Find first 0
    peak = underwater[:valley][underwater[:valley] == 0].index[-1]
    # Find last 0
    try:
        recovery = underwater[valley:][underwater[valley:] == 0].index[0]
    except IndexError:
        recovery = np.nan # drawdown not recovered
    return peak, valley, recovery

def get_symbol_returns_from_yahoo_f(symbol, start=None, end=None):
    """
```

```

Wrapper for pandas.io.data.get_data_yahoo().
Retrieves prices for symbol from yahoo and computes returns
based on adjusted closing prices.

Parameters
-----
symbol : str
    Symbol name to load, e.g. 'SPY'
start : pandas.Timestamp compatible, optional
    Start date of time period to retrieve
end : pandas.Timestamp compatible, optional
    End date of time period to retrieve

Returns
-----
pandas.DataFrame
    Returns of symbol in requested period.
"""

try:
    px = web.get_data_yahoo(symbol, start=start, end=end)
    px['date'] = px.index.to_list()
    #px['date'] = px['date'].apply(lambda x: pd.Timestamp(x))
    #px['date'] = pd.to_datetime(px['date'])
    #px['date'] = pd.to_datetime(px['date'], unit='s')
    px.set_index('date', drop=False, inplace=True)

    #px.index.rename('date', inplace=True)
    rets = px[['Adj Close']].pct_change().dropna()
    rets.rename(columns={'Adj Close': "adjclose"}, inplace=True)
except Exception as e:
    warnings.warn(
        'Yahoo Finance read failed: {}, falling back to Google'.format(e),
        UserWarning)
    px = web.get_data_google(symbol, start=start, end=end)
    rets = px[['Close']].pct_change().dropna()
if rets.index.tzinfo is None:
    rets.index = rets.index.tz_localize("UTC")
rets.columns = [symbol]
return rets

# Overriding broken pyfolio functions
empirical.utils.get_symbol_returns_from_yahoo = get_symbol_returns_from_yahoo_f
pf.timeseries.get_max_drawdown_underwater = get_max_drawdown_underwater_f

```

## Information Technology Sector Tear Sheet

```
In [3]: # Fetch historical data for Info Tech sector
stock_rets = pf.utils.get_symbol_rets('VGT')
```

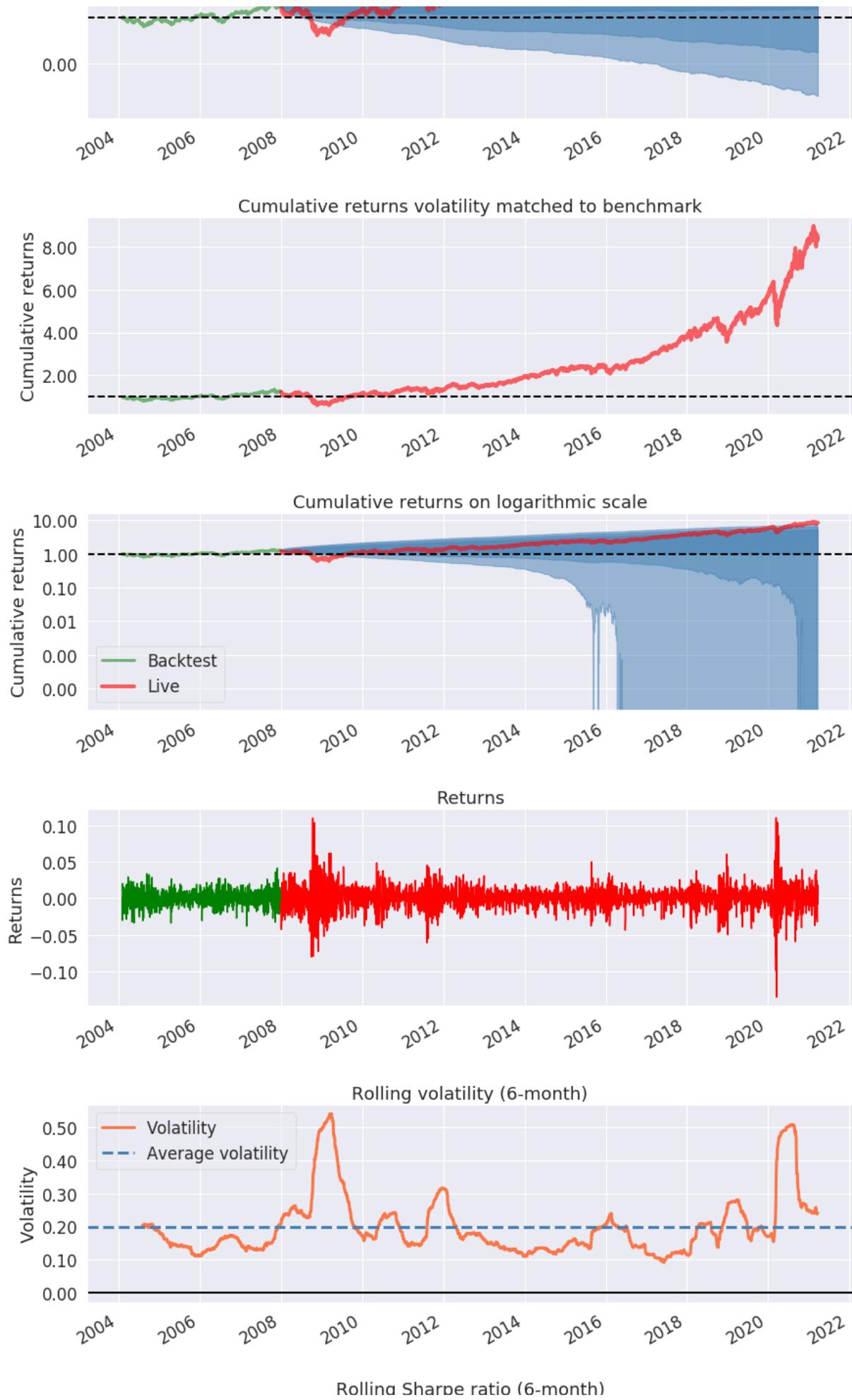
```
In [4]: # Create tear sheet for Info Tech sector
pf.create_returns_tear_sheet(stock_rets, live_start_date='2008-01-02')
```

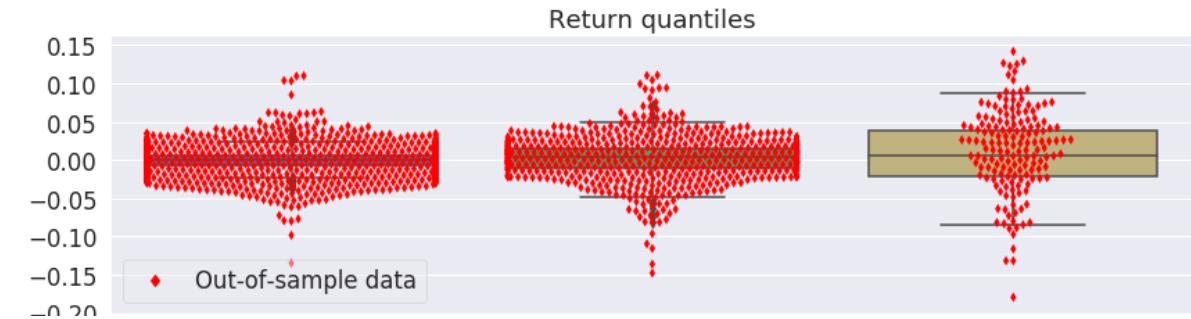
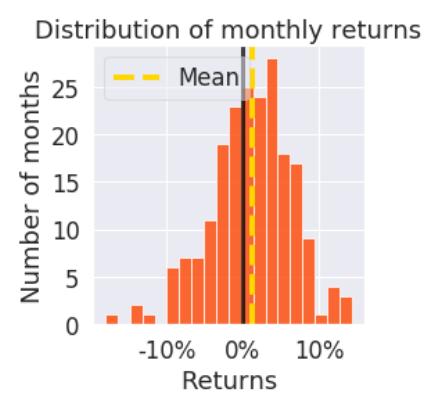
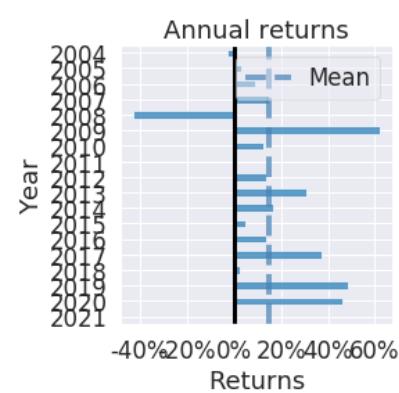
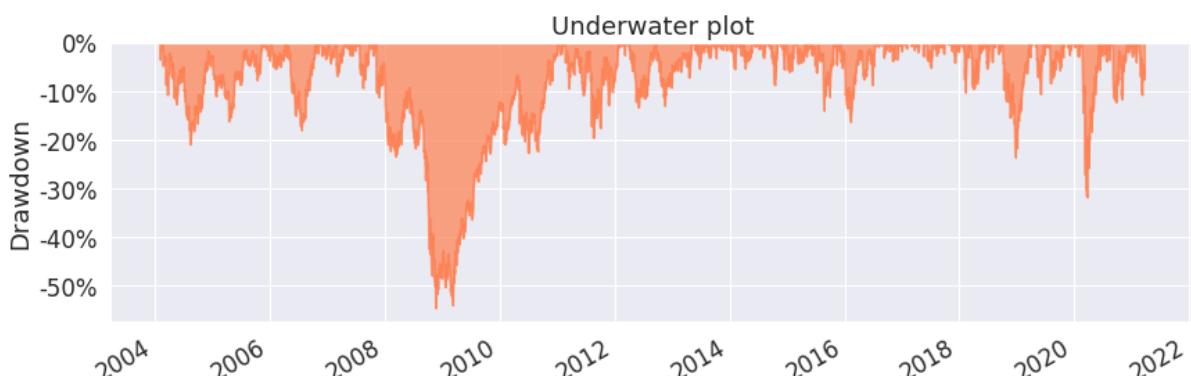
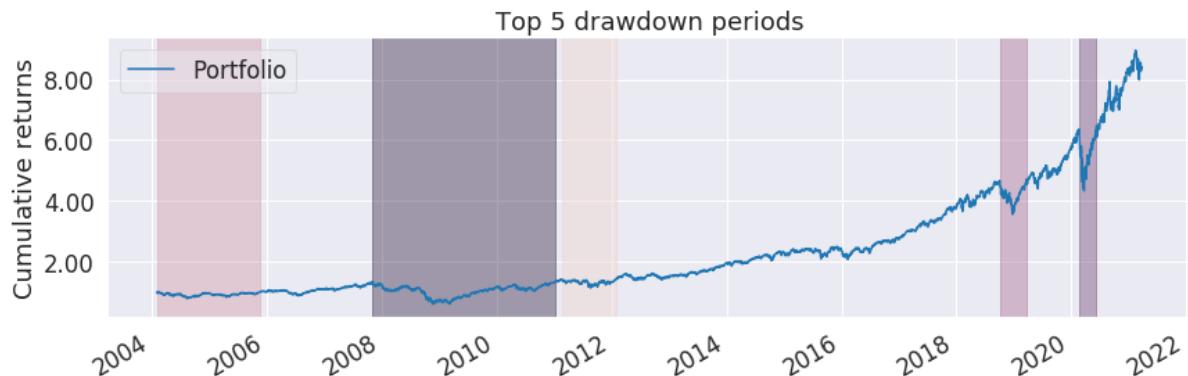
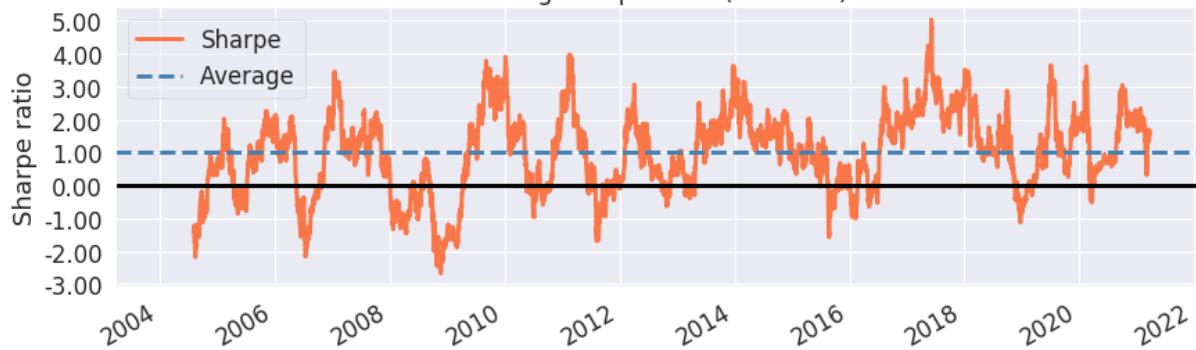
<b>Start date</b>	2004-02-02
<b>End date</b>	2021-03-23
<b>In-sample months</b>	46
<b>Out-of-sample months</b>	158

Start date	In-sample	Out-of-sample	1004-02-All		
End date	2021-03-23				
In-sample months	46				
Out-of-sample months	158				
	In-sample	Out-of-sample	All		
<b>Annual return</b>	5.9%	15.5%	13.2%		
<b>Cumulative returns</b>	25.0%	568.8%	736.0%		
<b>Annual volatility</b>	16.4%	23.3%	21.9%		
<b>Sharpe ratio</b>	0.43	0.73	0.68		
<b>Calmar ratio</b>	0.28	0.30	0.24		
<b>Stability</b>	0.74	0.95	0.89		
<b>Max drawdown</b>	-21.0%	-51.3%	-54.6%		
<b>Omega ratio</b>	1.07	1.15	1.13		
<b>Sortino ratio</b>	0.61	1.04	0.96		
<b>Skew</b>	-0.15	-0.18	-0.17		
<b>Kurtosis</b>	0.93	8.72	8.83		
<b>Tail ratio</b>	0.95	0.92	0.93		
<b>Daily value at risk</b>	-2.0%	-2.9%	-2.7%		
Worst drawdown periods	Net drawdown in %	Peak date	Valley date	Recovery date	Duration
0	54.63	2007-10-31	2008-11-20	2011-01-06	832
1	31.84	2020-02-19	2020-03-23	2020-06-09	80
2	23.67	2018-10-03	2018-12-24	2019-03-21	122
3	21.01	2004-02-02	2004-08-12	2005-11-17	469
4	19.64	2011-02-17	2011-08-19	2012-02-01	250



## portfolio\_tear\_sheets





-v.vv

Daily

Weekly

Monthly

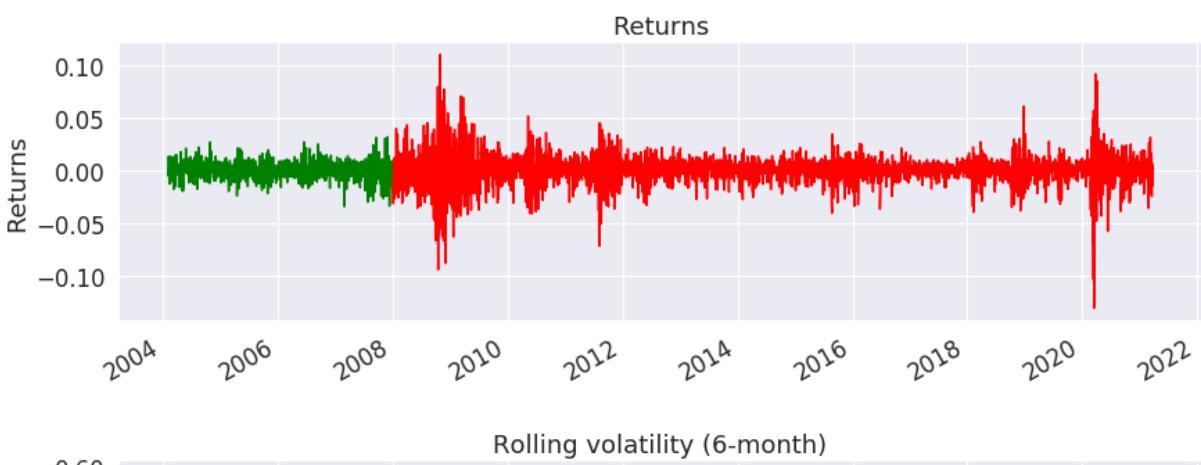
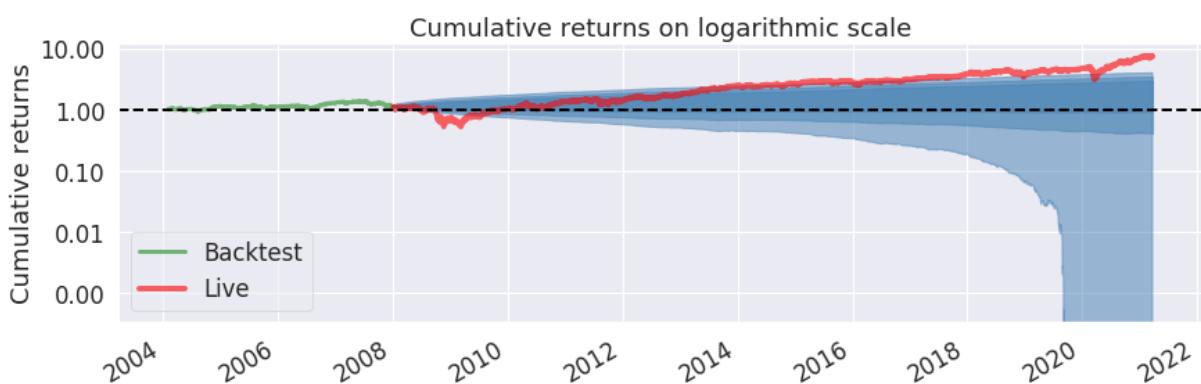
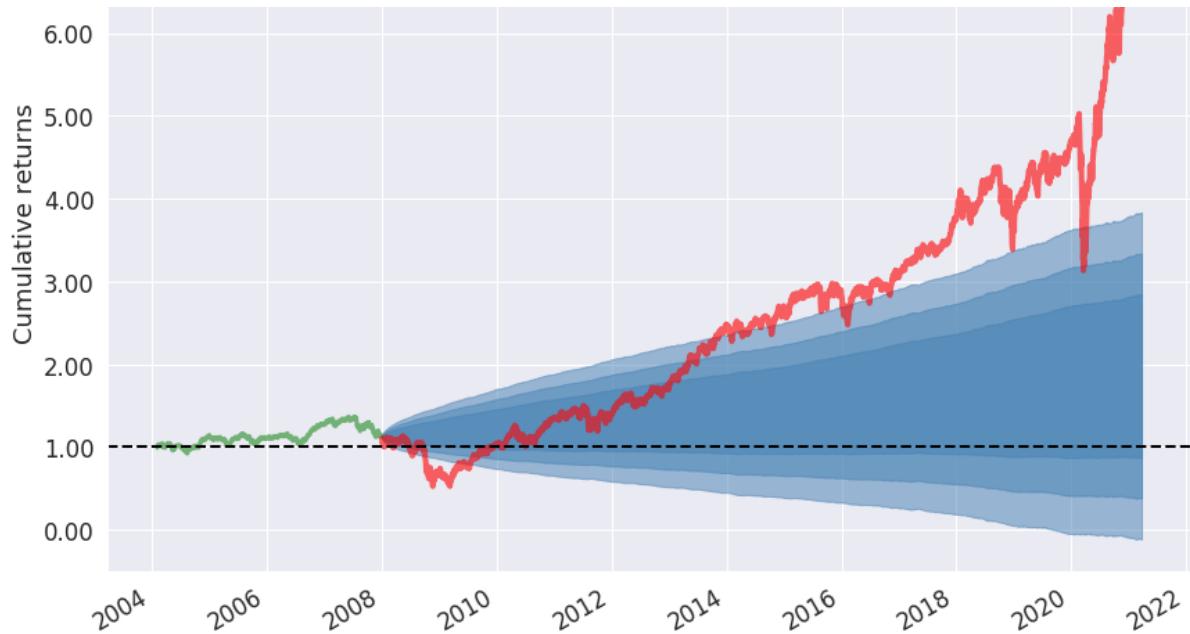
# Consumer Discretionary Sector Tear Sheet

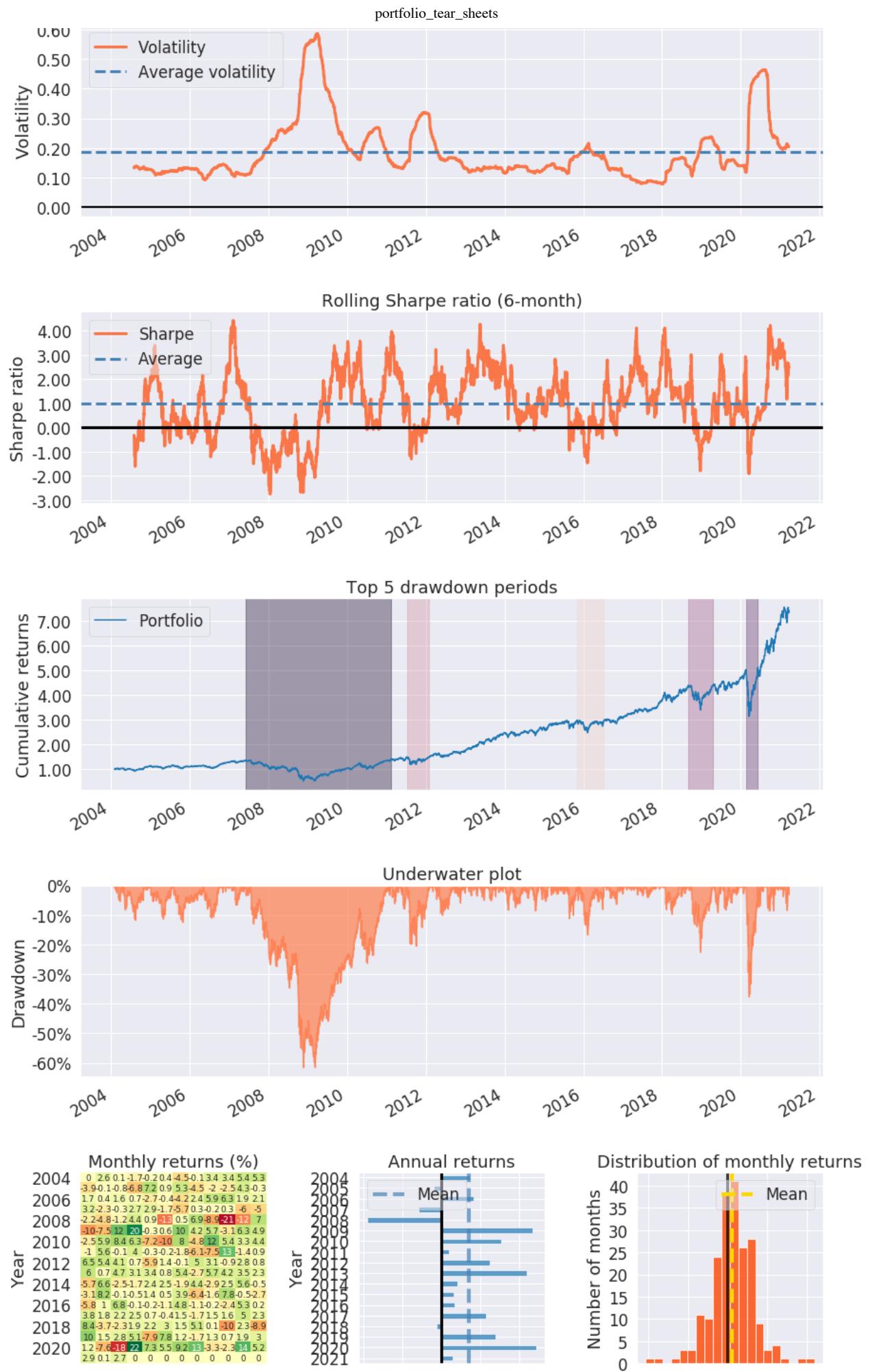
```
In [5]: # Fetch historical data for Consumer Discretionary sector
stock_rets = pf.utils.get_symbol_rets('VCR')
```

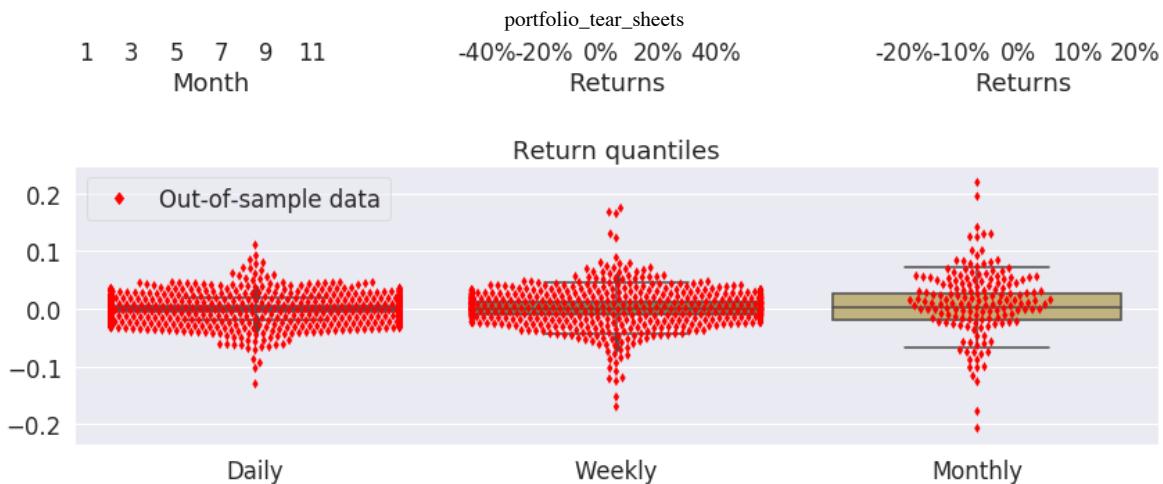
```
In [6]: # Create tear sheet for Consumer Discretionary sector
pf.create_returns_tear_sheet(stock_rets, live_start_date='2008-01-02')
```

<b>Start date</b>	2004-02-02					
<b>End date</b>	2021-03-23					
<b>In-sample months</b>	46					
<b>Out-of-sample months</b>	158					
	In-sample	Out-of-sample	All			
<b>Annual return</b>	3.1%	15.2%	12.3%			
<b>Cumulative returns</b>	12.8%	548.5%	631.2%			
<b>Annual volatility</b>	13.6%	22.9%	21.1%			
<b>Sharpe ratio</b>	0.29	0.73	0.66			
<b>Calmar ratio</b>	0.17	0.28	0.20			
<b>Stability</b>	0.70	0.95	0.86			
<b>Max drawdown</b>	-18.1%	-53.6%	-61.5%			
<b>Omega ratio</b>	1.05	1.15	1.13			
<b>Sortino ratio</b>	0.41	1.03	0.92			
<b>Skew</b>	-0.13	-0.38	-0.37			
<b>Kurtosis</b>	1.23	8.56	9.44			
<b>Tail ratio</b>	0.93	0.95	0.93			
<b>Daily value at risk</b>	-1.7%	-2.8%	-2.6%			
	Worst drawdown periods	Net drawdown in %	Peak date	Valley date	Recovery date	Duration
0		61.54	2007-06-04	2008-11-20	2011-02-07	961
1		37.68	2020-02-20	2020-03-18	2020-06-08	78
2		22.61	2018-09-04	2018-12-24	2019-04-16	161
3		20.49	2011-07-07	2011-10-03	2012-02-03	152
4		16.70	2015-11-03	2016-02-11	2016-07-12	181









## Healthcare Sector Tear Sheet

```
In [7]: # Fetch historical data for Healthcare sector
stock_rets = pf.utils.get_symbol_rets('VHT')
```

```
In [8]: # Create tear sheet for Healthcare sector
pf.create_returns_tear_sheet(stock_rets, live_start_date='2008-01-02')
```

<b>Start date</b>	2004-02-02		
<b>End date</b>	2021-03-23		
<b>In-sample months</b>	46		
<b>Out-of-sample months</b>	158		
	In-sample	Out-of-sample	All
<b>Annual return</b>	5.9%	12.1%	10.7%
<b>Cumulative returns</b>	25.3%	352.8%	467.1%
<b>Annual volatility</b>	11.5%	18.8%	17.4%
<b>Sharpe ratio</b>	0.56	0.70	0.67
<b>Calmar ratio</b>	0.43	0.31	0.27
<b>Stability</b>	0.86	0.96	0.91
<b>Max drawdown</b>	-13.8%	-38.8%	-39.1%
<b>Omega ratio</b>	1.10	1.14	1.13
<b>Sortino ratio</b>	0.80	0.99	0.95
<b>Skew</b>	-0.04	-0.17	-0.16
<b>Kurtosis</b>	1.05	10.35	11.21
<b>Tail ratio</b>	0.97	0.94	0.95
<b>Daily value at risk</b>	-1.4%	-2.3%	-2.1%
<b>Worst drawdown periods</b>	<b>Net drawdown in %</b>	<b>Peak date</b>	<b>Valley date</b>
<b>0</b>	39.12	2007-12-10	2009-03-05
			<b>Recovery date</b>
			2011-03-03
			<b>Duration</b>
			844

Worst drawdown periods	Net drawdown in %	Peak date	Valley date	Recovery date	Duration
------------------------	-------------------	-----------	-------------	---------------	----------

1	28.85	2020-02-19	2020-03-23	2020-06-08	79
---	-------	------------	------------	------------	----

2	20.26	2015-07-20	2016-02-11	2017-06-01	489
---	-------	------------	------------	------------	-----

3	18.31	2011-05-18	2011-08-08	2012-02-01	186
---	-------	------------	------------	------------	-----

4	16.97	2018-10-01	2018-12-24	2019-11-08	290
---	-------	------------	------------	------------	-----

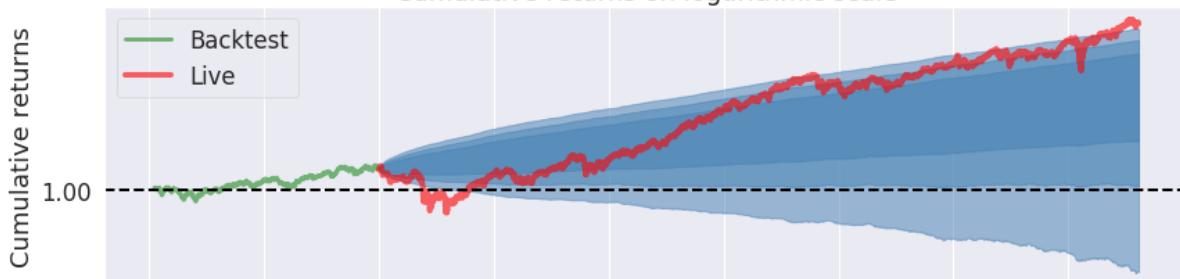
Cumulative returns



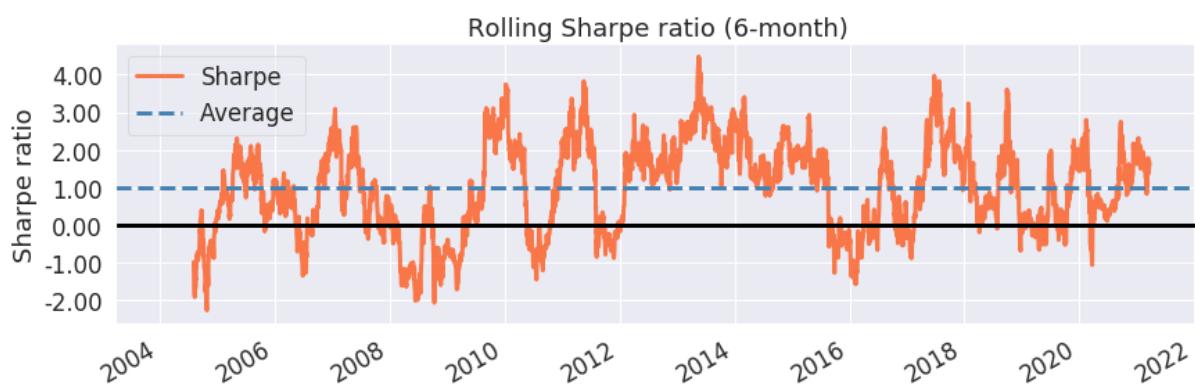
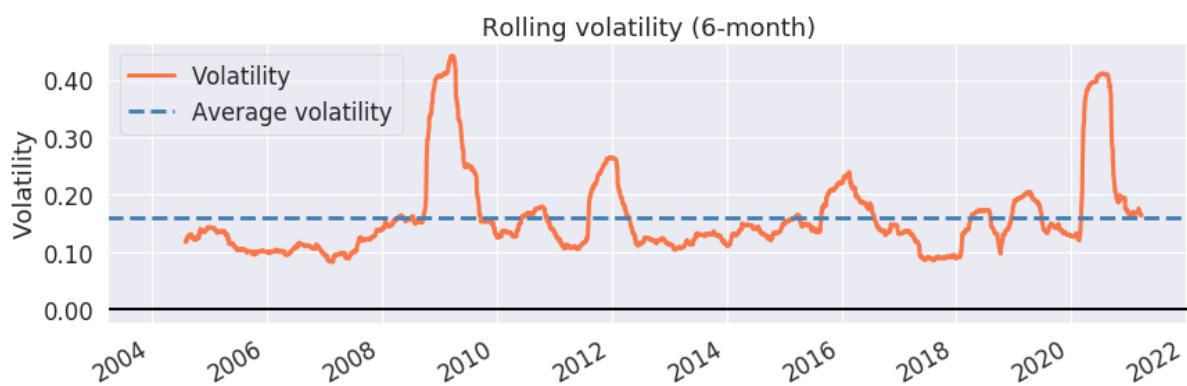
Cumulative returns volatility matched to benchmark

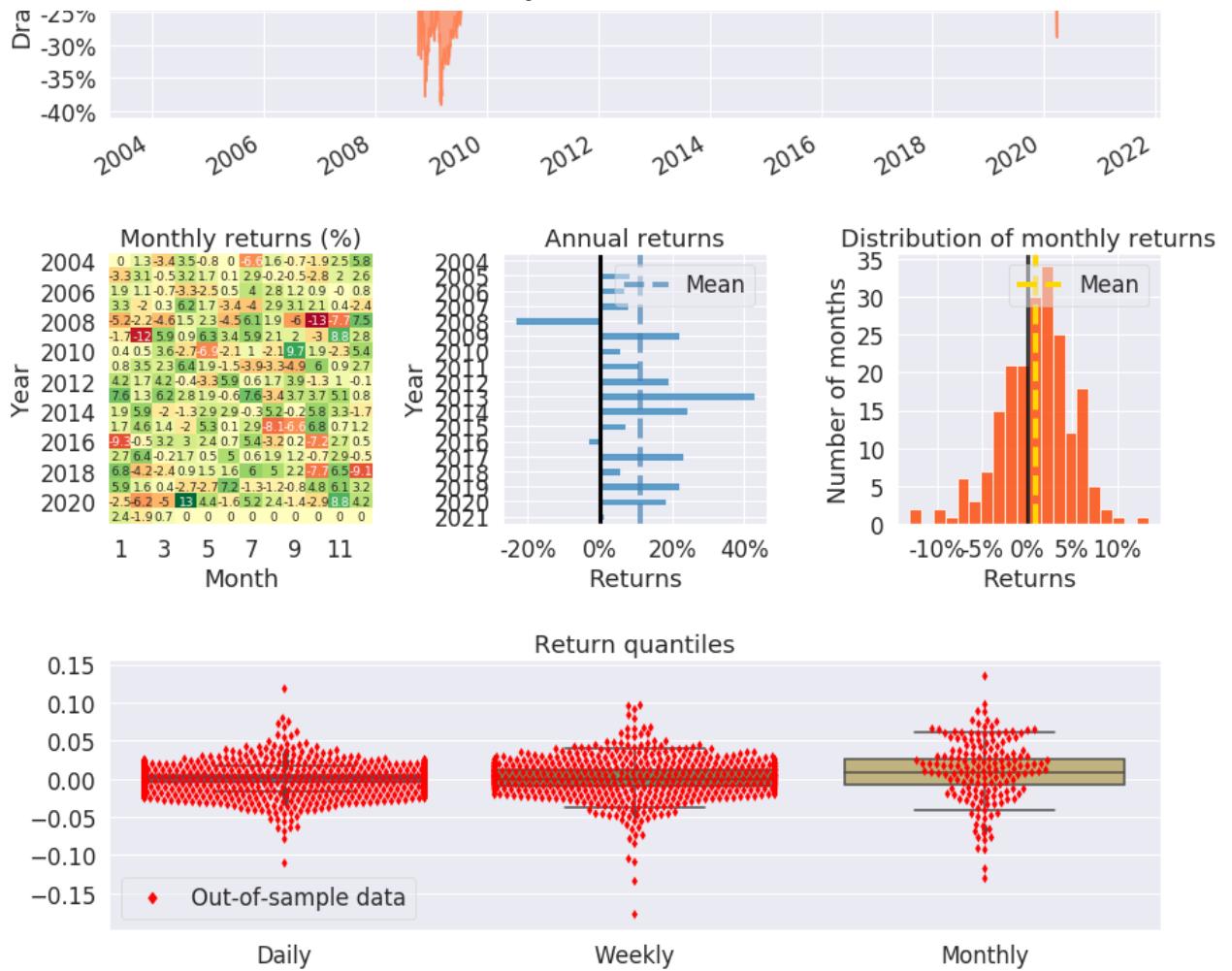


Cumulative returns on logarithmic scale



portfolio\_tear\_sheets





In [ ]: