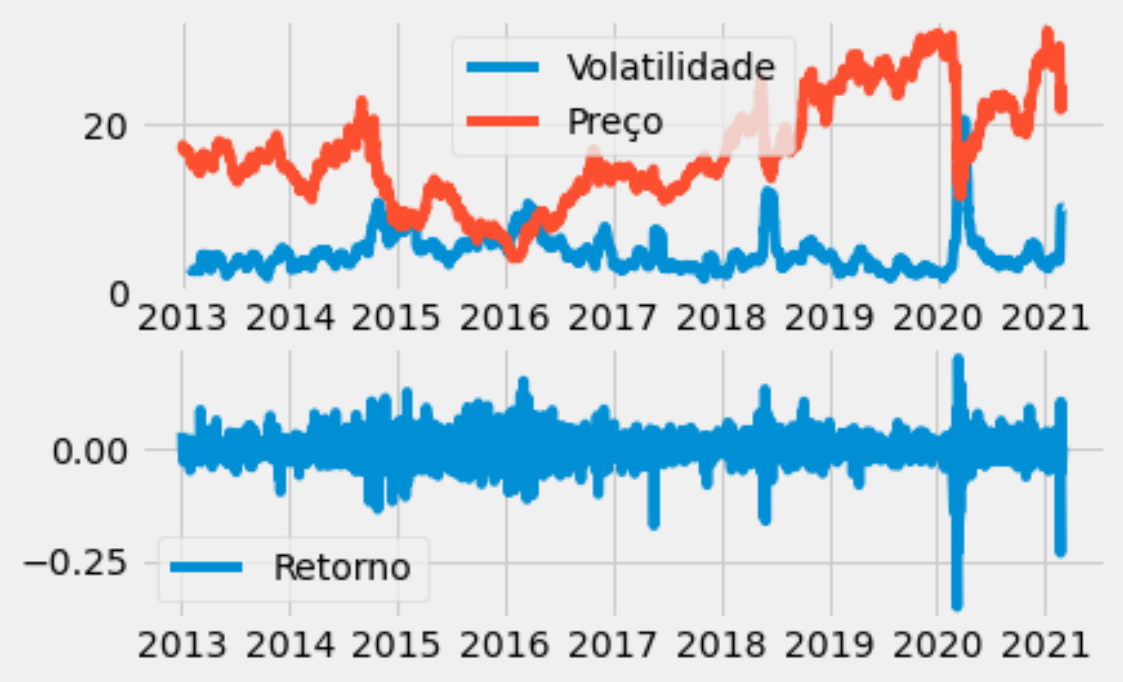
Estratégia de reversão a média

April 1, 2021

[9]:
```
import pandas as pd
import numpy as np
from datetime import datetime
from pylab import mpl, plt
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
import yfinance as yf
import pyfolio as pf
import warnings
warnings.filterwarnings('ignore')
```

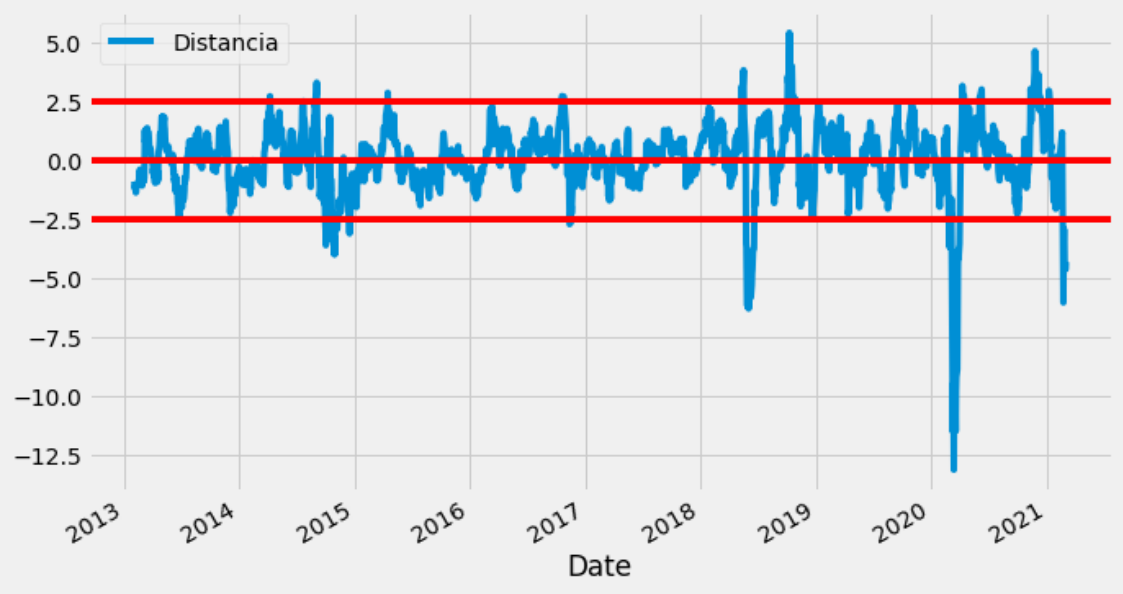
[40]:
```
petr = yf.download(tickers='PETR4.SA', start='2013-01-01')[['Adj Close']]
petr
petr['LogReturn'] = np.log(petr['Adj Close'] / petr['Adj Close'].shift(1))
petr['Volatility'] = petr['LogReturn'].rolling(21).std()*np.sqrt(252)

plt.subplot(2,1,1)
plt.plot(petr['Volatility']*10, label = 'Volatilidade')
plt.plot(petr['Adj Close'], label = 'Preço')
plt.legend()

plt.subplot(2,1,2)
plt.plot(petr['LogReturn'], label = 'Retorno')
plt.legend(loc='best')
plt.show()
```

```
[**********100%**********]  1 of 1 completed
```



[48]:
```
# Definindo a Média Movel Simples de 22 dias
SMA = 22

petr['SMA'] = petr['Adj Close'].rolling(SMA).mean()

# O limite para a geração do sinal é definido.

limite = 2.5

petr['Distancia'] = petr['Adj Close'] - petr['SMA']

# Plotando

petr['Distancia'].dropna().plot(figsize=(10,6), legend = True)
plt.axhline(limite, color = 'red')
plt.axhline(-limite, color = 'red')
plt.axhline(0, color = 'red')
```

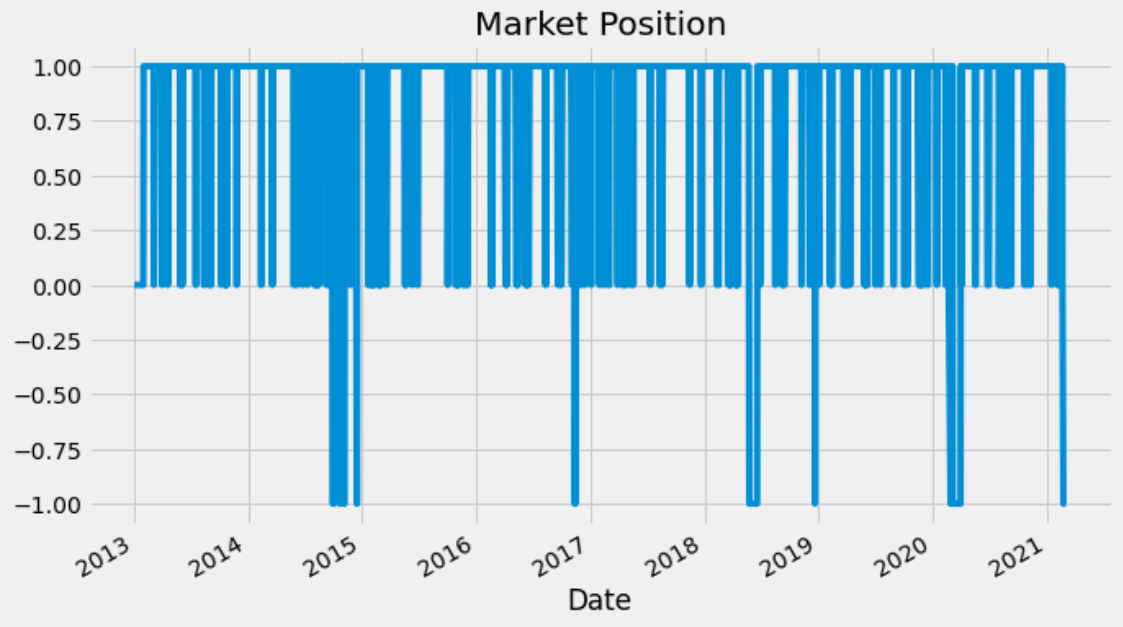
[48]:
```
<matplotlib.lines.Line2D at 0x22c3fbc3670>
```



[51]:
```
petr['Posicao'] = np.where(petr['Distancia'] < limite,
                           -1, np.nan)

petr['Posicao'] = np.where(petr['Distancia'] > -limite,
                           1, petr['Posicao'])
petr['Posicao'] = np.where(petr['Distancia']*
                           petr['Distancia'].shift(1) < 0, 0, petr['Posicao'])
petr['Posicao'] = petr['Posicao'].ffill().fillna(0)
```

[52]:
```
petr['Posicao'].plot(ylim = [-1.1, 1.1],
                     title = 'Market Position',
                     figsize = (10,6))
```

[52]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x22c41262100>
```



[53]:
```
petr['estrategia'] = petr['Posicao'].shift(1)*petr['LogReturn']

petr[['LogReturn', 'estrategia']].dropna().cumsum(
        ).apply(np.exp).plot(figsize = (10,6))
```

[53]: <matplotlib.axes._subplots.AxesSubplot at 0x22c4125d4c0>