

レシピ38 株価見通しの計算（P100）

2022/02/05

- このレシピでは、ある会社の株価（2010年初旬～2017年中旬）を利用する
- Booleanインデックス法を使って、この期間の終値の最高・最低10%のSeriesを抽出する
- それから全点をプロットし、その上位下位10%をハイライトする

(1) SLB社の株価データを読み込み、DateカラムをインデックスにしてDatetimeIndexに変換する

In [4]:

```
import pandas as pd
slb = pd.read_csv('slb_stock.csv', index_col='Date', parse_dates=['Date'])
slb.head()
```

Out[4]:

	Open	High	Low	Close	Volume
Date					
2010-01-04	66.39	67.20	66.12	67.11	5771234
2010-01-05	66.99	67.62	66.73	67.30	7366270
2010-01-06	67.17	68.94	67.03	68.80	9949946
2010-01-07	68.49	69.81	68.21	69.51	7700297
2010-01-08	69.19	72.00	69.09	70.65	13487621

(2) 終値をSeriesで選択し、要約統計量をSeriesで返す

- describe(percentiles=[.1, .9])
- describeは最小値（min）、中央値（50%）、最大値（max）は常に表示されるが、それ以外の範囲はリストで指定できる

In [25]:

```
slb_close = slb['Close']
slb_summary = slb_close.describe(percentiles=[.1, .9])
slb_summary
```

Out[25]:

```
count    1895.000000
mean      79.121905
std       11.767802
min       51.750000
10%       64.892000
50%       78.000000
90%       93.248000
max       117.950000
Name: Close, dtype: float64
```

(3) Booleanインデックス法を使って、最高および最低の10パーセンタイル内の終値を選ぶ

- .locを使えば、describeからも取得できるんだ...

In [27]:

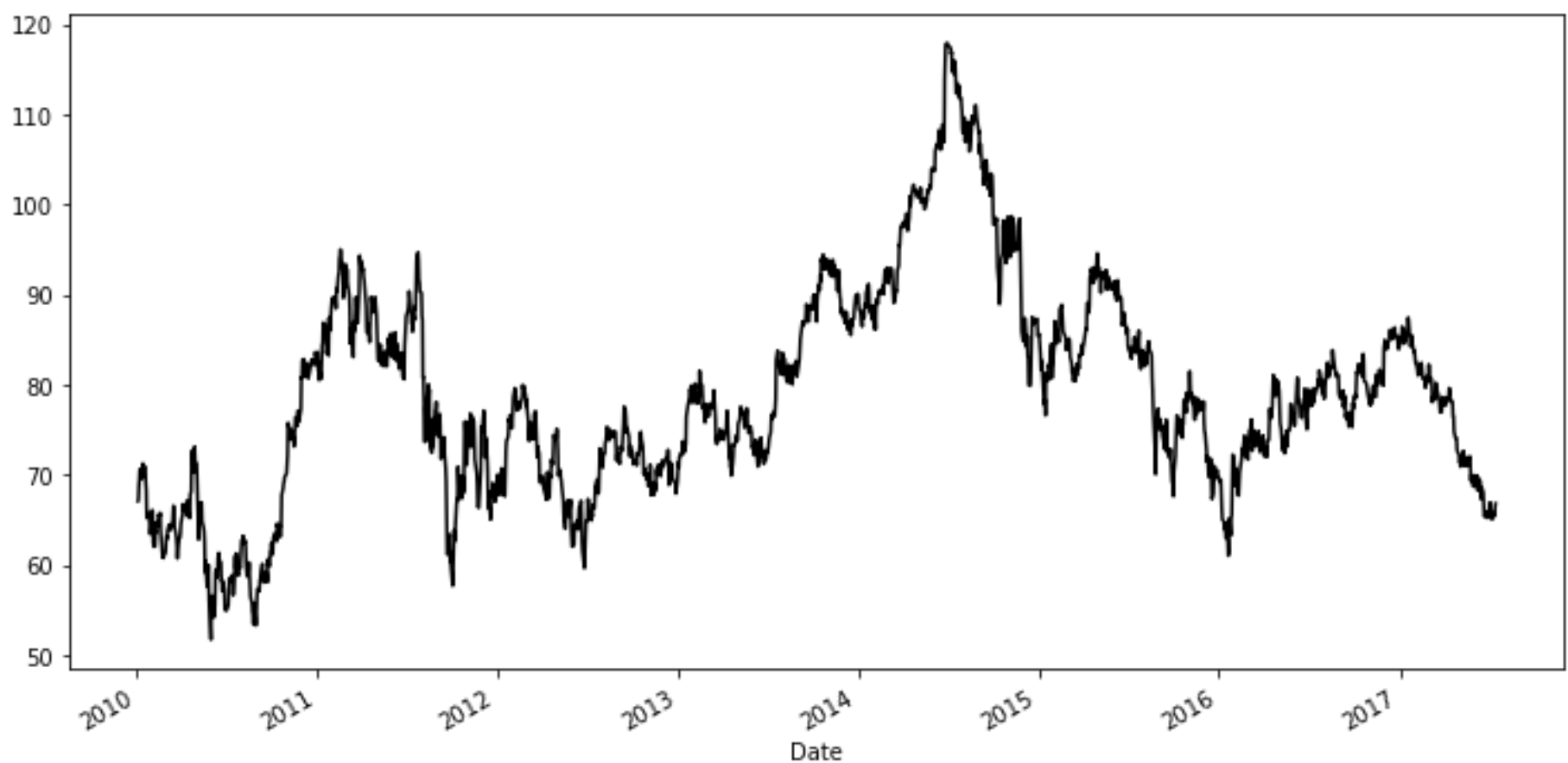
```
upper_10 = slb_summary.loc['90%']
lower_10 = slb_summary.loc['10%']
criteria = (slb_close < lower_10) | (slb_close > upper_10)
slb_top_bottom_10 = slb_close[criteria]
```

(4) フィルタリングしたSeriesを灰色で黒の終値の上にプロットする。10位および90位パーセンタイルの水平線を描く

```
In [30]: import matplotlib.pyplot as plt

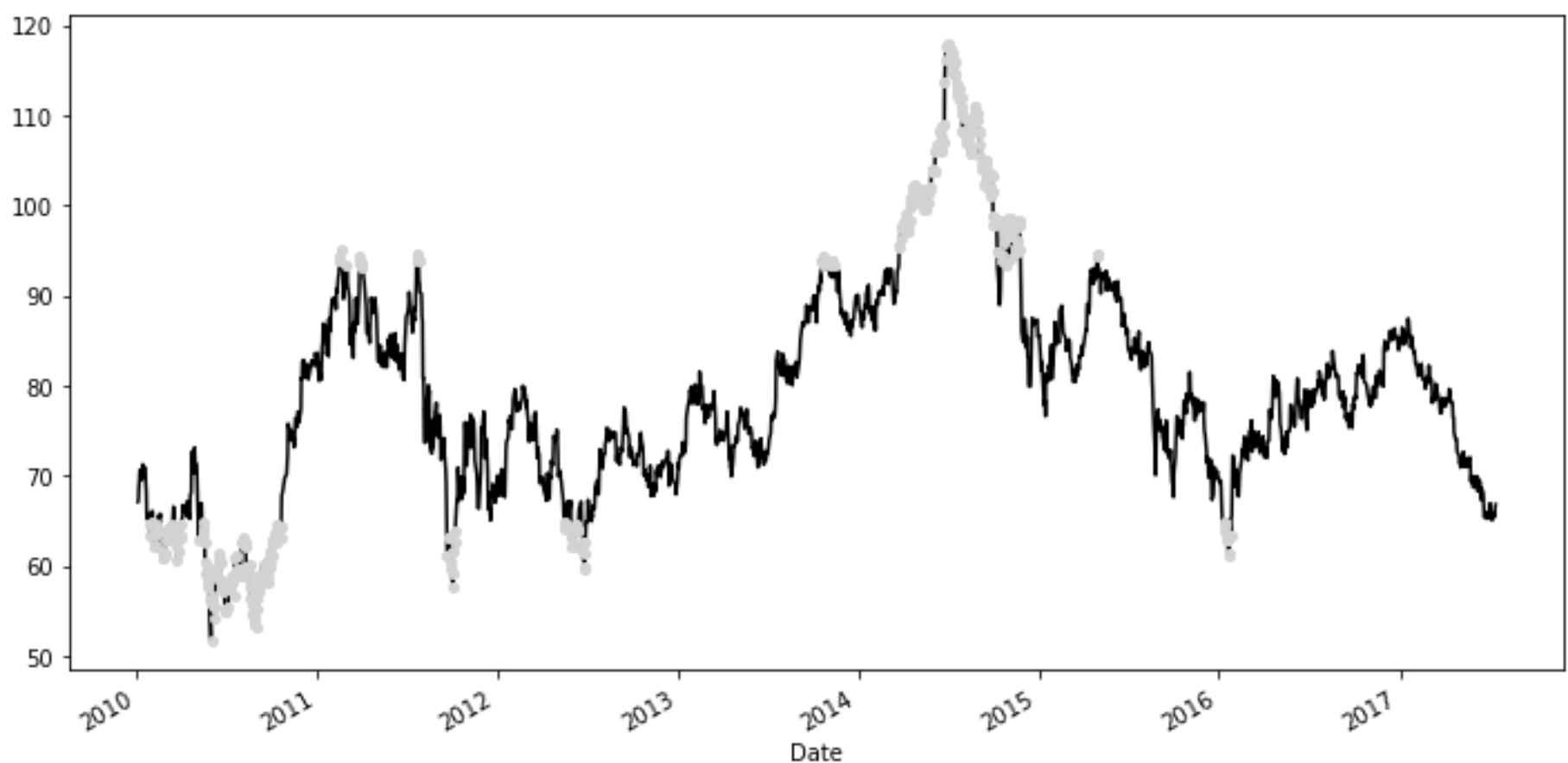
slb_close.plot(color='black', figsize=(12,6))
```

Out[30]: <AxesSubplot:xlabel='Date'>



```
In [34]: slb_close.plot(color='black', figsize=(12,6))
slb_top_bottom_10.plot(marker='o', style=' ', ms=4, color='lightgray') # style='空白'がポイント。ないと線でつながる
```

Out[34]: <AxesSubplot:xlabel='Date'>



```
In [36]: # cf
criteria
```

Out[36]: Date
2010-01-04 False
2010-01-05 False
2010-01-06 False
2010-01-07 False
2010-01-08 False
...

```
2017-07-10 False
2017-07-11 False
2017-07-12 False
2017-07-13 False
2017-07-14 False
Name: Close, Length: 1895, dtype: bool
```

In [39]:

```
# cf
print(criteria.index[0])
print(criteria.index[-1])
```

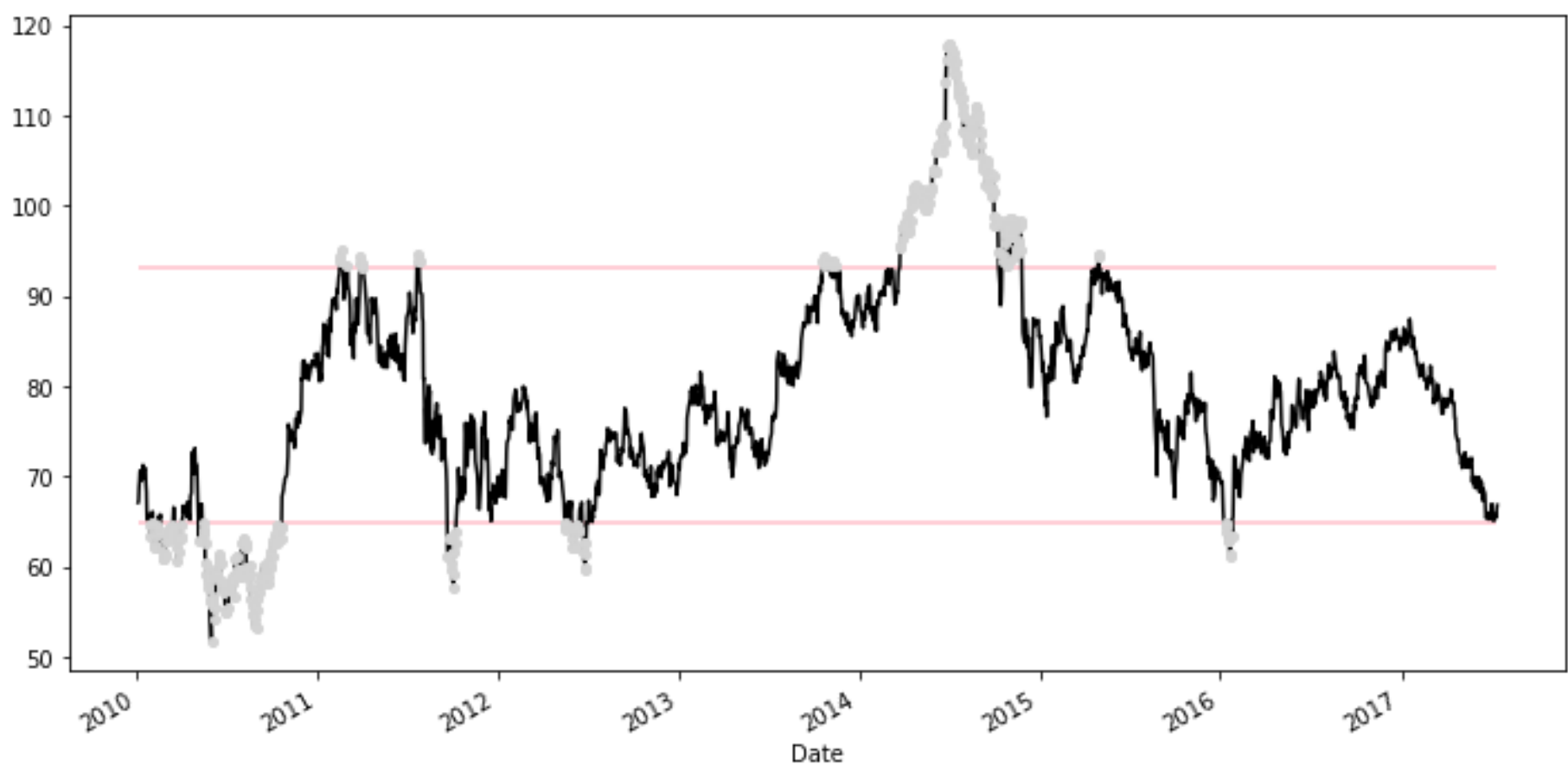
```
2010-01-04 00:00:00
2017-07-14 00:00:00
```

- `plt.hlines(y, xmin, xmax)` 水平線を引く
- `y`、`xmin`、`xmax`はデータ座標の値

In [44]:

```
slb_close.plot(color='black', figsize=(12,6))
# style='空白'がポイント。ないと線でつながる。msは取引高
slb_top_bottom_10.plot(marker='o', style=' ', ms=4, color='lightgray')
xmin = criteria.index[0]
xmax = criteria.index[-1]
plt.hlines(y=[lower_10, upper_10], xmin=xmin, xmax=xmax, color='pink')
```

Out[44]: <matplotlib.collections.LineCollection at 0x7ff32993d1c0>



(補足) 割愛