Statistical Outlier Detection Notes

Outlier Detection Using Mean and Standard Deviation (Z-Score Based Outlier Detection)

To detect outliers in a dataset Δ, we use the mean and standard deviation:

- $\mu(\Delta)$: Mean of the data
- $\sigma(\Delta)$: Standard deviation of the data

Normal Range

The normal range is defined as:

$$\mu(\Delta) \pm 2\sigma(\Delta)$$

This means most data points (about 95% if normally distributed) are expected to lie within this range.

Outlier Condition

A value is considered an outlier if:

$$\Delta < \mu(\Delta) - 2\sigma(\Delta) \quad \text{or} \quad \Delta > \mu(\Delta) + 2\sigma(\Delta)$$

- Δ - Orderbook Delta Depth of 5% from Coinbase (BTC/USD)
- $\mu(\Delta)$ - Mean of Δ
- $\sigma(\Delta)$ - Standard deviation of the dataset

Lookahead Bias

Strength of Signal

To not only detect outlier price points but also see how far they are expanded from the mean I use the Z-score for each data point being detected as an outliers

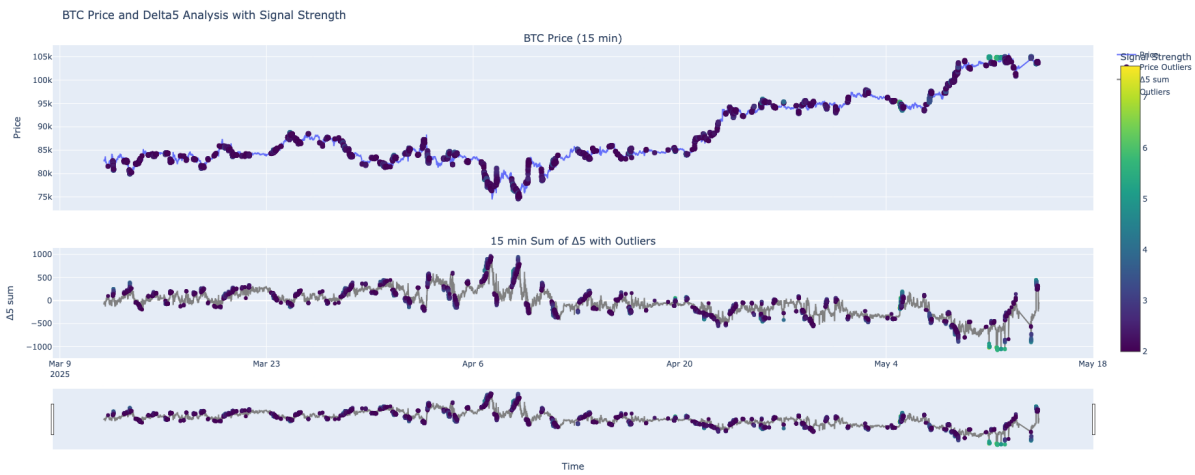

Figure 1: Heatmap Outlier

The Z-score is calculated as:

$$Z = \frac{\Delta_5 - \mu(\Delta_5)}{\sigma(\Delta_5)}$$

Only points outside the $[\mu - 2\sigma, \mu + 2\sigma]$ interval are considered outliers. For these, the signal strength is defined as $|Z|$, indicating how extreme the value is compared to the distribution.

Example: A point with a Z-score of +3.1 is a stronger signal than one at +2.1, since it is farther from the mean. Non-outliers receive a signal strength of 0. [1]

[1] Visualisation inside of Figure 1

Idea behind

- This method assumes data is roughly normally distributed.
- Using 2σ captures approximately 95% of data points under a normal distribution.
- You can adjust the multiplier (e.g., 3σ) for stricter or looser thresholds.

Future Plans

- Test on more data
- use rolling windows (e.g. 1 day or 1 week) for local context.
- Compare sensitivity with +- 1.5σ or +-2.5σ

Measuring Volatility After Price Outlier Detection

$$r_t = \frac{P_t - P_{t-1}}{P_{t-1}} \tag{1}$$

Dictionary of Terms

- P_t Asset price at time t.
- $r_t = \dfrac{P_t - P_{t-1}}{P_{t-1}}$ – 1-minute price return at time t.
- $\sigma_t^{(15)}$ – Realized volatility: the standard deviation of the next 15 one-minute returns,

$$\sigma_t^{(15)} = \sqrt{\frac{1}{14}\sum_{i=1}^{15}\big(r_{t+i} - \bar{r}_t\big)^2}, \quad \bar{r}_t = \frac{1}{15}\sum_{i=1}^{15} r_{t+i}$$

  aligned so that at time t it measures volatility over $t+1$ to $t+15$.

In Py code

```
import pandas as pd
df = pd.read_csv(file_path)
df.set_index('timestamp', inplace=True)
#Compute 1-min return of delta_5

df['r_t'] = df['price'].pct_change().fillna(0)



#compute rolling std of the future 15 min window

window = 15

#rolling on r_t, then shift forward so index t hold vol of t+1...t+15
df['future_vol_15] = (
    df['r_t']
    .rolling(window=window)
    .std()
    .shift(-window)
)
```

Combining Indicators

Here I visulised the swing points, the EMA spread and the 100 outliers with the highest Z-Score in the same plot.

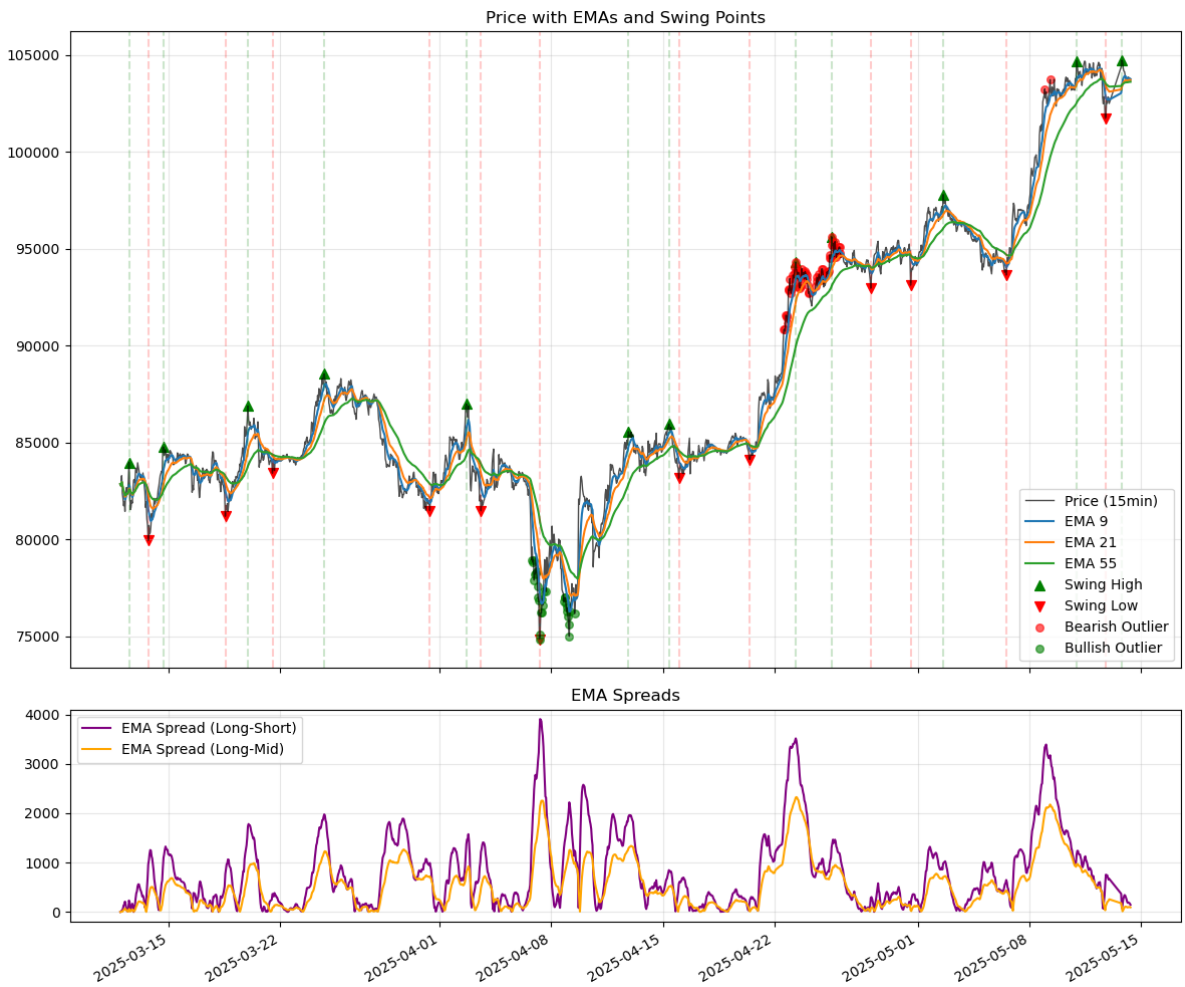

Figure 2: combined indicators png

[2]

[2]Chart made with Matplotlib and Seaborn