

PyBotTrader User Guide

Jaime Lopez

Version 0.0.8 - Jan. 2025

Contents

Introduction	1
Indicators	1
List of Indicators	3
Moving Average	3
Moving Variance	3
Exponential Moving Average	4
Return of Investment	4
Relative Strength Index	4
Moving Average Convergence/Divergence	4
Average True Range	4

Introduction

PyBotTrader is an experimental Python library designed to help on creating trading bots, intended for retail traders. It offers tools for real-time financial analysis, including indicators like moving averages (like MA, EMA, RSI, MACD, and ROI), which update dynamically with new data. The library includes data streamers to handle sequential data from sources like CSV files or the YFinance API, and basic portfolio managers for back-testing simple buy/sell strategies. Users can define custom strategies that integrate data streams, indicators, and decision-making rules to generate trading signals. A basic trader module is included for testing strategies, making the library a versatile framework for algorithmic trading experimentation.

Indicators

Indicators are aggregate measurements of financial data which intend is to reveal patterns and provide insights for decision making. In PyBotTrader, indicators are the most basic building block to implement trading algorithms.

There exist many libraries to compute indicators, like `ta-lib`, however PyBotTrader provides its own implementations designed for streaming data. Instead of making calculations from scratch at every moment, indicators in PyBotTrader keep memory of the previous results and get just updated when new data arrives. This is a more efficient approach for timed data.

All the available indicators in PyBotTrader share the same interface. Once they have been initialized, you can call the `update` method to compute new data. For example, one of the more basic indicators is the simple moving average, designated as `MA` in PyBotTrader. This represents the average of the `n` most recent data points.

```
from pybottrader.indicators import MA
```

```
ma = MA(period=3)
ma.update(1)
ma.update(2)
ma.update(3)
```

```
print(ma[0])  # Output is 2
```

```
ma.update(4)
```

```
print(ma[0])  # Output is 3
```

In the previous code examples, a moving average instance is created to represent the average of the last three data points. To access the value of the moving average, array notation is used. A zero index correspond to the current moment: `ma[0]`.

By default, indicators in PyBotTrader only keep memory of the most recent value. In the previous example, when a new data point is captured, the indicator value is recomputed and the previous value is forgotten. However, you are able to modify this behavior by defining how many values to be remembered assinging a value to the argument `mem_size` when an indicator is created.

```
from pybottrader.indicators import MA
```

```
ma = MA(period=3, mem_size=2)
ma.update(1)
ma.update(2)
ma.update(3)
ma.update(4)
ma.update(5)
```

```
print(ma[0])  # Output is 4
print(ma[-1]) # Output is 3
print(ma[-2]) # Error, invalid index
```

In the previous example, a moving average object is configured to remind two values, the current one is `ma[0]` and the previous one is `ma[-1]`. Observe that to access previous values, negative indices are used. Because the memory size is only for two values, trying to access `ma[-2]` produces an exception.

Using negative indices can seem strange for software developers, but it is a natural way to represent past events when the current moment is designed with the 0 index. Moreover, under this logic, positive indices can be used to represent future events.

Indicators in PyBotTrader are implemented in C++ instead of Python. In that way, your running code can take advantage of the speed and memory locality in your system.

At this moment, a limited number of indicators is implemented. In future versions, more indicators will be included.

List of Indicators

Moving Average

```
MA(int period, int mem_size = 1)
```

$$MA_n = \frac{1}{n} \sum_{i=1}^n x_i$$

$$MA_{t,n} = \frac{1}{n} \sum_{i=t-n+1}^t x_i$$

The moving average is used to determine the trend direction of a variable. It is calculated by adding up data points during a specific period divided by the number of time periods.

Moving Variance

```
MV(int period, int mem_size = 1)
```

$$MV_n = \frac{1}{n} \sum_{i=1}^n (x_i - MA_n)^2$$

$$MV_{t,n} = \frac{1}{n} \sum_{i=t-n+1}^t (x_i - MA_{t,n})^2$$

Exponential Moving Average

```
EMA(int period, double alpha = 2.0, int mem_size=1)
```

Return of Investment

```
ROI(int mem_size = 1)
```

Relative Strength Index

```
RSI(int period = 14, int mem_size = 1)
```

Moving Average Convergence/Divergence

```
MACD(int short_period, int long_period, int diff_period, int mem_size = 1)
```

Average True Range

```
“c++ ATR(int period, int mem_size = 1)
```