

# MetaTrader4or5 による自動売買

はじめに  
インジケータ  
進捗状況  
おわりに

大谷 和樹

富山県立大学 情報基盤工学講座

1. はじめに
2. インジケータ
3. 進捗状況
4. おわりに

## 背景

MetaTrader5 (MT5) と Python で、様々なインジケーターを利用して自動売買を行うシステムの開発を行っている。研究の背景や目的などについては、しっかりとした文にまとめることができていないため、今後まとめていく必要がある。

## 目的

自動売買で勝てるシステムを作る。

## 使用するライブラリ TA-Lib

### 動作確認

TA-Lib を使ったインジケータの計算が実際に行うことができるのかを確かめるために、まずは過去データを用いてインジケータの計算を行った。

今回は任天堂の過去データを使用した。

```
# 単純移動平均(SMA: Simple Moving Average)
output = np.c_[output, talib.SMA(close)]
cols += ['SMA']

# 加重移動平均(WMA: Weighted Moving Average)
output = np.c_[output, talib.WMA(close)]
cols += ['WMA']

# 指数移動平均(EMA: Exponential Moving Average)
output = np.c_[output, talib.EMA(close)]
cols += ['EMA']
```

Figure: TA-Lib のコード

はじめに

インジケータ

進捗状況

おわりに

はじめに  
インジケーター  
進捗状況  
おわりに

```
1 単純移動平均(SMA: Simple Moving Average)
2 加重移動平均(WMA: Weighted Moving Average)
3 指数移動平均(EMA: Exponential Moving Average)
4 2重指数移動平均(DEMA: Double Exponential Moving Average)
5 3重指数移動平均(TEMA: Triple Exponential Moving Average)
6 三角移動平均(TMA: Triangular Moving Average)
7 Kaufmanの適応型移動平均(KAMA: Kaufman Adaptive Moving Average)
8 MESAの適応型移動平均(MAMA: MESA Adaptive Moving Average)
9 トレンドライン(Hilbert Transform – Instantaneous Trendline)
10 ボリンジャー・バンド(Bollinger Bands)
11 MidPoint over period
12
13 変化率(ROC: Rate of change Percentage)
14 モメンタム(Momentum)
15 RSI: Relative Strength Index
16 MACD: Moving Average Convergence/Divergence
17 APO: Absolute Price Oscillator
18 PPO: Percentage Price Oscillator
19 CMO: Chande Momentum Oscillator
20
21 ヒルベルト変換 – Dominant Cycle Period
22 ヒルベルト変換 – Dominant Cycle Phase
23 ヒルベルト変換 – Phasor Components
24 ヒルベルト変換 – SineWave
25 ヒルベルト変換 – Trend vs Cycle Mode
26
27 60日単純移動平均
28 15日ボリンジャー・バンド
29 21日RSI
```

Figure: 計算したインジケータ

Figure: インジケータの計算結果

## 自動売買

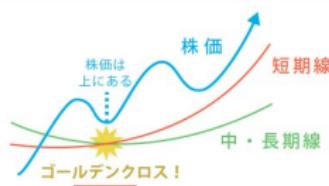
平松くんが作成していたプログラムの中で、売買ルールに関係している部分を自身が作ったものに書き換え、実際に動かした。  
今回は単純移動平均 (SMA) を用いたゴールデンクロス・デッドクロスで売買を決定している。

- Close のかわりに Tick を使用
- 計算に使用する期間は、短期 100・長期 200 に設定

## ゴールデンクロスとデッドクロス

ゴールデンクロスとは、短期の単純移動平均線が、長期の単純移動平均線を下から上に突き抜けるケースのこと。買いシグナルの1つ。デッドクロスは、逆に短期の単純移動平均線が、長期の単純移動平均線を上から下に突き抜けるケースのこと。売りシグナルの一つ。

ゴールデンクロス（買いシグナル）



デッドクロス（売りシグナル）



Figure: ゴールデンクロス

Figure: デッドクロス

# 実行結果

8/14

はじめに  
インジケーター  
進捗状況  
おわりに

2020.07.29 11:39...	usdjay	667678093	sell	0.1	104.861			2020.07.29 11:48:21	104.870	-90	-0.01 %
2020.07.29 11:59...	usdjay	667694211	sell	0.1	104.912			2020.07.29 12:19:37	104.916	-40	-0.00 %
2020.07.29 12:30...	usdjay	667721093	sell	0.1	104.910			2020.07.29 12:39:48	104.944	-340	-0.03 %
2020.07.29 12:43...	usdjay	667733078	sell	0.1	104.921			2020.07.29 12:52:27	104.913	80	0.01 %
2020.07.29 13:00...	usdjay	667747724	sell	0.1	104.930			2020.07.29 13:03:22	104.925	50	0.00 %
2020.07.29 13:07...	usdjay	667754121	sell	0.1	104.934			2020.07.29 13:16:29	104.966	-320	-0.03 %
2020.07.29 13:22...	usdjay	667768285	sell	0.1	104.933			2020.07.29 13:25:53	104.940	-70	-0.01 %
2020.07.29 13:27...	usdjay	667772102	sell	0.1	104.945			2020.07.29 13:47:47	104.998	-530	-0.05 %
2020.07.29 13:48...	usdjay	667792139	sell	0.1	104.989			2020.07.29 13:51:12	105.004	-150	-0.01 %
2020.07.29 13:53...	usdjay	667795674	sell	0.1	104.996			2020.07.29 14:04:42	104.992	40	0.00 %
2020.07.29 14:14...	usdjay	667819137	sell	0.1	105.019			2020.07.29 14:22:23	104.995	240	0.02 %
2020.07.29 14:29...	usdjay	667836351	sell	0.1	104.974			2020.07.29 14:33:55	104.979	-50	-0.00 %
2020.07.29 14:39...	usdjay	667845620	sell	0.1	104.915			2020.07.29 14:50:51	105.027	-920	-0.09 %
2020.07.29 15:13...	usdjay	66785674	sell	0.1	105.155			2020.07.29 15:24:41	105.107	480	0.05 %
2020.07.29 15:26...	usdjay	667901119	sell	0.1	105.087			2020.07.29 15:30:53	105.078	90	0.01 %
2020.07.29 15:41...	usdjay	667916174	sell	0.1	105.077			2020.07.29 16:08:02	105.037	400	0.04 %
2020.07.29 16:23...	usdjay	667967928	sell	0.1	104.976			2020.07.29 16:30:31	105.061	-850	-0.08 %
2020.07.29 16:37...	usdjay	667989406	#667967928, エキスパートによる注文					2020.07.29 16:43:35	105.062	-610	-0.06 %
2020.07.29 16:49...	usdjay	668006108	sell	0.1	105.022			2020.07.29 16:55:24	105.045	-230	-0.02 %
2020.07.29 16:56...	usdjay	668014184	sell	0.1	105.038			2020.07.29 17:00:08	105.006	320	0.03 %
2020.07.29 17:08...	usdjay	668031000	buy	0.1	104.964			2020.07.29 17:20:25	104.987	230	0.02 %
2020.07.29 17:45...	usdjay	668081224	buy	0.1	104.942			2020.07.29 18:10:25	105.071	1,290	0.12 %
2020.07.29 18:15...	usdjay	668129712	buy	0.1	105.085			2020.07.29 18:46:09	105.029	-560	-0.05 %
2020.07.29 19:00...	usdjay	668173792	buy	0.1	105.070			2020.07.29 19:03:16	105.055	-150	-0.01 %
2020.07.29 19:05...	usdjay	668178404	buy	0.1	105.038			2020.07.29 19:09:07	105.000	-380	-0.04 %

Figure: SMA を用いて自動売買した結果

残高 : 97,788JPY → 95,148JPY

結果 : -2640

はじめに  
インジケータ  
進捗状況  
おわりに

## 指摘事項 1

インジケータの計算をする際に使用するデータの値は一定の間隔ごとに取得しないといけない

## 修正 1

MT5 からとってきた tick データをどんどん取得していく中で、1 秒ごとにその時点で最新の tick データ resample をして ohlc の形で別のデータフレームに保存していくようにした。これにより、一定時間間隔の ohlc のデータフレームを作成することができた。秒足の部分に関しては指定できるので、任意の時間感覚で resample することができる。

# 前回からの修正1

10/14

はじめに  
インジケータ  
進捗状況  
おわりに

time					
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:40	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:40	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:41	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:40	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:41	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:42	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:40	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:41	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:42	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:43	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
<b>time</b>					
2020-10-01 11:57:39	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:40	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:41	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:42	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:43	105.519	105.519	105.519	105.519	105.519
2020-10-01 11:57:44	105.519	105.519	105.519	105.519	105.519

Figure: 1 秒足の ohlc データの取得結果

はじめに  
インジケーター  
進捗状況  
おわりに

## 指摘事項 2

データフレームに入れ続けることでデータが溢れてしまう

## 修正 2

データフレームの中身がある程度の大きさになったら古いものから削除していくようとする。実際には、どんどん取得されてくる tick データをデータフレーム 1 に格納していく、その中から 1 秒ごとに resample して別のデータフレーム 2 に格納している。データフレーム 1 の中身をどんどん削除していくばデータフレーム 2 も対応して減っていく。

# 前回からの修正1

12/14

はじめに  
インジケーター  
進捗状況  
おわりに

		open	high	low	close
time					
2020-10-01 10:42:24	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
time					
2020-10-01 10:42:24	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:25	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
time					
2020-10-01 10:42:24	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:25	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:26	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
time					
2020-10-01 10:42:25	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:26	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:27	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
time					
2020-10-01 10:42:25	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:26	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:27	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:28	105.519	105.519	105.519	105.519	105.519
	open	high	low	close	
time					
2020-10-01 10:42:26	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:27	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:28	105.519	105.519	105.519	105.519	105.519
2020-10-01 10:42:29	105.519	105.519	105.519	105.519	105.519

Figure: データフレームの中身を削除していく例

はじめに  
インジケーター  
進捗状況  
おわりに

## オリジナリティ

先生と相談した結果、マルチングールやココモ式、ポートフォリオ選択等のオペレーションズ・リサーチのシステムを組み込んでいく方針にした。この中でどれを組み込むのかはそれぞれ調査した結果決定する。

## 今回の進捗

- ① 一定時間間隔の ohlc のデータフレームを作成する
- ② データフレームの中身がある程度の大きさになつたら古いものから削除していくように変更
- ③ 上記 2 つを組み込んで動かしてみたら売買が行われなかつた & 日付変更時にエラーで止まつてしまつた

## 今後の課題

- ① 動かなかつた原因の調査
- ② 秒足の ohlc のデータをローソク足として描画
- ③ インジケータの最適化
- ④ オペレーションズ・リサーチに関して調査