

# MetaTrader5 と Python による自動売買

# 富山県立大学 情報基盤工学講座

## 1515010 大谷和樹

指導教員：奥原浩之

# 1 はじめに

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、しっかりととした文にまとめることができないため、今後まとめていく必要がある。

## 2 TA-Lib

前回の研究紹介をした際に、使用するインジケータの計算は Python で簡単にテクニカル指標が計算できる「pyti」というライブラリを使用することにしていた。しかし、うまく pyti のプログラムが動かなかったため、pyti と同じように Python でテクニカル指標が計算できる「TA-Lib」というライブラリを使用することにした。

```

# 简单移动平均(SMA: Simple Moving Average)
output = np.c_[output, talib.SMA(close)]
cols += ['SMA']
# 加权移动平均(WMA: Weighted Moving Average)
output = np.c_[output, talib.WMA(close)]
cols += ['WMA']
# 指数移动平均(EMA: Exponential Moving Average)
output = np.c_[output, talib.EMA(close)]
cols += ['EMA']
# 2重指数移动平均(DEMA: Double Exponential Moving Average)
output = np.c_[output, talib.DEMA(close)]
cols += ['DEMA']
# 3重指数移动平均(TEMA: Triple Exponential Moving Average)
output = np.c_[output, talib.T3(close)]
cols += ['TEMA']
# 三角移动平均(TMA: Triangular Moving Average)
output = np.c_[output, talib.TRIMA(close)]
cols += ['TMA']
# Kaufman适应性移动平均(KAMA: Kaufman Adaptive Moving Average)
output = np.c_[output, talib.KAMA(close)]
cols += ['KAMA']
# MESA高维适应性移动平均(MAMA: MESA Adaptive Moving Average)
for arr in talib.MAMA(close):
    output = np.c_[output, arr]
cols += ['MAMA', 'FAMA']
# ハーバートランジン(Hilbert Transform - Instantaneous Trendline)
output = np.c_[output, talib.HT_TRENDLINE(close)]
cols += ['HT TRENDLINE']
# ボリンジャー・バンド(Bollinger Bands)
for arr in talib.BBANDS(close):
    output = np.c_[output, arr]
cols += ['BBANDS_upperband', 'BBANDS_middleband', 'BBANDS_lowband']
# MIDPoint over period
output = np.c_[output, talib.MIDPOINT(close)]
cols += ['MIDPOINT']

```

図 1: TA-Lib でインジケータを計算するプログラム (一部)

上の図1からわかるように、TA-Libを使うこと1行のコードでインジケータの計算を行うことができる。

### 3 進捗状況

今回は、TA-Lib を使ったインジケーターの計算が実際に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ーの計算を行った。計算結果が以下の図 2 である。初めの方の空白部分は計算に使うための情報が揃っていないため計算が行われていなかった部分である。

ことができる事が確認できたため、今度はMT5を使って実際に自動売買をする事ができるか確認した。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。その際の売買の結果が以下の図3である。

2007/29/11~29	usd/jpy	667528951	sel	0.1	104.851	2007/29/11~29	104.851	104.870	+0.1	+0.1%
2007/29/11~29	usd/jpy	667528951	sel	0.1	104.851	2007/29/10~29	104.851	104.870	+0.1	+0.1%
2007/29/12~30	usd/jpy	667711093	sel	0.1	104.910	2007/29/12~30	104.910	104.944	+0.034	+0.3%
2007/29/12~30	usd/jpy	667711093	sel	0.1	104.910	2007/29/12~30	104.910	104.944	+0.034	+0.3%
2007/29/12~30	usd/jpy	667733078	sel	0.1	104.931	2007/29/12~30	104.931	104.913	-0.018	-0.1%
2007/29/12~30	usd/jpy	667747724	sel	0.1	104.930	2007/29/13~30	104.930	104.925	-0.005	-0.0%
2007/29/13~30	usd/jpy	667754121	sel	0.1	104.934	2007/29/13~30	104.934	104.966	+0.032	+0.3%
2007/29/13~30	usd/jpy	667768285	sel	0.1	104.933	2007/29/13~30	104.933	104.940	+0.007	+0.1%
2007/29/13~30	usd/jpy	667772102	sel	0.1	104.945	2007/29/13~30	104.945	104.998	+0.053	+0.5%
2007/29/13~30	usd/jpy	667772102	sel	0.1	104.945	2007/29/13~30	104.945	104.998	+0.053	+0.5%
2007/29/13~30	usd/jpy	667795674	sel	0.1	104.986	2007/29/13~30	104.986	104.992	+0.006	+0.1%
2007/29/13~30	usd/jpy	667795674	sel	0.1	104.986	2007/29/13~30	104.986	104.992	+0.006	+0.1%
2007/29/14~29	usd/jpy	667811317	sel	0.1	105.019	2007/29/14~29	105.019	104.995	-0.024	-0.2%
2007/29/14~29	usd/jpy	667836351	sel	0.1	104.974	2007/29/14~30	104.974	104.979	+0.005	+0.1%
2007/29/14~29	usd/jpy	667845420	sel	0.1	104.935	2007/29/14~30	104.935	105.027	+0.092	+0.9%
2007/29/15~15	usd/jpy	667865674	sel	0.1	105.115	2007/29/15~24	105.115	105.107	-0.008	-0.1%
2007/29/15~15	usd/jpy	667901119	sel	0.1	105.087	2007/29/15~30	105.087	105.078	-0.009	-0.1%
2007/29/15~15	usd/jpy	667901119	sel	0.1	105.087	2007/29/15~30	105.087	105.037	-0.050	-0.5%
2007/29/16~25	usd/jpy	667917945	sel	0.1	105.176	2007/29/16~25	105.176	105.161	-0.015	-0.1%
2007/29/16~25	usd/jpy	667917945	sel	0.1	105.176	2007/29/16~25	105.176	105.161	-0.015	-0.1%
2007/29/16~25	usd/jpy	667980495	sel	0.1	105.001	2007/29/16~25	105.001	105.062	+0.061	+0.6%
2007/29/16~25	usd/jpy	667980495	sel	0.1	105.001	2007/29/16~25	105.001	105.062	+0.061	+0.6%
2007/29/16~25	usd/jpy	668006108	sel	0.1	105.022	2007/29/16~25	105.022	105.045	+0.023	+0.2%
2007/29/16~25	usd/jpy	668006108	sel	0.1	105.022	2007/29/16~25	105.022	105.045	+0.023	+0.2%
2007/29/16~25	usd/jpy	668014184	sel	0.1	105.038	2007/29/17~20	105.006	105.020	+0.014	+0.2%
2007/29/17~20	usd/jpy	668014184	buy	0.1	104.964	2007/29/17~20	104.964	104.987	+0.023	+0.2%
2007/29/17~20	usd/jpy	668014184	buy	0.1	104.964	2007/29/17~20	104.964	104.987	+0.023	+0.2%
2007/29/17~20	usd/jpy	668081224	buy	0.1	104.942	2007/29/18~20	105.071	105.071	1.290	0.12%
2007/29/17~20	usd/jpy	668081224	buy	0.1	104.942	2007/29/18~20	105.071	105.029	-0.560	-0.5%
2007/29/18~06	usd/jpy	668179172	buy	0.1	105.085	2007/29/18~06	105.085	105.055	-0.150	-0.1%
2007/29/18~06	usd/jpy	668179172	buy	0.1	105.085	2007/29/18~06	105.085	105.090	+0.041	+0.1%
2007/29/18~06	usd/jpy	668179172	buy	0.1	105.085	2007/29/18~06	105.085	105.090	+0.041	+0.1%

図 3: SMA を用いた売買の結果

元々の平松くんのプログラムと同じようにSMA単体だけでは勝てないことが見てわかる。

## 4 おわりに

使用するライブラリを変更することにした。こちらのライブラリは実際に動くことが確認できたので、今後は使用するインジケータの選別とそれを用いた売買ルールの設定・動作確認を急ぐと共に、研究の肝である機械学習での売買ルールの選別方法を考える部分に取り掛かりたい。

参考文献

- [1] <https://qiita.com/ryoshi81/items/983c06e0bf859b280eba>  
Accessed: July 30, 2020.
  - [2] [https://www.mql5.com/ja/docs/integration/python\\_metatrader5/mt5copyratesrange\\_py](https://www.mql5.com/ja/docs/integration/python_metatrader5/mt5copyratesrange_py),  
Accessed: July 30, 2020.
  - [3] <https://tkstock.site/2019/11/21/python-quandl-get-stock-price-data/>  
Accessed: July 30, 2020.

図 2: 過去データを用いたインジケータの計算結果

これにより実際に TA-Lib を用いてインジケータの計算を行う