

К.Ю. Островська, Н.О. Кислова, О.О. Головацький
**ДОСЛІДЖЕННЯ ТЕХНІЧНИХ ІНДИКАТОРІВ
ДЛЯ ОПТИМАЛЬНОЇ СТРАТЕГІЇ БІРЖЕВОГО РИНКУ
З ВИКОРИСТАННЯМ PYTHON ТА БІБЛІОТЕКИ TA-LIB**

В результаті роботи, було розглянуто дві стратегії, які побудовані на найпопулярніших індикаторах технічного аналізу MACD - Moving Average Convergence / Divergence, RSI - осцилятори Relative Strength Index - «індекс відносної сили» та ЕМА - експоненціальне ковзне середнє. Результати показали, що обидві стратегії збиткові. У зв'язку з тим, що ринок крипто-валют має велику волатильність.

Ключові слова: технічні індикатори, аналіз, крипто-валюта, стратегія, біржовий ринок, мова, бібліотека, python, ta-lib.

З розвитком комп'ютерної техніки та систем електронної торгівлі все більшого поширення серед трейдерів ставав один з класичних підходів до здійснення фондових операцій - технічний аналіз. Даний підхід відноситься до математичних методів прогнозування ринку, його методи ґрунтуються на виявленні ринкових тенденцій за рахунок аналізу цін і обсягів торгів попередніх періодів. Завдяки тому що технічний аналіз зародився в той час, коли комп'ютерів ще не існувало, більшість його методів є досить нескладними і інтуїтивно зрозумілими, що також сприяло їх поширенню. Незважаючи на численні переваги і широке поширення методів технічного аналізу, питання про їх ефективності залишається відкритим.

Серед інструментів технічного аналізу існують як методи, так і індикатори. З практики ринкової торгівлі відомо, що індикатори набули більшого поширення, ніж методи, головним чином, по причини більшої простоти і зручності використання [1].

Технічними індикатори названі тому, що вони використовують лише статистичні показники торгів (ринків) і не беруть до уваги фундаментальні показники торгованих інструментів, наприклад, такі як виручка і прибуток компаній, чиї цінні папери торгуються на фондовому ринку.

Треjder в роботі використовує різні інструменти аналізу, що дозволяють підвищити ефективність операцій на ринку і збільшити ймовірність успіху. Одними з таких інструментів є технічні індикатори, отримані шляхом здійснення ряду математичних дій.

Дослідження технічних індикаторів:

ЕМА (Exponential Moving Average) - експоненціальне ковзне середнє. Особливість інструменту - більш складна побудова, що дозволяє усунути дві проблеми:

- ЕМА надає великої ваги параметрам, які мають місце в останні кілька днів. Отже, даний показник відноситься до категорії зважених;

- Виникає шанс застосування всіх даних цін за весь термін дії ринку активу.

Обчислення робиться за формулою:

$$EMA_t = EMA_{t-1} + (k \times (P_t - EMA_{t-1})), \quad (1)$$

де t – сьогоднішня, $t-1$ - вчорашній день, $k = 2 / (n + 1)$, де n порядок ковзаючого середнього (КС).

Динамічний індикатор MACD (Moving Average Convergence / Divergence) найчастіше відносять до категорії трендових, але він може грати роль і осцилятора. За даним параметру можна судити про співвідношення між парою КС ціни. Побудова індикатора проводиться з урахуванням різниці між двома ЕМА, що мають періоди в 12 і 26 днів (параметр задається за замовчуванням). Для більш точного нанесення кращих місць для угоди (продажу або покупки) на графік Moving Average Convergence / Divergence може наноситися додаткова (сигнальна) лінія з дев'яти експоненційних ковзають середніх від MACD-Line при обов'язковому згладжування (параметр за замовчуванням - 9).

Обчислення гістограми відбувається за іншим принципом - з ЕМА з більш низьким порядком (12) віднімається ЕСС більшого порядку (26). Після з отриманого параметра повинна бути вирахована 9-денна ЕМА від різниці ЕМА з порядок 12 і цим же експоненціальним ковзним середнім, але з порядком 26:

$$MACD = EMA(P, 12) - EMA(P, 26) - EMA[(EMA(P, 12) - EMA(P, 26))9], \quad (2)$$

де P - ціна, $EMA(P, n)$ - ЕСС порядку n .

Ефективність і простота методики двох ковзних середніх зробила її по-справжньому популярною в середовищі аналітиків. При цьому кращі результати MADC відображає при її аналізі на невеликих часових проміжках (від одного дня і більше). З більшою обережністю потрібно проводити аналіз Moving Average Convergence / Divergence на періоді менше одного дня. Що стосується часових і більш «коротких» періодів, то вони хоч і інформативні, але криють в собі масу неправдивих сигналів.

Ефективність MADC проявляється на волатильних ринках, коли коливання ціни здійснюється в більшому діапазоні.

Один з найпопулярніших осциляторів у аналітиків і трейдерів - Relative Strength Index (RSI) - «індекс відносної сили». Його параметр коливається в межах від 0 до 100. Один з найпопулярніших методів аналізу осцилятора - знаходження розбіжностей, при яких буде створюватися новий максимум. У свою чергу, RSI вже не здатний перейти рівень минулого максимуму.

Така розбіжність говорить про ймовірні розбіжності цін. Якщо RSI спочатку прямує вниз і опускається нижче рівня западини, то відбувається «фіналізація» неуспішного розмаху. В даному випадку можна говорити про майбутнє розвороті ціни.

Розрахунок RSI Relative Strength Index проводиться так:

$$RSI = 100 - [100 / (1 + RS)]; \quad (3)$$
$$RS = AU_x / AD_x,$$

де x – кількість днів; AU - середнє значення, що закрились вище попередніх цін за x днів; AD - середнє значення, що закрились нижче попередніх цін за x днів.

За допомогою осцилятора можна обчислити реальну силу настроїв биків і ведмедів на ринку в певний термін за рахунок відстеження цін закриття протягом вже наступного періоду. Параметр RSI Relative Strength Index має свій діапазон - від нуля до ста.

У порівнянні з іншими індикаторами, Relative Strength Index відрізняється найбільшою точністю. Причина - здатність індикатора випереджати поточний рух ціни, даючи трейдеру можливість передбачити ринок і зробити вигідну угоду. Наприклад, лінія осцилятора може помінятися за 1-2 діб до моменту, як це зробить робити поточна ціна [2].

Для роботи нам буде потрібно бібліотека для наукових розрахунків `numpy`, бібліотека `TA-Lib` і модуль `requests`. Так само знадобиться пакет `matplotlib`, з його допомогою ми будемо реалізовувати графіки [3].

Для початку нам необхідно підключитися до торговому майданчику про отримати історію торгів. В якості торгового майданчика будемо використовувати одну з найпопулярніших бірж з торгівлі крипто-валютами `Binance`.

```
import numpy as np
from talib import MA, RSI, MACD
import matplotlib.pyplot as plt
import ccxt
from pandas import DataFrame, to_datetime
from qtpylib_indicators import crossed_above, crossed_below
def main():
    exchange = ccxt.binance()
    ticker = exchange.fetch_ohlcv(symbol = 'XRP/BTC', timeframe = '30m')
if __name__ == '__main__':
    main()
```

Розглянемо код вище на самому початку ми підключили необхідні нам модулі потім створили об'єкт `exchange` який представляє торгову площадку. І далі необхідно запросити історію торгів з допомогою методу `fetch_ohlcv` в якості параметрів передали пару 'XRP / BTC' (Ripple / Bitcoin) часовий проміжок даних 30 хвилин.

Далі необхідно написати метод, який розбере історію торгів.

```
def parse_ticker_dataframe(ticker):
    columns = ['date', 'open', 'high', 'low', 'close', 'volume']
    frame = DataFrame(ticker, columns=columns)
    frame['date'] = to_datetime(frame['date'],
                               unit='ms',
                               utc=True,
                               infer_datetime_format=True)
    frame = frame.groupby(by='date', as_index=False, sort=True).agg({
        'open': 'first',
        'high': 'max',
        'low': 'min',
        'close': 'last',
        'volume': 'max',
    })
    frame.drop(frame.tail(1).index, inplace=True) # eliminate partial
    candle
    return frame
```

```
def main():
    exchange = ccxt.binance()
    ticker = exchange.fetch_ohlcv(symbol = 'XRP/BTC', timeframe = '30m')
    dataframe = parse_ticker_dataframe(ticker)
    print(dataframe.iloc[:5])
```

Перевіримо результат:

	date	high	close	open	low	volume
494	2018-12-07 15:30:00	0.000089	0.000089	0.000089	0.000088	1346554.0
495	2018-12-07 16:00:00	0.000089	0.000089	0.000089	0.000089	2208911.0
496	2018-12-07 16:30:00	0.000090	0.000089	0.000089	0.000089	2495455.0
497	2018-12-07 17:00:00	0.000089	0.000089	0.000089	0.000089	1512177.0
498	2018-12-07 17:30:00	0.000089	0.000089	0.000089	0.000089	1832909.0

За замовчуванням метод `fetch_ohlcv` повертає інформацію для 500 свічок. Тепер у нас є більш наочна історія торгів розглянемо останній рядок, тож ми бачимо, що о 17:30 7 грудня 2018 року сесія була відкрита ціною 0,000089 біткоїн за одну Манету закрилась ціною 0,000089 найвища ціна за сесію 0,000089 і найнижча ціна 0,000089 і обсяг продажів 1832909 монет. На рисунку 1, відображено ціну закриття.

Далі зробимо розрахунок індикатора MACD і додамо його в наш графік (див. рисунок 2):

```
def applyMACD(dataframe):
    macd, macdsignal, macdhist = MACD(dataframe['close'].values, fastperiod=12, slowperiod=26, signalperiod=9)
    dataframe['macd'] = macd
    dataframe['macdsignal'] = macdsignal
    dataframe['macdhist'] = macdhist
    return dataframe

def main():
    # Тут пропущений код який був представлений раніше
    dataframe = applyMACD(dataframe)
    #print MACD
    ax1.plot(dataframe.index, dataframe['macd'], label='MACD',
            color='blue')
    ax1.plot(dataframe.index, dataframe['macdsignal'], label='MACD Signal',
            color='orange')
    ax1.plot(dataframe.index, dataframe['macdhist'], label='MACD Signal',
            color='red')
    ax1.fill_between(dataframe.index, y1=0, y2=dataframe['macdhist'],
            color='red', alpha='0.3')
    ax1.xaxis.set_major_formatter(formatter)
    ax1.set_xlabel('Date')
    ax1.set_ylabel('Price')
```

`ax1.grid()`



Рисунок 1 - Графік цін закриття сесії

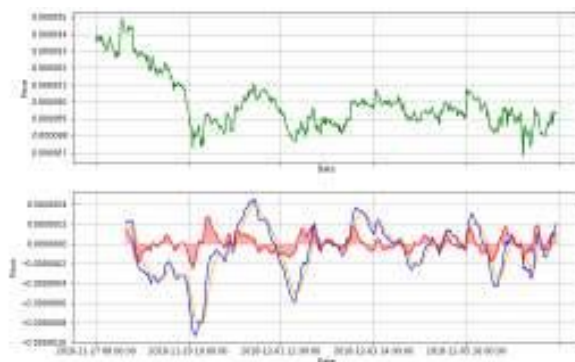


Рисунок 2 - Графік цін закриття сесії та індикатор MACD

Отже ми бачимо на нижньому графіку (див. рисунок 2) представлений індикатор MACD, де синя лінія - це лінія MACD, жовта лінія - це сигнальна лінія MACD і червона лінія - це гістограма MACD.

Далі додамо індикатор RSI (див. рисунок 3).

Отже на останньому графіку (див. рисунок 3) у нас представлений індикатор RSI за допомогою якого можна зробити аналіз ринку на перекупленність (вже не продається) або переподанність (вже не покупается). Синя напівпрозора зона - це зона, коли ринок знаходиться в стабільному стані.

Стану "перекупленність" і "перепроданність" як правило визначається індикаторами технічного аналізу (в основному осциляторами), такими як MACD, Momentum, Stochastic, RSI і іншими.

Далі додамо 2 останніх індикатора: це ЕМА з двома часовими періодами (див. рисунок 4).

```
def applyEMA(dataframe, timeperiod):
    dataframe['ema{0}'.format(timeperiod)] = EMA(dataframe['close'].values,
timeperiod)
    return dataframe
def main():
    #Здесь пропущен код который был представлен ранее
    dataframe = applyEMA(dataframe, 12)
    dataframe = applyEMA(dataframe, 26)
```

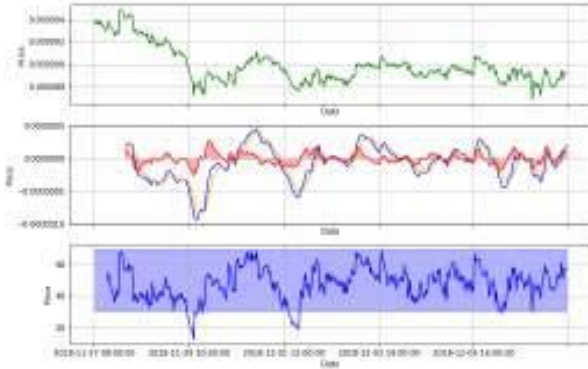


Рисунок 3 - Графік цін закриття сесії та індикатори MACD і RSI

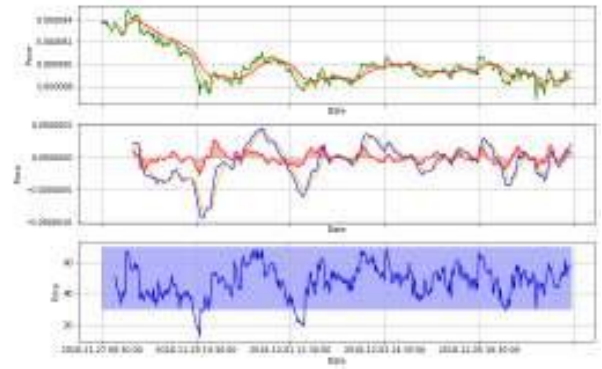


Рисунок 4 - Графік цін закриття сесії та індикатори MACD, RSI, ЕМА з двома часовими періодами

Отже на верхньому графіку (див. рисунок 4) у нас відображені два графіка технічного індикатору ЕМА. Жовта крива лінія - це ЕМА з тимчасовим періодом 12 і червона крива - це ЕМА з тимчасовим періодом 26.

Отже обрисуюмо стратегію №1.

Купівля буде здійснюватися за таких умов:

- 1) 12 - періодна ЕМА перетинає 26 - періодну ЕМА знизу.
- 2) MACD переходить вище нульової лінії.

Продаж буде здійснюватися за таких умов:

- 1) коли 12 - періодна ЕМА йде нижче 26 - періодної ЕМА.
- 2) MACD відправляється нижче нульової лінії.

Для початку реалізуємо метод, який буде визначати перетин ліній:

```
def crossed(series1, series2, direction=None):
    if isinstance(series1, np.ndarray):
        series1 = Series(series1)
    if isinstance(series2, (float, int, np.ndarray)):
        series2 = Series(index=series1.index, data=series2)
    if direction is None or direction == "above":
        above = Series((series1 > series2) & (
            series1.shift(1) <= series2.shift(1)))
    if direction is None or direction == "below":
        below = Series((series1 < series2) & (
            series1.shift(1) >= series2.shift(1)))
    if direction is None:
        return above or below
    return above if direction == "above" else below
```

```
def crossed_above(series1, series2):
    return crossed(series1, series2, "above")
def crossed_below(series1, series2):
    return crossed(series1, series2, "below")
```

Точки на покупку:

```
dataframe.loc [((dataframe ['macd'] > 0) &
                crossed_above (dataframe ['ema12'], dataframe
['ema26']))], 'buy'] = 1
```

Точки продажі:

```
dataframe.loc[ ((dataframe['macd'] < 0) &
                crossed_below(dataframe['ema12'], dataframe['ema26'])),
'sell'] = 1
```

Розставимо крапки покупки та продажу на графіку цін закриття сесії (див. рисунок 5):

```
for index, row in dataframe.iterrows ():
    if row ['buy'] == 1:
        ax0.scatter (x = index, y = dataframe.loc [index, 'close'],
color = 'blue', zorder = 3)
    elif row ['sell'] == 1:
        ax0.scatter (x = index, y = dataframe.loc [index, 'close'],
color = 'red', zorder = 3)
plot.tight_layout ()
```

В результаті отримали графік цін закриття сесії з позначками покупки та продажу криптовалюти (див. рисунок 5).

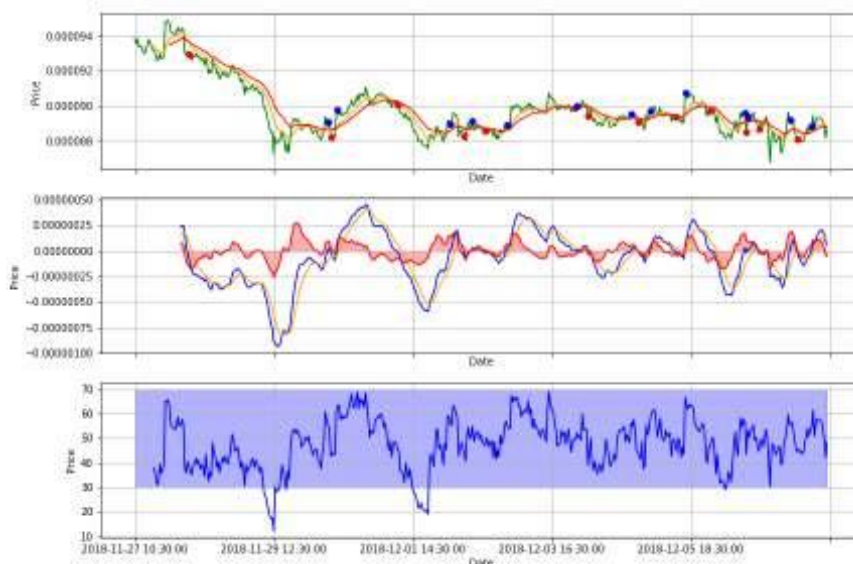


Рисунок 5 - Графік цін закриття сесії та індикатори MACD, RSI, ЕМА з двома часовими періодами з позначками покупки та продажу криптовалюти. Стратегія №1

В таблиці 1 представлені результати підрахунку прибутку по стратегії №1.

Таблиця 1

Результати підрахунку прибутку по стратегії № 1

Покупка	Продаж	Прибуток
0.0000891	0.00008826	-0.00000084
0.00008979	0.00009008	0.00000029
0.000089	0.0000883	-0.00000070
0.00008913	0.00008857	-0.00000056
0.00008895	0.00008989	0.00000094
0.00009	0.00008945	-0.00000055
0.00008957	0.00008912	-0.00000045
0.00008975	0.00008936	-0.00000039
0.00009074	0.00008974	-0.00000100
0.0000896	0.00008854	-0.00000106
0.00008941	0.00008871	-0.00000070
0.00008924	0.00008813	-0.00000111

Перевіримо стратегію №2.

Купівля буде здійснюватися за таких умов:

1) RSI знаходиться в зоні перепроданості (менше 30).

Продаж буде здійснюватися за таких умов:

1) коли 12 – періодна ЕМА йде нижче 26 – періодної ЕМА.

2) MACD відправляється нижче нульової лінії.

Отже реалізуємо нашу стратегію:

```
dataframe.loc[((dataframe['rsi'] <= 30)), 'buy'] = 1
dataframe.loc[((dataframe['macd'] < 0) &
                crossed_below(dataframe['ema12'], dataframe['ema26'])),
'sell'] = 1
```

В таблиці 2.2 представлені результати підрахунку прибутку по стратегії №2.

Таблиця 2

Результати підрахунку прибутку по стратегії № 2

Покупка	Продажа	Прибуток
0.00008992	0.00008826	-0.00000166
0.00008875	0.0000883	-0.00000045
0.00008817	0.00008854	0.00000037



Рисунок 6 - Графік цін закриття сесії та індикатори MACD, RSI, ЕМА з двома часовими періодами з позначками покупки та продажу криптовалюти. Стратегія №2

В результаті проведеної роботи, було розглянуто дві стратегії, які побудовані на найпопулярніших індикаторах технічного аналізу MACD, RSI, ЕМА. Результати показали, що обидві стратегії збиткові. У зв'язку з тим, що ринок крипто-валют має велику волатильність.

ЛІТЕРАТУРА

1. Кац, Джеффри Оуэн, МакКормик, Донна Л. Энциклопедия торговых стратегий / Пер. с англ. – М.: Альпина Пабlishер, 2002.–400с.
2. Швагер Д. Технический анализ, полный курс. - М.: Альпина Пабlishер, 2001. - 768с.
3. Колби, Роберт. Энциклопедия технических индикаторов рынка. - М.: Альпина Пабlishер, 2011. - 840 с.
4. Марк Лутц. Программирование на Python (4-е издание). - М.: Символ-Плюс», 2011. - 992 с.