

TechAnalysis

Cały zestaw modułów, klas i funkcji, które odpowiadają za analizę techniczną.

Wymagania systemowe :

- Zainstalowany Python 2.7.2
- Dołączona biblioteka pyNum w wersji 1.6.1

Moduł trendAnalysis

Moduł do analizy technicznej, zawiera funkcje wykrywające trend, oraz wszystkie formacje, które bazują na trendzie, czyli:

- formacja głowy i ramion,
- formacja odwróconej głowy i ramion
- formacja potrójnego szczytu/dna
- formacja rogu
- formacje geometryczne (trójkąty, prostokąty, kliny)
- flagi
- chorągiewki.

Ponadto wszystkie funkcje do znajdowania formacji zwracają moc sygnału w zależności od wielu czynników jak czas kształtowania się formacji, ilość cech charakterystycznych, zgodność z modelem książkowym czy odległość w czasie.

Atrybuty:

rectVul = 0.03 - o ile procent mogą odbiegać od siebie szczytowe wartości lewego i prawego ramienia w formacji głowy i ramion (HS)

hsVul = 0.1 - na ile podzbiorów dzielimy tablice przy wyznaczaniu linii kanału

div = 8 - na ile podzbiorów dzielimy tablice przy wyznaczaniu formacji HS

hsDiv = 12 - ile razy robimy przesunięcie przy wyznaczaniu formacji HS w celu uniknięcia pechowego podziału

dateVul = 6 - jaka ma być minimalna różnica procentowa między szczytem w głowie a ramionami, formacja HS

hsDiff = 0.03 - jaka ma być maksymalna różnica procentowa między kolejnymi szczytami w formacjach potrójnego szczytu i dna (triple)

tripleDiff = 0.03 - na ile podzbiorów dzielimy tablice przy wyznaczaniu formacji potrójnego szczytu

tripleDiv = 12 - jak długi może być okres formowania się formacji V w stosunku do badanego okresu

hornVul = 0.1 - jak bardzo lewy wierzchołek formacji rogu (horn) może odbiegać od prawego

hornDiv = 0.1 - ile razy wzrost procentowy na formacji przekracza średni wzrost

hornDiff = 5 - ile jednostek czasu trwa formowanie się podstawy

flagBaseTime = 3 – minimalny okres dla flag

formVul = 5 - Jest to coś jak trendVul, tylko dla formacji. Tzn. jest to wartość nachylenia prostej w stopniach, aby uznać linię za nie-poziomą.

convergenceVul = 15 - Miara zbieżności prostych, tzn różnica kątów nachylenia, powyżej której traktujemy linie jako nie-równoległe przy badaniu formacji.

Funkcje dla użytkownika:

getChannellines(array, a=3, b=4):

Zwraca linie wsparcia i oporu w postaci tablicy `[[x0, index_of_x0], [x1, index_of_x1], ...]`, argumenty:

a, b - jaką część tablicy wziąć, np. a = 1, b = 1 - cała tablica, a = 1, b = 2, druga połowa wejściowej tablicy

Output:

- tablice sup, res, które są zawierają punkty wyznaczające odpowiednio linie wsparcia i oporu

lookForHeadAndShoulders(values, volume, analyze=0):

Szukamy formacji głowy i ramion

Argumenty:

values – tablica wartości

volume – tablica wolumenu

analyze – czy jest używana w analizie, jeśli tak, zwraca tylko moc sygnału 0 – 1

Output: tablica: [moc sygnału, [punkty tworzące linie szyi]]

lookForReversedHeadAndShoulders(values, volume, analyze=0):

Szukamy odwróconej formacji głowy i ramion

Argumenty:

values – tablica wartości

volume – tablica wolumenu

analyze – czy jest używana w analizie, jeśli tak, zwraca tylko moc sygnału 0 - 1

Output: tablica: [moc sygnału, [punkty tworzące linie szyi]]

lookForTripleTop(values, volume, analyze=0):

Szukamy formacji potrójnego szczytu:

Argumenty:

values – tablica wartości

volume – tablica wolumenu

analyze – czy jest używana w analizie, jeśli tak, zwraca tylko moc sygnału 0 - 1

Output: tablica: [moc sygnału, [punkty tworzące linie szyi]]

lookForTripleBottom(values, volume, analyze=0):

Szukamy formacji potrójnego dna:

Argumenty:

values – tablica wartości

volume – tablica wolumenu

analyze – czy jest używana w analizie, jeśli tak, zwraca tylko moc sygnału 0 - 1

Output: tablica: [moc sygnału, [punkty tworzące linie szyi]]

hornTops(values, volume):

Szukamy formacji rogu wyższego:

Argumenty:

values – tablica wartości

volume – tablica wolumenu

Output: moc sygnału

hornBottoms(values, volume):

Szukamy formacji rogu niższego:

Argumenty:

values – tablica wartości

volume – tablica wolumenu

Output: moc sygnału

findGeometricFormations(values) - znajduje formacje geometryczne (trójkąty, kliny, prostokąty) na danej tablicy i przypisuje im wartości. Wewnątrz funkcji wywoływane jest *findGeometricFormationsOnFragment* dla całej tablicy, ostatnich 3/4, ostatniej połowy i ostatniej ćwiartki. Wartość formacji zależy od jej wielkości, tzn. formacja znaleziona na całej tablicy jest warta więcej niż formacja znaleziona na ćwiartce.

values – tablica, na której szukamy formacji (kursy zamknięć)

zwracana wartość: lista, której elementy to wartości zwracane przez *findGeometricFormationsOnArray*

findGeometricFormationsOnFragment(values,a,b) – znajduje formację geometryczną na podtablicy values określonej za pomocą parametrów a i b. Odbywa się to poprzez wywołanie *getChannelLines* z odpowiednimi parametrami, a następnie sprawdzeniu jakie kąty tworzą ze sobą proste wsparcia i oporu.

Prostokąt: linie wsparcia i oporu poziome

Trójkąt niższy: linia oporu malejąca, wsparcia pozioma

Trójkąt wyższy: linia oporu pozioma, wsparcia rosnąca

Trójkąt symetryczny: linia oporu malejąca, wsparcia rosnąca

Klin niższy: linia wsparcia malejąca, oporu jeszcze szybciej malejąca

Klin niższy: linia oporu rosnąca, wsparcia jeszcze szybciej rosnąca

values – j. w.

a, b – patrz *getChannelLines*

zwracana wartość: (nazwa,
(resx0,resy0,resx1,resy1),(supx0,supy0,supx1,supy1),wartość)

gdzie nazwa='rising_wedge', 'falling_wedge', 'rising_triangle', 'falling_triangle',
'symmetric_triangle', 'rect'

XXXx0,XXXx1,XXXy0,XXXy1 – współrzędne punktów wyznaczających proste
wsparcia/oporu na wykresie

wartość = 1 (modyfikowana w findGeometricFormations)