



## Variable Index Dynamic Average VIDYA

KivancOzBilgic VIDYA Tem 22, 2020



Moving Averages Volatility

27 2874

Chande's Variable Index Dynamic Average ( VIDYA ) is similar to the Exponential Moving Average ( EMA ), but automatically adjusts the smoothing weight based on price volatility .

VIDYA was developed by Tushar Chande and presented in [Technical Analysis](#) of Stocks & [Commodities](#) magazine in March 1992.

In the first version, a standard deviation was used as the [Volatility Index](#)  
Chande modified VIDYA in October 1995 to use a new [Chande Momentum Oscillator \(CMO\)](#) as the [Volatility Index](#).

So I decided to add an option for the users to change the calculation method.  
Default calculation is by using Chande's Momentum Oscillator  
Users can change the calculation method to Standart Deviation by unchecking the box on the settings menu.

Another option is to fix the CMO length parameter to 9 to make VIDYA more sensitive to price movements.  
CMO parameter is defaultly fixed to 9 but users can change the length of the parameter to the VIDYA length by unchecking the relevant box.

As a moving average, VIDYA smooths the market noises and shows the market trends more clearly.

To achieve the goals, the indicator filters out the market fluctuations (noises) by averaging the price values of the periods, over which it is calculated. In the process, some extra value (weight) is added to the average prices, as it is done during calculations of all weighted indicators, such as [EMA](#) , [LWMA](#) , and [SMA](#) . But during the VIDYA indicator's calculation, every period's price receives a weight increment adapted to the current market's volatility .

Note that the value of the smoothing factor (k) is calculated with the help of the period's [EMA](#) , and increasing or decreasing of the value is achieved by using [CMO Chande Momentum Oscillator](#) as a measure of the market's [volatility](#) . As a result, the Indicator slows down and does not react to the market's [volatility](#) when it increases and, on the contrary, speeds up when a strong steady trend takes place.

Note:  
Alarm added for color changes.

Hope you use this one at profitable trades.

Twitter: @kivancozbilgic

YouTube (Turkish): <http://youtube.com/c/kivancozbilgic>

YouTube (English): <https://www.youtube.com/channel/UCJWtINVEwUceyGt45UpB2-Q>



Twitter



Youtube



Instagram

### Açık kaynak kodlu komut dosyası

Gerek TradingView ruhuyla, bu beğenilen yazarı, yatırımcının anlayabilmesi ve doğrulayabilmesi için onu açık kaynak olarak yayınladı. Yazarın eline sağlık! Buru ücretsiz olarak kullanabilirsiniz, ancak bu kodun bir yayında yeniden kullanımı [Kullanım Koşulları](#) ile yönetilir. Bir grafikte kullanmak için favorilere ekleyebilirsiniz.

### Feragatname

Bilgiler ve yayınlar, TradingView tarafından sağlanan veya onaylanan finansal, yatırım, işlem veya diğer türden tavsiye veya tavsiyeler anlamına gelmez ve teşkil etmez. [Kullanım Şartları](#)nda daha fazlasını okuyun.

Bu komut dosyasını bir grafikte kullanmak ister misiniz?

[★ Favori göstergelere ekle](#)

```
1 // This source code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/
2 // © KivancOzBilgic
3
4 //@version=4
5 study("VIDYA", overlay=true)
6 src = input(close, title="Source")
7 pds = input(9, "Length")
8 fixCMO = input(title="Fixed CMO Length (9)?", type=input.bool, defval=true)
9 select = input(title="Calculation Method: CMO/StDev?", type=input.bool, defval=true)
10 alpha = 2/(pds+1)
11 momm = change(src)
12 f1(m) => m >= 0.0 ? m : 0.0
13 f2(m) => m >= 0.0 ? 0.0 : -m
14 m1 = f1(momm)
15 m2 = f2(momm)
16 sm1 = fixCMO ? sum(m1, 9) : sum(m1, pds)
17 sm2 = fixCMO ? sum(m2, 9) : sum(m2, pds)
18 percent(mom, div) => 100 * mom / div
```



