## Accumulation/Distribution (AD)

**Accumulation**/**Distribution**' **Accumulation**/**distribution** is a momentum indicator that attempts to gauge supply and demand by determining whether investors are generally "accumulating," or buying, or "**distributing**," or selling, a certain stock by identifying divergences between stock price and volume flow.

## The Accumulation/Distribution (AD) study attempts to quantify the amount of volume flowing into or out of an instrument by identifying the position of the close of the period in relation to that period’s high/low range. The volume for the period is then allocated accordingly to a running continuous total.

##### **Formula**

AD = cumulative ((((Close - Low) - (High - Close)) / (High - Low)) \* Volume))

The calculation of the A/D consists of three formulas - money flow multiplier (MFM), money flow volume (MFV), and ADL.

The first formula involved in the calculation is the MFM:

**MFM = ((Close – Low) – (High – Close)) / (High – Low)**

Then we need to calculate the MFV:

**MFV = MFM x Volume on the Period**

Lastly, we have the ADL:

**ADL = Previous Period ADL + Current MFV**

Calculation**-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Open | High | Low | Close | Volume |
| 40.36 | 40.39 | 40.36 | 40.39 | 1600 |
| 40.39 | 40.4 | 40.36 | 40.36 | 5300 |
| 40.36 | 40.36 | 40.36 | 40.36 | 100 |
| 40.65 | 40.65 | 40.65 | 40.65 | 200 |
| 40.7 | 40.7 | 40.63 | 40.63 | 1100 |
| 40.72 | 40.72 | 40.72 | 40.72 | 1100 |
| 40.83 | 40.88 | 40.81 | 40.82 | 14500 |
| 41.17 | 41.2 | 41.12 | 41.2 | 17300 |
| 40.9 | 40.9 | 40.88 | 40.88 | 17000 |
| 40.88 | 40.88 | 40.88 | 40.88 | 0 |
| 41.27 | 41.27 | 41.27 | 41.27 | 100 |
| 40.97 | 41.08 | 40.97 | 41.08 | 200 |
| 41.02 | 41.05 | 41.02 | 41.05 | 400 |
| 41.03 | 41.03 | 41.03 | 41.03 | 1700 |
| 40.96 | 40.99 | 40.96 | 40.99 | 600 |
| 41 | 41.04 | 41 | 41.04 | 4600 |
| 41.05 | 41.12 | 41.03 | 41.08 | 2500 |
| 41.15 | 41.15 | 41.14 | 41.14 | 200 |
| 41.12 | 41.12 | 41.07 | 41.12 | 5600 |
| 41.12 | 41.12 | 40.95 | 41.07 | 2400 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| close-low | High-Close | High-Low | ((close-low)-(high-close)) | ((close-low)-(high-close))/(high-low) **MFM** | (((close-low)-(high-close))/(high-low))\*Volume **MFM\*V** | Cumulative(**Previous Period ADL + Current MFV)** |
| 0.029998 | 0 | 0.029998 | 0.029998 | 1 | 1600 | 1600 |
| 0 | 0.040001 | 0.040001 | -0.040001 | -1 | -5300 | -3700 |
| 0 | 0 | 0 | 0 | 0 | 0 | -3700 |
| 0 | 0 | 0 | 0 | 0 | 0 | -3700 |
| 0 | 0.07 | 0.07 | -0.07 | -1 | -1100 | -4800 |
| 0 | 0 | 0 | 0 | 0 | 0 | -4800 |
| 0.009999 | 0.060001 | 0.07 | -0.050002 | -0.714314286 | -10357.6 | -15157.6 |
| 0.080002 | 0 | 0.080002 | 0.080002 | 1 | 17300 | 2142.443 |
| 0 | 0.020001 | 0.020001 | -0.020001 | -1 | -17000 | -14857.6 |
| 0 | 0 | 0 | 0 | 0 | 0 | -14857.6 |
| 0 | 0 | 0 | 0 | 0 | 0 | -14857.6 |
| 0.110001 | 0 | 0.110001 | 0.110001 | 1 | 200 | -14657.6 |
| 0.029999 | 0 | 0.029999 | 0.029999 | 1 | 400 | -14257.6 |
| 0 | 0 | 0 | 0 | 0 | 0 | -14257.6 |
| 0.030003 | 0 | 0.030003 | 0.030003 | 1 | 600 | -13657.6 |
| 0.040001 | 0 | 0.040001 | 0.040001 | 1 | 4600 | -9057.56 |
| 0.050003 | 0.039997 | 0.09 | 0.010006 | 0.111177778 | 277.9444 | -8779.61 |
| 0 | 0.010003 | 0.010003 | -0.010003 | -1 | -200 | -8979.61 |
| 0.049999 | 0 | 0.049999 | 0.049999 | 1 | 5600 | -3379.61 |
| 0.119999 | 0.049999 | 0.169998 | 0.07 | 0.41176955 | 988.2469 | -2391.37 |

**Use Cases**

* Open a trade when you find matching signals. Open a trade in the direction of the trend if the two indicators contradict (diverge, cross) during high or increasing volumes.
* Place a stop above/beyond a support/resistance level created prior to the signal.
* Stay in the trade as long as the two indicators are supporting your trading decision.

**Python**

import numpy as np

import matplotlib.pyplot as plt

import matplotlib as mpl

import pandas as pd

from sklearn import linear\_model

from sklearn.exceptions import NotFittedError

# Importing the dataset

dataset = pd.read\_csv('C:\\Users\\Rama\\Desktop\\INR.csv')

H=dataset.iloc[:,3:4].values

L=dataset.iloc[:,4:5].values

C=dataset.iloc[:,5:6].values

V=dataset.iloc[:,7:8].values

print(V)

x=C-L

y=H-C

z=H-L

X=(x-y)/z

#X = X.replace(np.nan, 0)

X[np.isnan(X)]=0#replacing values of nan with 0

Y=X\*V

A=np.cumsum(Y)

print(A)

plt.plot(V, color = 'red', label = 'Volume')

plt.plot(A, color = 'blue', label = 'Accumulation')

plt.title('AD')

plt.xlabel('Voliume')

plt.ylabel('Cumulation')

plt.legend()

plt.show()

