

# backtest-doppia-media-mobile

August 14, 2022

```
[1]: # https://colab.research.google.com/drive/1F-Nv2Sn7YGzGmUXmz7xx0-6fHPuZQUb?
↳usp=sharing

# !pip install binance-connector pandas numpy matplotlib mplfinance

import statistics
from binance.spot import Spot as Client
import pandas as pd
import numpy as np
import mplfinance as mpf

def dati(valuta, limite):
    client = Client(base_url = 'https://api.binance.com')
    dati = client.klines(valuta, '1m', limit = limite)
    dati_df = pd.DataFrame(dati)
    dati_df[0] = (pd.to_datetime(dati_df[0], unit = 'ms'))
    dati_df = dati_df.drop(columns = [6, 7, 8, 9, 10, 11])
    dati_df.columns = ['Date', 'Open', 'High', 'Low', 'Close', 'Volume']
    dati_df = dati_df.set_index('Date')
    dati_df['Open'] = dati_df['Open'].astype(float)
    dati_df['High'] = dati_df['High'].astype(float)
    dati_df['Low'] = dati_df['Low'].astype(float)
    dati_df['Close'] = dati_df['Close'].astype(float)
    dati_df['Volume'] = dati_df['Volume'].astype(float)
    return dati_df

def doppia_media_mobile(dati, window_short, window_long):
    doppia_media_mobile_df = pd.DataFrame(index = dati.index)
    doppia_media_mobile_df['Segnali'] = 0.0
    doppia_media_mobile_df['Open'] = dati['Open']
    doppia_media_mobile_df['High'] = dati['High']
    doppia_media_mobile_df['Low'] = dati['Low']
    doppia_media_mobile_df['Close'] = dati['Close']
    doppia_media_mobile_df['Volume'] = dati['Volume']
    doppia_media_mobile_df['Media mobile corta'] = dati['Close'].rolling(window_
↳= window_short, min_periods = 1, center = False).mean()
```

```

doppia_media_mobile_df['Media mobile lunga'] = dati['Close'].rolling(window_
↪= window_long, min_periods = 1, center = False).mean()
doppia_media_mobile_df['Segnali'] [window_short:] = np.
↪where(doppia_media_mobile_df['Media mobile corta'] [window_short:] >
↪doppia_media_mobile_df['Media mobile lunga'] [window_short:], 1.0, 0.0)
doppia_media_mobile_df['Ordini'] = doppia_media_mobile_df['Segnali'].diff()
return doppia_media_mobile_df

def grafico(dati):
    mm_df = dati[['Media mobile corta', 'Media mobile lunga']]
    mm = mpf.make_addplot(mm_df)
    mpf.plot(dati, type='candle', volume=True, addplot=mm, figsize = (20, 10))

dati = dati('BNBUSDT', 60)
doppia_media_mobile_df = doppia_media_mobile(dati, 3, 7)
grafico(doppia_media_mobile_df)

```



[ ]: