SMA_CROSSOVER.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk Editing

Task 02: Using Python to analyze data and create an investing/trading strategy using simple moving average crossover

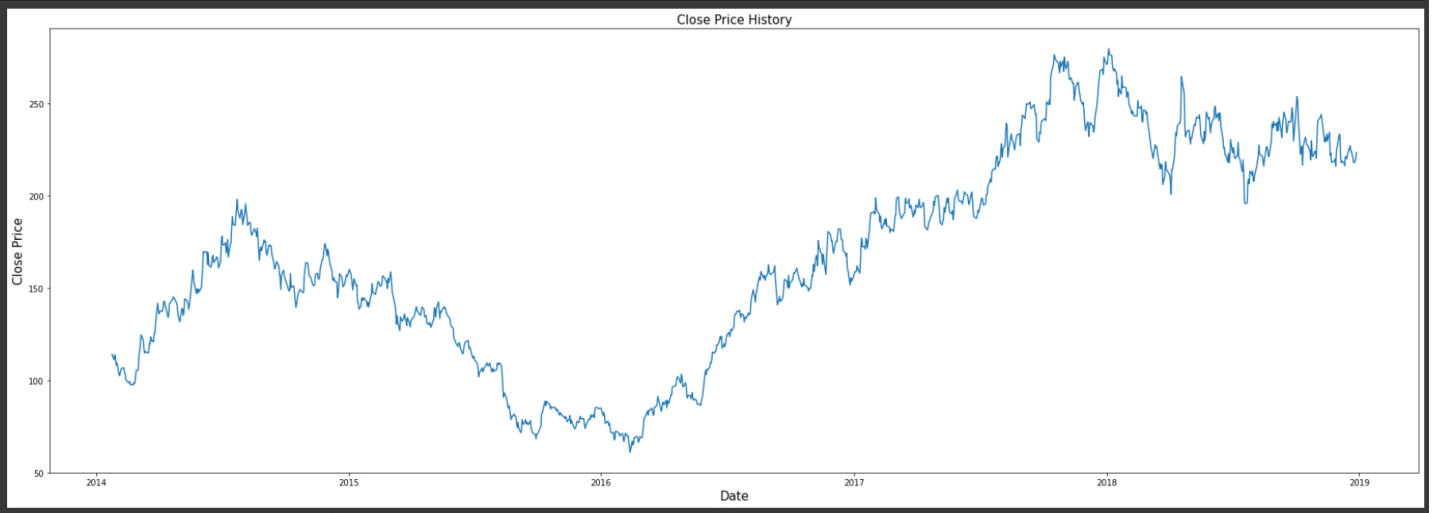
[1]
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

[2]
```
from google.colab import files
files.upload()
```

```
Choose Files HINDALCO.csv
• HINDALCO.csv(n/a) - 75604 bytes, last modified: 8/5/2021 - 100% done
Saving HINDALCO.csv to HINDALCO.csv
{'HINDALCO.csv': b'datetime,close,high,low,open,volume,instrument\r\n2014-01-24 00:00:00,114,115.35,113,113.15,5737135,HINDALCO\r\n2014-01-27 00:00:00,111.1,112.7,109.3,112,8724577,HINDA
```

[4]
```
df = pd.read_csv('HINDALCO.csv')
df = df.set_index(pd.DatetimeIndex(df['datetime'].values))

plt.figure(figsize=(30,10))
plt.title("Close Price History", fontsize=15)
plt.plot(df.close)
plt.xlabel("Date", fontsize=15)
plt.ylabel("Close Price", fontsize=15)
plt.show()
df
```



	datetime	close	high	low	open	volume	instrument
2014-01-24	2014-01-24 00:00:00	114.00	115.35	113.00	113.15	5737135	HINDALCO
2014-01-27	2014-01-27 00:00:00	111.10	112.70	109.30	112.00	8724577	HINDALCO
2014-01-28	2014-01-28 00:00:00	113.80	115.00	109.75	110.00	4513345	HINDALCO
2014-01-29	2014-01-29 00:00:00	111.75	114.75	111.15	114.50	4713458	HINDALCO
2014-01-30	2014-01-30 00:00:00	108.10	110.70	107.60	110.20	5077231	HINDALCO
...	...	...	...	...	...	...	...
2018-12-21	2018-12-21 00:00:00	224.00	227.45	221.80	224.25	6933691	HINDALCO
2018-12-24	2018-12-24 00:00:00	217.85	224.00	216.95	224.00	4667022	HINDALCO
2018-12-26	2018-12-26 00:00:00	218.60	219.80	212.90	213.85	6554865	HINDALCO
2018-12-27	2018-12-27 00:00:00	220.60	223.90	219.60	221.40	7947144	HINDALCO
2018-12-28	2018-12-28 00:00:00	223.50	226.05	221.00	221.00	5074920	HINDALCO

1215 rows × 7 columns

Function to calculate Moving Average:

[5]
```
def SimpleMovingAverage(data, period, column="close"):
  return data[column].rolling(window=period).mean()
```

```
df['MA-SHORT-SPAN'] = SimpleMovingAverage(df, 7)
df['MA-LONG-SPAN'] = SimpleMovingAverage(df, 50)
```

+ Code + Text

```
df['Signal'] = np.where(df['MA-SHORT-SPAN'] > df["MA-LONG-SPAN"], 1, 0)
df['Position'] = df['Signal'].diff()

df['Buy'] = np.where(df['Position'] == 1, df['close'], np.NaN)
df['Sell'] = np.where(df['Position'] == -1, df['close'], np.NaN)
df
```

	datetime	close	high	low	open	volume	instrument	MA-SHORT-SPAN	MA-LONG-SPAN	Signal	Position	Buy	Sell
2014-01-24	2014-01-24 00:00:00	114.00	115.35	113.00	113.15	5737135	HINDALCO	NaN	NaN	0	NaN	NaN	NaN
2014-01-27	2014-01-27 00:00:00	111.10	112.70	109.30	112.00	8724577	HINDALCO	NaN	NaN	0	0.0	NaN	NaN
2014-01-28	2014-01-28 00:00:00	113.80	115.00	109.75	110.00	4513345	HINDALCO	NaN	NaN	0	0.0	NaN	NaN
2014-01-29	2014-01-29 00:00:00	111.75	114.75	111.15	114.50	4713458	HINDALCO	NaN	NaN	0	0.0	NaN	NaN
2014-01-30	2014-01-30 00:00:00	108.10	110.70	107.60	110.20	5077231	HINDALCO	NaN	NaN	0	0.0	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2018-12-21	2018-12-21 00:00:00	224.00	227.45	221.80	224.25	6933691	HINDALCO	223.742857	226.302	0	0.0	NaN	NaN
2018-12-24	2018-12-24 00:00:00	217.85	224.00	216.95	224.00	4667022	HINDALCO	223.392857	226.121	0	0.0	NaN	NaN
2018-12-26	2018-12-26 00:00:00	218.60	219.80	212.90	213.85	6554865	HINDALCO	223.171429	226.163	0	0.0	NaN	NaN
2018-12-27	2018-12-27 00:00:00	220.60	223.90	219.60	221.40	7947144	HINDALCO	222.478571	226.041	0	0.0	NaN	NaN
2018-12-28	2018-12-28 00:00:00	223.50	226.05	221.00	221.00	5074920	HINDALCO	222.228571	225.876	0	0.0	NaN	NaN

1215 rows × 13 columns

Visual Representaion or Buy & Sell Signals

```
plt.figure(figsize=(30,10))
plt.title("Close Price History with Buy & Sell Signals", fontsize=15)
plt.plot(df.close, alpha=0.5, label="Close")
plt.plot(df["MA-SHORT-SPAN"], alpha=0.25, label="MA-SHORT-SPAN")
plt.plot(df["MA-LONG-SPAN"], alpha=0.25, label="MA-LONG-SPAN")

plt.scatter(df.index, df.Buy, alpha=1, label="Buy Signal", marker='^', color='green')
plt.scatter(df.index, df.Sell, alpha=1, label="Sell Signal", marker='v', color='red')

plt.xlabel("Date", fontsize=15)
plt.ylabel("Close Price", fontsize=15)
plt.show()
```

1s completed at 11:01 PM