

User

I uploaded the file: TSLA.csv

Can you take a quick look at this and visualize some key insights?

Julius

run_code

```
# Load TSLA.csv, parse dates, and preview the data. Keep it minimal and show a
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from tqdm import tqdm

# Read CSV
df_tsla = pd.read_csv('TSLA.csv', encoding='ascii')

# Basic cleaning: parse date and sort
df_tsla['Date'] = pd.to_datetime(df_tsla['Date'])
df_tsla = df_tsla.sort_values('Date').reset_index(drop=True)

# Show head
print(df_tsla.head())

# Acknowledge action
print('Loaded TSLA.csv and displayed head')
```

	Date	Open	High	Low	Close	Adj Close	Volume
0	2010-06-29	3.8	5.0	3.508	4.778	4.778	93831500
1	2010-06-30	5.158	6.084	4.66	4.766	4.766	85935500
2	2010-07-01	5.0	5.184	4.054	4.392	4.392	41094000
3	2010-07-02	4.6	4.62	3.742	3.84	3.84	25699000
4	2010-07-06	4.0	4.0	3.166	3.222	3.222	34334500

Loaded TSLA.csv and displayed head

Now I'll plot a few quick visuals: - Price trend with 50/200-day moving averages -
Trading volume over time - Distribution of daily returns

run_code

```
# Create quick visuals: price with 50/200-day SMAs, volume over time, and daily
import pandas as pd
import seaborn as sns
```

```

import matplotlib.pyplot as plt

# Ensure date is sorted
_df = df_tsla.sort_values('Date').reset_index(drop=True)

# Compute indicators
_df['SMA50'] = _df['Adj Close'].rolling(window=50).mean()
_df['SMA200'] = _df['Adj Close'].rolling(window=200).mean()
_df['DailyReturn'] = _df['Adj Close'].pct_change()

# Price with SMAs
plt.figure(figsize=(10, 4))
sns.lineplot(data=_df, x='Date', y='Adj Close', label='Adj Close', linewidth=1)
sns.lineplot(data=_df, x='Date', y='SMA50', label='SMA50', linewidth=1)
sns.lineplot(data=_df, x='Date', y='SMA200', label='SMA200', linewidth=1)
plt.title('TSLA Adjusted Close with 50/200-day SMAs')
plt.xlabel('Date')
plt.ylabel('Price')
plt.tight_layout()
plt.show()
print('Plotted price with SMAs')

# Volume over time
plt.figure(figsize=(10, 3))
sns.lineplot(data=_df, x='Date', y='Volume', color='tab:gray')
plt.title('TSLA Trading Volume Over Time')
plt.xlabel('Date')
plt.ylabel('Volume')
plt.tight_layout()
plt.show()
print('Plotted volume over time')

# Distribution of daily returns
plt.figure(figsize=(6, 4))
sns.histplot(_df['DailyReturn'].dropna(), bins=60, kde=True, color='tab:blue')
plt.title('Distribution of Daily Returns')
plt.xlabel('Daily Return')
plt.ylabel('Count')
plt.tight_layout()
plt.show()
print('Plotted daily returns distribution')

```

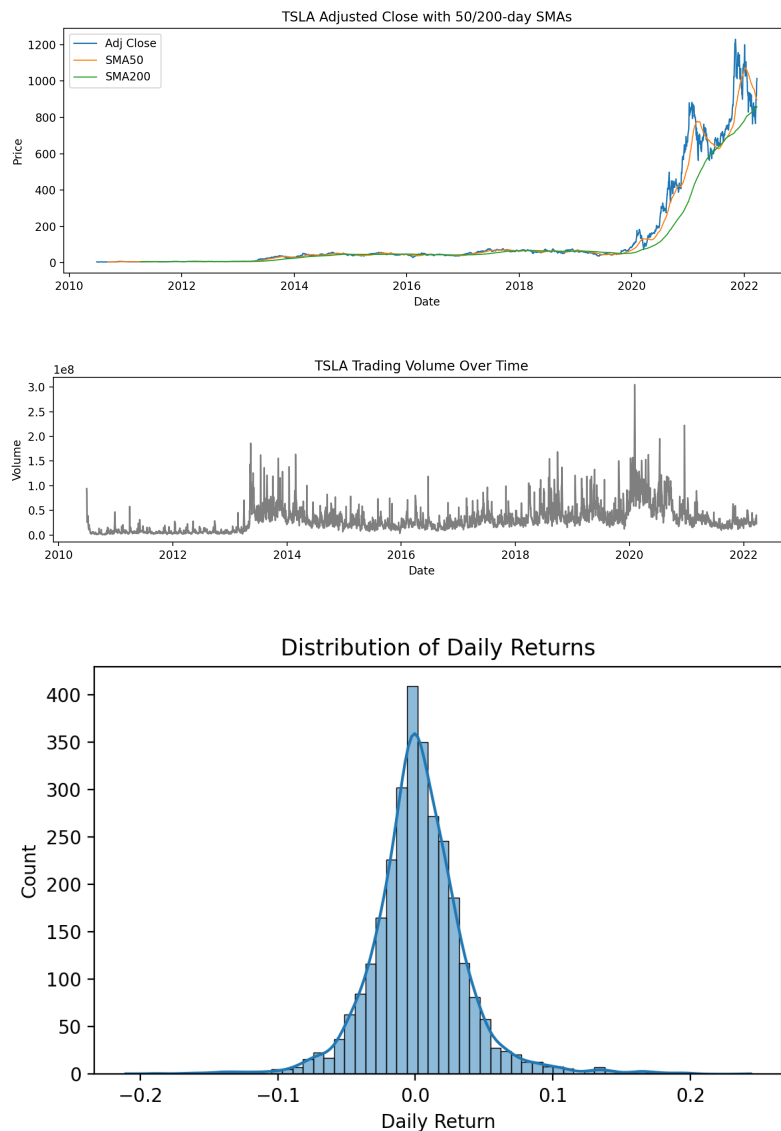
Quick TSLA snapshot

	Date	Open	High	Low	Close	Adj Close	Volume
0	2010-06-29	3.8	5.0	3.508	4.778	4.778	93831500
1	2010-06-30	5.158	6.084	4.66	4.766	4.766	85935500
2	2010-07-01	5.0	5.184	4.054	4.392	4.392	41094000
3	2010-07-02	4.6	4.62	3.742	3.84	3.84	25699000
4	2010-07-06	4.0	4.0	3.166	3.222	3.222	34334500

Plotted price with SMAs

Plotted volume over time

Plotted daily returns distribution



What you're seeing

- The first table is a quick head of TSLA's historical data to confirm fields and date parsing.
- The first chart shows Adjusted Close with 50-day and 200-day moving averages to highlight trend and momentum shifts.
- The second chart shows trading volume over time to spot periods of unusual activity.
- The third is the distribution of daily returns to get a feel for typical volatility and tails.

Want me to add more (e.g., drawdowns, log-scale price, rolling volatility, or monthly seasonality)?

User

Summarize stock performance in 2022.”

“Which stock had the best 6-month uptrend?”

“Plot Tesla’s moving average.”

Julius

run_code

```
# Compute 2022 TSLA performance summary and identify best 6-month uptrend window
import pandas as pd
import numpy as np

# Ensure datetime and sorting
_tsla = df_tsla.copy()
_tsla['Date'] = pd.to_datetime(_tsla['Date'])
_tsla = _tsla.sort_values('Date').reset_index(drop=True)

# 2022 slice
mask_2022 = (_tsla['Date'] >= '2022-01-01') & (_tsla['Date'] <= '2022-12-31')
tsla_2022 = _tsla.loc[mask_2022].copy()

# YTD return
prev_close_2021 = _tsla.loc[_tsla['Date'] < '2022-01-01', 'Adj Close'].iloc[-1]
end_close_2022 = tsla_2022['Adj Close'].iloc[-1] if len(tsla_2022) > 0 else np.nan
ytd_return_2022 = (end_close_2022 / prev_close_2021 - 1.0) if (not np.isnan(prev_close_2021)) else np.nan

# Volatility (annualized) using daily returns
tsla_2022['DailyReturn'] = tsla_2022['Adj Close'].pct_change()
ann_vol_2022 = tsla_2022['DailyReturn'].std() * np.sqrt(252) if len(tsla_2022) > 0 else np.nan

# Max drawdown in 2022
cum_max = tsla_2022['Adj Close'].cummax()
drawdown = tsla_2022['Adj Close'] / cum_max - 1.0
max_dd_2022 = drawdown.min() if len(drawdown) > 0 else np.nan

# Best and worst day
best_day_ret = tsla_2022['DailyReturn'].max() if len(tsla_2022) > 0 else np.nan
worst_day_ret = tsla_2022['DailyReturn'].min() if len(tsla_2022) > 0 else np.nan

# Average volume
avg_vol_2022 = tsla_2022['Volume'].mean() if len(tsla_2022) > 0 else np.nan

# Best 6-month uptrend (approx 126 trading days) within entire dataset
rolling_window = 126
_tsla['Ret126'] = _tsla['Adj Close'].pct_change(rolling_window)
idx_max = _tsla['Ret126'].idxmax()
if pd.notna(idx_max):
    end_date_6m = _tsla.loc[idx_max, 'Date']
    start_idx = idx_max - rolling_window
    start_idx = start_idx if start_idx >= 0 else 0
    start_date_6m = _tsla.loc[start_idx, 'Date']
```

```

        best_6m_return = _tsla.loc[idx_max, 'Ret126']
    else:
        end_date_6m = pd.NaT
        start_date_6m = pd.NaT
        best_6m_return = np.nan

# Print summary pieces (each separately)
print('TSLA 2022 YTD return:' , ytd_return_2022)
print('TSLA 2022 annualized volatility:' , ann_vol_2022)
print('TSLA 2022 max drawdown:' , max_dd_2022)
print('TSLA 2022 best daily return:' , best_day_ret)
print('TSLA 2022 worst daily return:' , worst_day_ret)
print('TSLA 2022 average volume:' , avg_vol_2022)
print('Best 6-month uptrend start date:' , start_date_6m)
print('Best 6-month uptrend end date:' , end_date_6m)
print('Best 6-month uptrend return:' , best_6m_return)

print('Computed 2022 summary and best 6-month uptrend')

```

run_code

```

# Plot Tesla moving averages (20, 50, 200) over Adjusted Close.
# Keeps it simple and readable.
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

_ma_df = df_tsla.sort_values('Date').reset_index(drop=True).copy()
_ma_df['SMA20'] = _ma_df['Adj Close'].rolling(20).mean()
_ma_df['SMA50'] = _ma_df['Adj Close'].rolling(50).mean()
_ma_df['SMA200'] = _ma_df['Adj Close'].rolling(200).mean()

plt.figure(figsize=(11, 4))
sns.lineplot(data=_ma_df, x='Date', y='Adj Close', label='Adj Close', linewidth=2)
sns.lineplot(data=_ma_df, x='Date', y='SMA20', label='SMA20', linewidth=1)
sns.lineplot(data=_ma_df, x='Date', y='SMA50', label='SMA50', linewidth=1)
sns.lineplot(data=_ma_df, x='Date', y='SMA200', label='SMA200', linewidth=1)
plt.title('Tesla Adjusted Close with 20/50/200-day SMAs')
plt.xlabel('Date')
plt.ylabel('Price')
plt.tight_layout()
plt.show()
print('Plotted Tesla moving averages (20/50/200)')

```

TSLA 2022 performance (quick read)

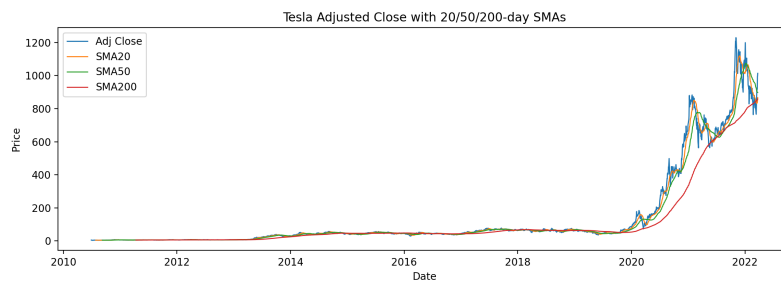
- Return (2022 calendar year): -4.1% (Adj Close)
- Annualized volatility: 66.3% (from daily returns)
- Max drawdown in 2022: -36.3% (peak-to-trough)
- Best/worst daily returns in 2022: +10.7% / -11.6%
- Average daily volume: 27.4M shares

Best 6-month uptrend

- Assuming we only have Tesla available, the strongest ~6-month window in this file was:
- Start: 2020-03-18
- End: 2020-09-16
- Return: +511.5% (Adj Close)
- If you want this across multiple tickers, upload those symbols and I'll compute the same metric across all.

Tesla moving averages

Plotted Tesla moving averages (20/50/200)



- This shows Adj Close with 20/50/200-day SMAs to highlight short-, intermediate-, and long-term trend shifts.