monte-carlo-drift

September 29, 2021

[1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from tqdm import tqdm
sns.set()
```

[2]:

```
df = pd.read_csv('../dataset/TSLA.csv')
df.head()
```

[2]:

```
         Date        Open        High         Low       Close   Adj Close  \
0  2018-03-23  311.250000  311.250000  300.450012  301.540009  301.540009
1  2018-03-26  307.339996  307.589996  291.359985  304.179993  304.179993
2  2018-03-27  304.000000  304.269989  277.179993  279.179993  279.179993
3  2018-03-28  264.579987  268.679993  252.100006  257.779999  257.779999
4  2018-03-29  256.489990  270.959991  248.210007  266.130005  266.130005

     Volume
0   6654900
1   8375200
2  13872000
3  21001400
4  15170700
```

[3]:

```
number_simulation = 100
predict_day = 30

close = df['Close'].tolist()
returns = pd.DataFrame(close).pct_change()
last_price = close[-1]
results = pd.DataFrame()
avg_daily_ret = returns.mean()
variance = returns.var()
daily_vol = returns.std()
daily_drift = avg_daily_ret - (variance / 2)
drift = daily_drift - 0.5 * daily_vol ** 2
```

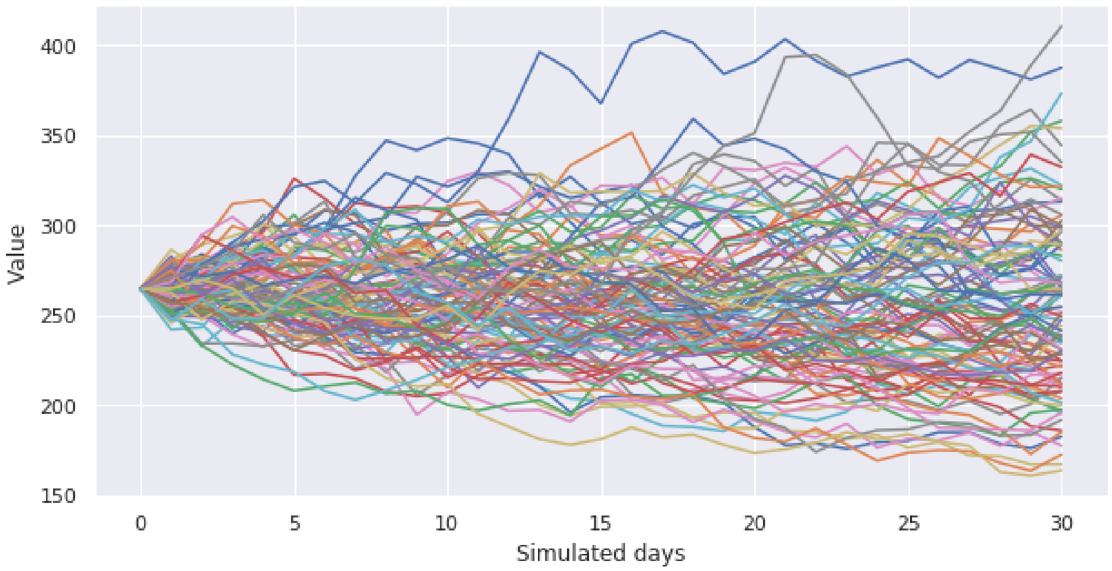
```
results = pd.DataFrame()

for i in tqdm(range(number_simulation)):
    prices = []
    prices.append(df.Close.iloc[-1])
    for d in range(predict_day):
        shock = [drift + daily_vol * np.random.normal()]
        shock = np.mean(shock)
        price = prices[-1] * np.exp(shock)
        prices.append(price)
    results[i] = prices
```

```
100%|          | 100/100 [00:00<00:00, 104.14it/s]
```

[4]:

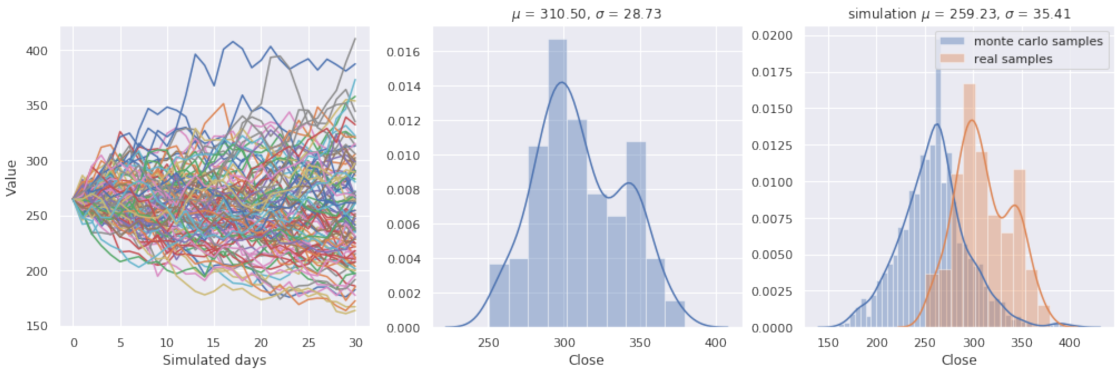
```
plt.figure(figsize=(10,5))
plt.plot(results)
plt.ylabel('Value')
plt.xlabel('Simulated days')
plt.show()
```



[5]:

```
raveled = results.values.ravel()
raveled.sort()
cp_raveled = raveled.copy()

plt.figure(figsize=(17,5))
plt.subplot(1,3,1)
plt.plot(results)
plt.ylabel('Value')
plt.xlabel('Simulated days')
plt.subplot(1,3,2)
sns.distplot(df.Close,norm_hist=True)
plt.title('$\mu$ = %.2f, $\sigma$ = %.2f'%(df.Close.mean(),df.Close.std()))
plt.subplot(1,3,3)
sns.distplot(raveled,norm_hist=True,label='monte carlo samples')
sns.distplot(df.Close,norm_hist=True,label='real samples')
plt.title('simulation $\mu$ = %.2f, $\sigma$ = %.2f'%(raveled.mean(),raveled.
 ↪std()))
plt.legend()
plt.show()
```


Value
Simulated days
μ = 310.50, σ = 28.73
Close
simulation μ = 259.23, σ = 35.41
monte carlo samples
real samples
Close