import numpy as np

import matplotlib.pyplot as plt

from scipy import signal

trading\_volume=np.random.randint(50,150,126)

noise=np.random.randint(0,5,126)

trading\_volnoise=trading\_volume+noise

def filterr(data,cutoff,fs,order=5):

nyq=0.5\*fs

n\_off=cutoff/nyq

b,a=signal.butter(order,n\_off,btype='low',analog=False)

y=signal.filtfilt(b,a,data)

return y

cutoff=0.1

fs=1

order=3

smoothed\_vol=filterr(trading\_volnoise,cutoff,fs,order)

print(smoothed\_vol)

total\_vol=np.sum(smoothed\_vol.reshape((-1,7)),axis=1)

plt.plot(trading\_volume,label="trade\_vol")

plt.plot(trading\_volnoise,alpha=0.6,color="red",label="noisy\_trade\_vol")

plt.plot(smoothed\_vol,alpha=0.9,label="smoothed\_Vol",color="green")

# plt.scatter(np.linspace(0,126,18),total\_vol)

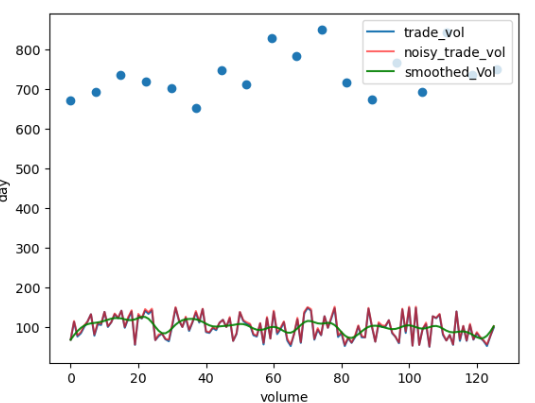
plt.xlabel("volume")

plt.ylabel("day")

plt.legend(loc="upper right")

# plt.plot(total\_vol)plt.scatter(np.linspace(0,126,18),total\_vol)

plt.show()



def detect\_periods(trading\_volnoise):

avg\_vol = np.mean(trading\_volnoise)

threshold = 1.5 \* avg\_vol

periods = []

period = []

for i in range(0, len(trading\_volnoise)):

if trading\_volnoise[i] > threshold:

period.append(i)

else:

if len(period) > 3:

periods.append([period[0], period[-1]])

period = []

if len(period) > 3:

periods.append([period[0], period[-1]])

plt.figure(figsize=(10, 6))

plt.plot(trading\_volnoise, label="Trading Volume")

for p in periods:

plt.axvspan(p[0], p[1], color='red', alpha=0.3, label="High Volume Period" if p == periods[0] else "")

plt.axhline(threshold, color='green', linestyle='--', label=f"150% of Avg Volume ({threshold:.2f})")

plt.legend()

plt.title("Trading Volume with Detected High Volume Periods")

plt.xlabel("Day")

plt.ylabel("Volume")

plt.show()

return periods

detect\_periods(trading\_volnoise)

