Gary He

Homework 5, Unit 1: Filter Design, DFT Properties and Applications

Prof. Jordan C. Hanson

March 29, 2025

1 Memory Bank

- **Convolution**: this is an operation that characterizes the response $h[n]$ of a linear system.

$$y[i] = h[n] * x[n] = \sum_{j=0}^{M-1} h[j]x[i-j] \quad (1)$$

  In words, the output at sample i is equal to the produce of the system response h and the input signal x, summed over the proceeding M samples (from $j = 0$ to $j = M - 1$).

- **Discrete Delta Function**, $\delta[n]$: A standard impulse response that contains one non-zero sample. It has the following property:

$$x[n] = \delta[n] * x[n] \quad (2)$$

- **Discrete Fourier Transform**, for a sampled, digitized signal x_n:

$$X_{\mathrm{k}} = \sum_{n=0}^{N-1} x_n e^{-2\pi j (k/N) n} \quad (3)$$

- In DFT analysis, we often need to know the Δt, time duration for samples, and the sampling rate, f_{s}. Note that $1/f_{\mathrm{s}} = \Delta t$.
- For a sinusoid of frequency f (Hz), the period is $T = 1/f$ (seconds).
- **Inverse Discrete Fourier Transform**, for a sampled, digitized signal X_k in the frequency domain:

$$x_{\mathrm{n}} = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{2\pi j (k/N) n} \quad (4)$$

2 Discrete Fourier Transform, Applications

1. **Download and Graph Data.** On our Course Moodle page under Unit 1, download the file "Stock Data, Google Alphabet Inc., 2015-2025." Move it into a folder accessible to `octave`. Use the `csvread` function to import the data into the `octave` workspace. Plot the data and label the x-axis "Days" and the y-axis "Price (USD)." For example, if the CSV data is stored in a variable `data`, then plot it via

```
plot(data(:,1),data(:,2),'-','color','black')
```

2. **Create the discrete Fourier transform.** The units of the graph are stock price (closing) in USD, versus days. Day 1 corresponds to January 1st, 2015. Using techniques we covered in previous code labs, create and graph the magnitude of the DFT of the stock data.
3. **Identify Peaks and Frequencies**. What peaks, if any, do you observe? What are the corresponding frequencies?

3 Filter Design

1. **Smoothing the Time Series Data.** As in a previous code lab, implement a *running average filter* kernel, and convolve it with the time series data. Use this filter to smooth the data, and plot it with the original, unfiltered data.
2. **Graph the Filtered Spectrum**. Add the magnitude of the DFT of the filtered data to the graph of the magnitude of the DFT of the raw data. Does the result make sense?

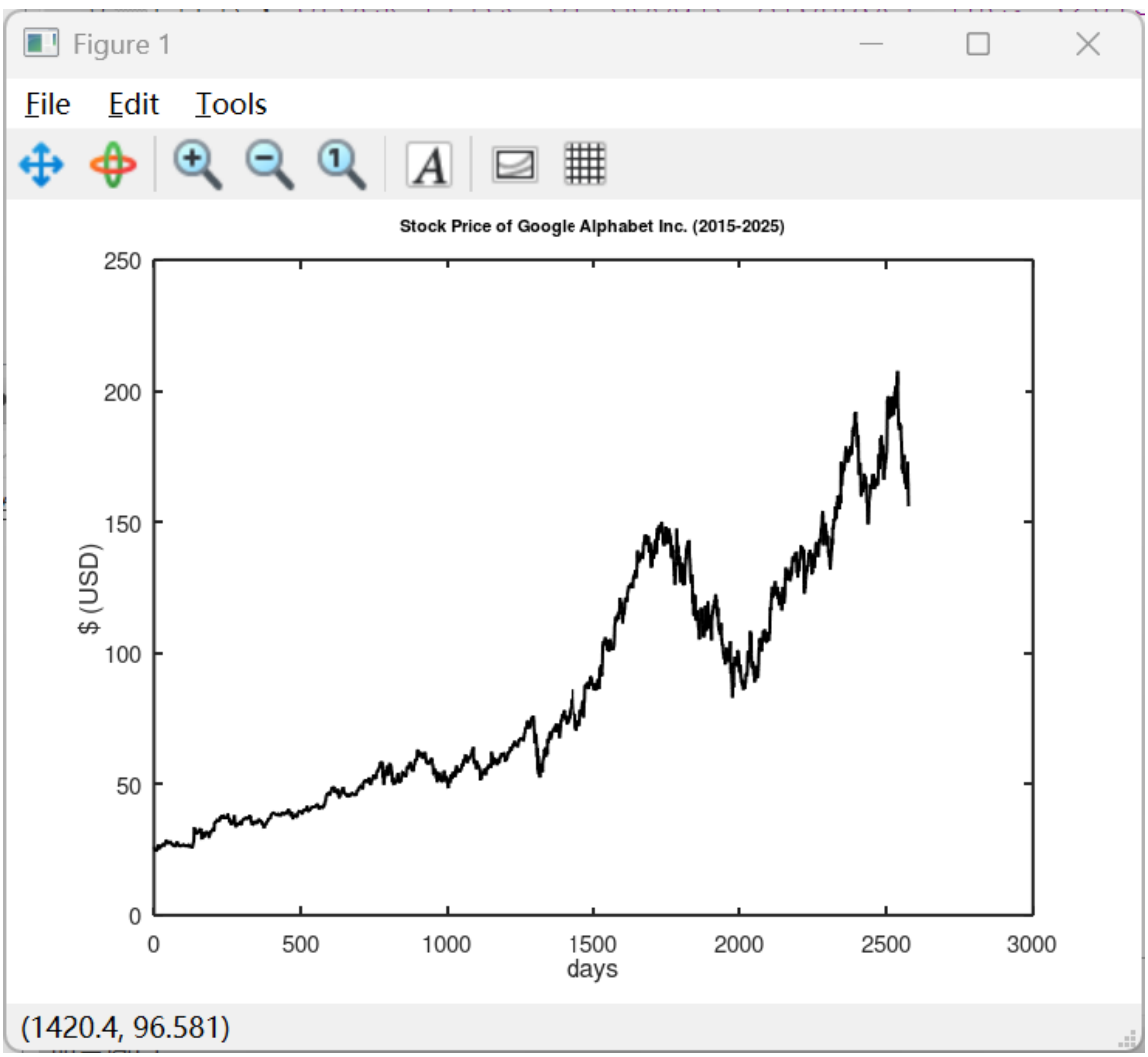
1. **Download and Graph Data.** On our Course Moodle page under Unit 1, download the file "Stock Data, Google Alphabet Inc., 2015-2025." Move it into a folder accessible to **octave**. Use the **csvread** function to import the data into the **octave** workspace. Plot the data and label the x-axis "Days" and the y-axis "Price (USD)." For example, if the CSV data is stored in a variable **data**, then plot it via

   ```
   plot(data(:,1),data(:,2),'-','color','black')
   ```

```
%==================Problem 2.1=====================
data = csvread('C:/Users/49902/Desktop/problem_Set#5/StockData_2015_2025.csv'); %read file

figure 1;
plot(data(:,1), data(:,2), '-', 'color', 'black');
xlabel('days');
ylabel('$ (USD)');
title('Stock Price of Google Alphabet Inc. (2015-2025)');

```

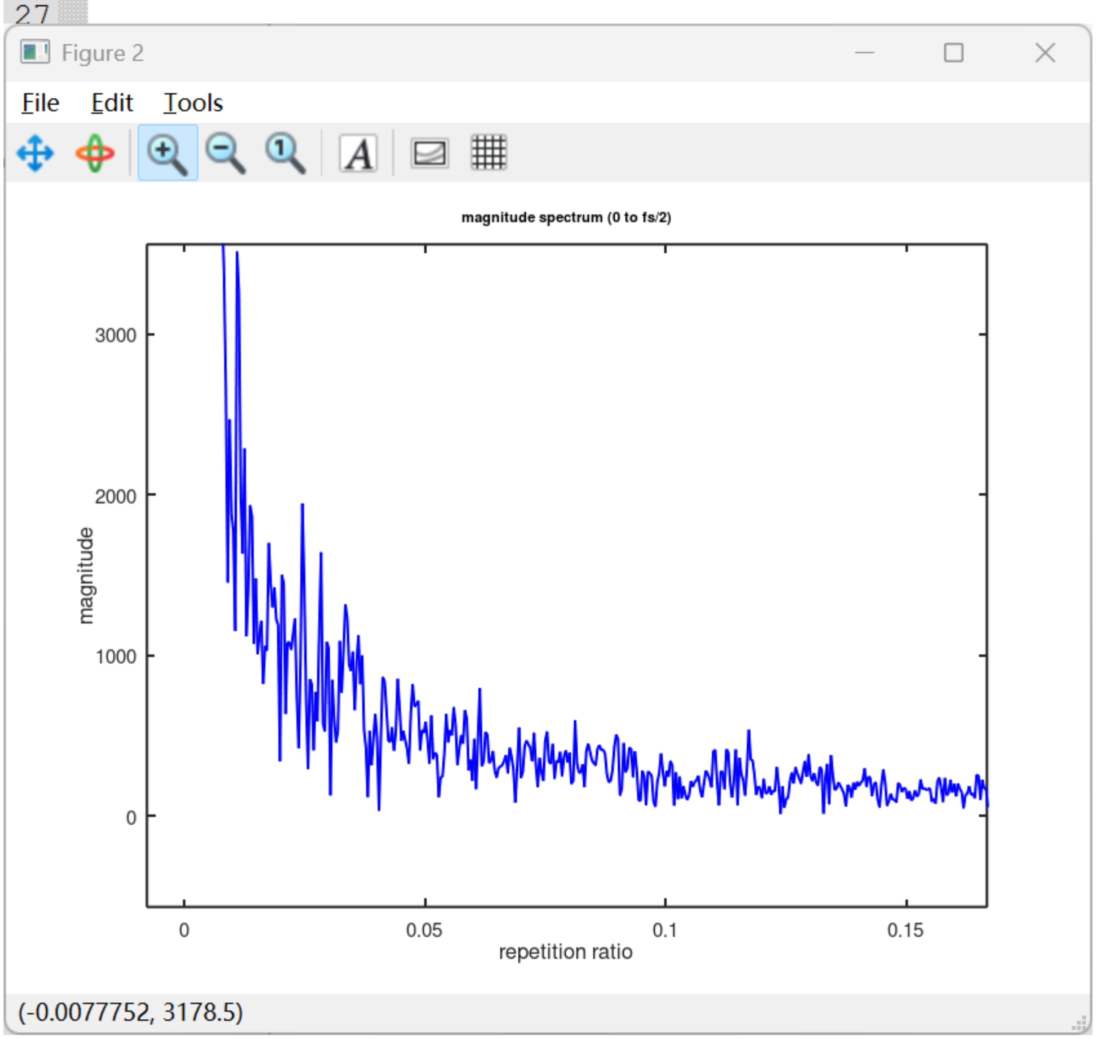


2. **Create the discrete Fourier transform.** The units of the graph are stock price (closing) in USD, versus days. Day 1 corresponds to January 1st, 2015. Using techniques we covered in previous code labs, create and graph the magnitude of the DFT of the stock data.

```
%========= problem 2.2 =====================
x = data(:,2);
n = length(x);
fs = 1;
X = fft(x);
f = (0:n-1) * fs / n;
magnitude = abs(X);% magnitude spectrum

% 0 to 1/2 f
figure 2;
plot(f(1:floor(n/2)), magnitude(1:floor(n/2)), 'b-');
xlabel('repetition ratio');
ylabel('magnitude');
title('magnitude spectrum (0 to fs/2)');

```



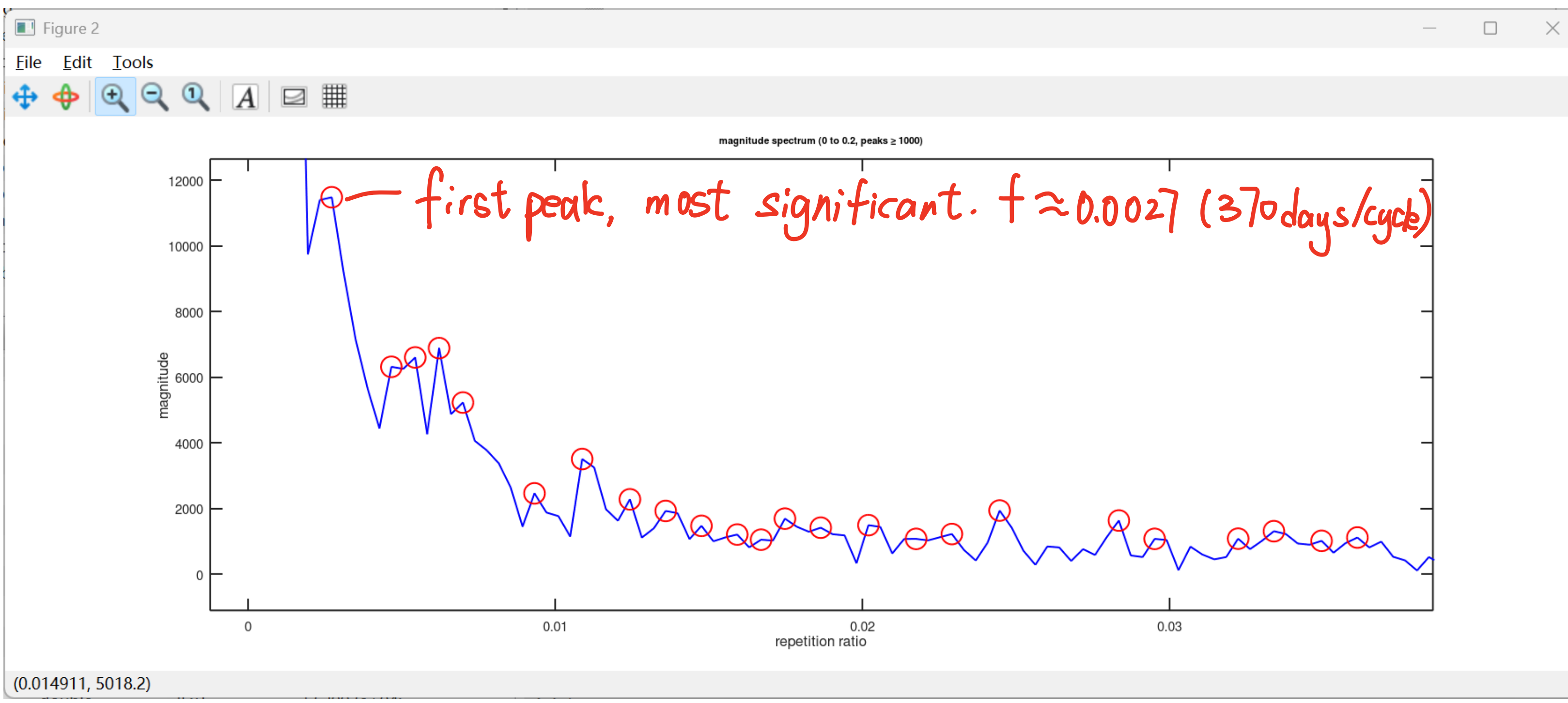
3. **Identify Peaks and Frequencies.** What peaks, if any, do you observe? What are the corresponding frequencies?

```
%========= problem 2.3 ====================
x = data(:,2);
n = length(x);
fs = 1;
X = fft(x);
f = (0:n-1) * fs / n;
magnitude = abs(X); % magnitude spectrum

%only 0 to 0.2 ratio
index = find(f <= 0.2);
f_range = f(index);
m_range = magnitude(index);

% find peaks, only those with magnitude >= 1000
[pks, locs] = findpeaks(m_range);
valid_index = find(pks >= 1000);
pks = pks(valid_index);
locs = locs(valid_index);


figure(2);
plot(f_range, m_range, 'b-'); % blue line: spectrum
hold on;
plot(f_range(locs), pks, 'ro'); % red dots: peaks ≥ 1000
xlabel('repetition ratio');
ylabel('magnitude');
title('magnitude spectrum (0 to 0.2, peaks ≥ 1000)');
```



In the frequency spectrum, the first significant peak appears at f=0.0027, which corresponds to a cycle of approximately 370 days. The magnitude of this peak exceeds 10,000, shows a strong and stable long-term trend that repeats roughly once a year. Additionally, most peaks with a magnitude above 1,000 are found within the frequency range of f=0~0.04. This indicates that the stock price tends to exhibit low-frequency, long-term cyclical behavior, rather than short-term fluctuations.

3 Filter Design

1. **Smoothing the Time Series Data.** As in a previous code lab, implement a *running average filter* kernel, and convolve it with the time series data. Use this filter to smooth the data, and plot it with the original, unfiltered data.

```
%==================Problem 3.1 ========================
% days and price
days = data(:,1);
price = data(:,2);
h = ones(1,10)/10;% 10 pt average filter
smoothed_price = conv(price, h, "same"); %same lenth

figure(3);
%original
subplot(2,1,1);
plot(days, price, '-', 'color', 'black');
xlabel('days');
ylabel('$ (USD)');
title('Original Stock Price');

%smoothed
subplot(2,1,2);
plot(days, smoothed_price, '-', 'color', 'red');
xlabel('days');
ylabel('$ (USD)');
title('Smoothed Stock Price (10-day average)');
```

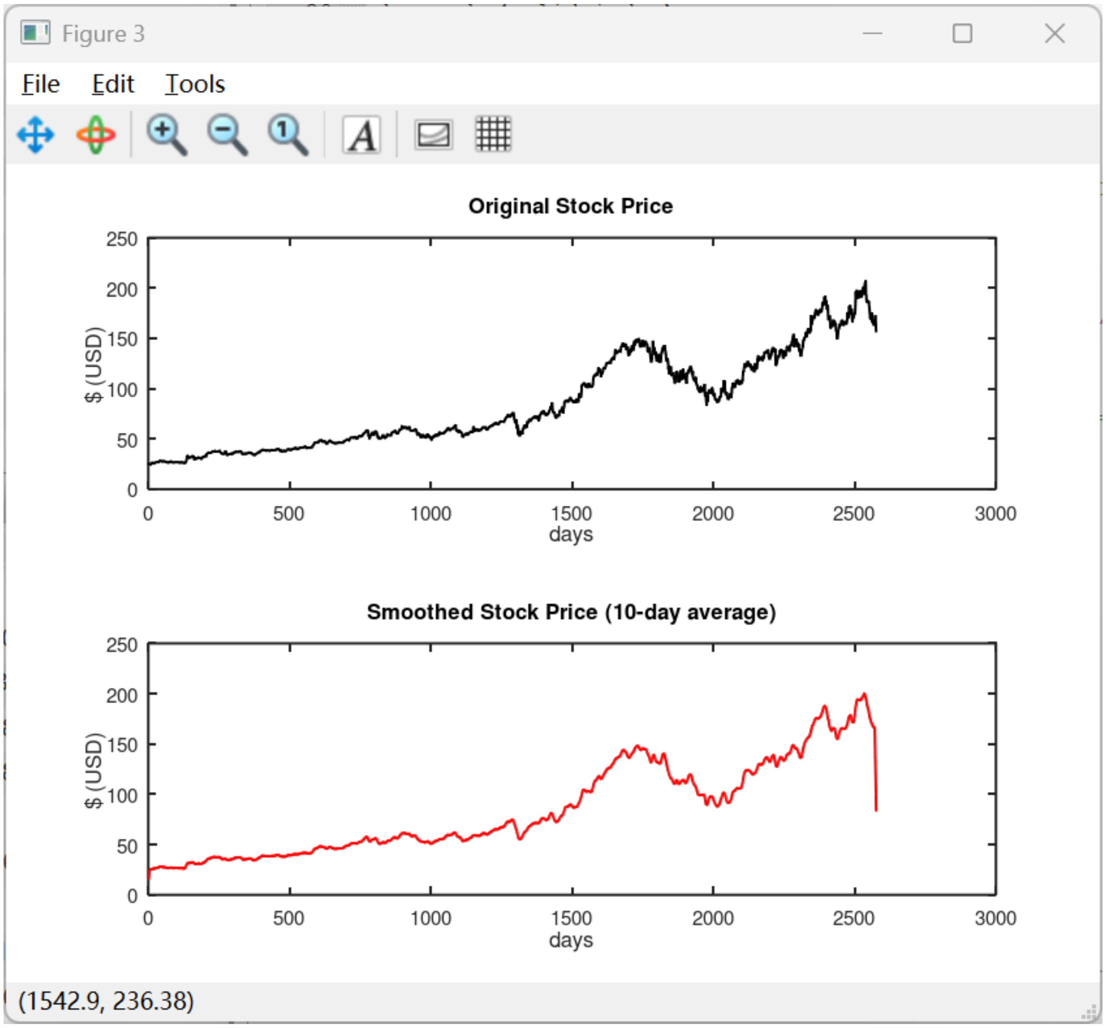


Figure 3
File
Edit
Tools
Original Stock Price
$ (USD)
250
200
150
100
50
0
0
500
1000
1500
2000
2500
3000
days
Smoothed Stock Price (10-day average)
$ (USD)
250
200
150
100
50
0
0
500
1000
1500
2000
2500
3000
days
(1542.9, 236.38)

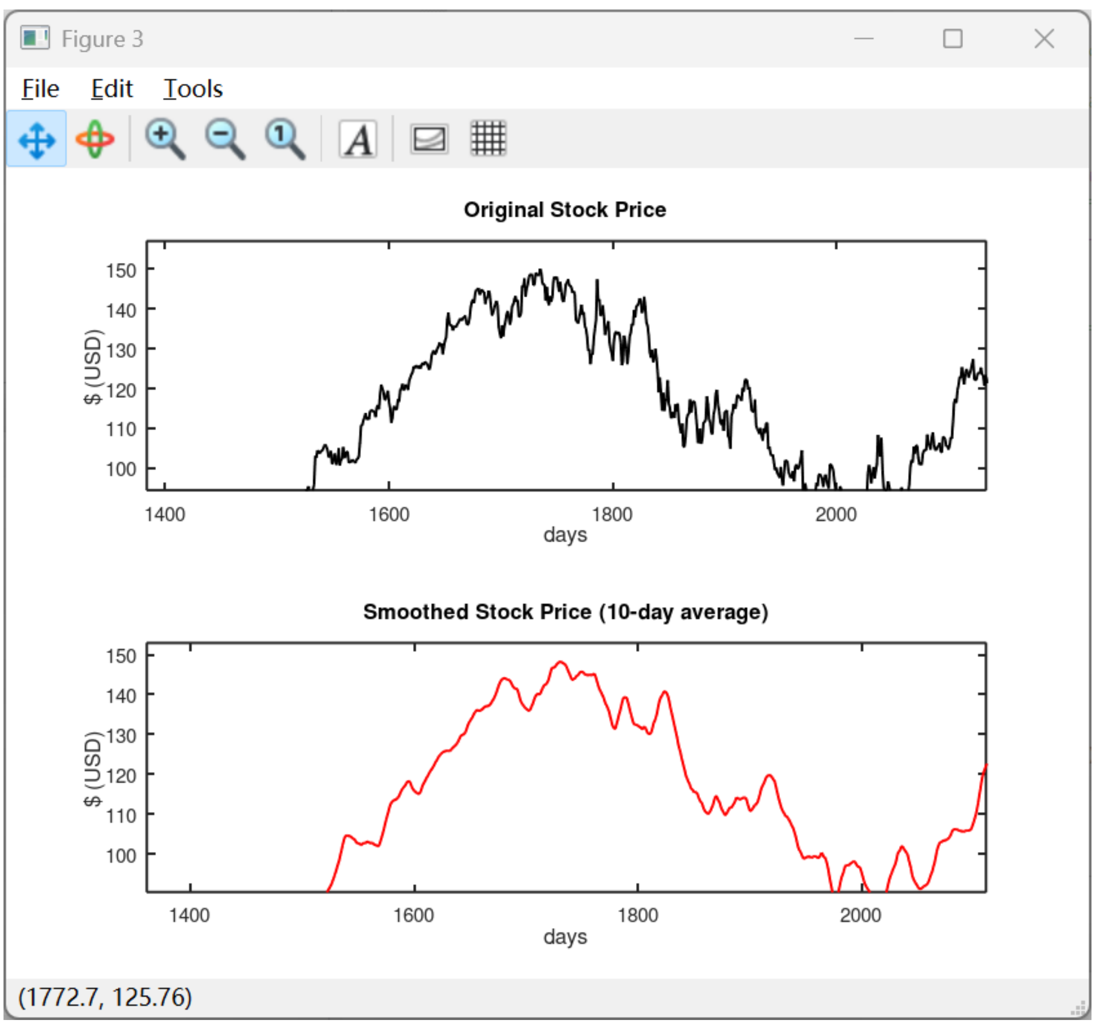


Figure 3
File
Edit
Tools
Original Stock Price
$ (USD)
150
140
130
120
110
100
1400
1600
1800
2000
days
Smoothed Stock Price (10-day average)
$ (USD)
150
140
130
120
110
100
1400
1600
1800
2000
days
(1772.7, 125.76)

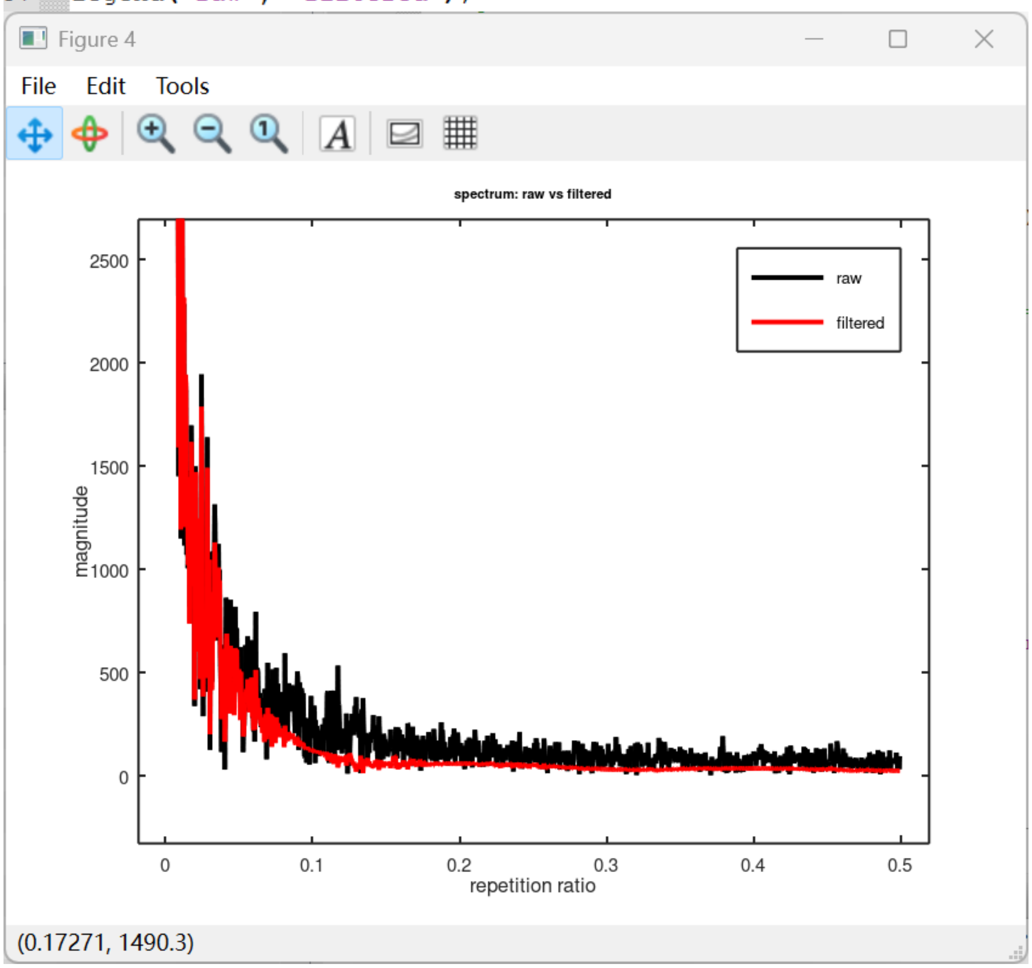
2. **Graph the Filtered Spectrum**. Add the magnitude of the DFT of the filtered data to the graph of the magnitude of the DFT of the raw data. Does the result make sense?

```
%=================Problem 3.3 ========================
n = length(price);
fs = 1;
f = (0:n-1) * fs / n;

X_raw = fft(price);
X_filtered = fft(smoothed_price);

mag_raw = abs(X_raw);
mag_filtered = abs(X_filtered);
half_n = floor(n/2);f


figure(4);
plot(f(1:half_n), mag_raw(1:half_n), 'k-', 'linewidth', 1); %original
hold on;
plot(f(1:half_n), mag_filtered(1:half_n), 'r-', 'linewidth', 1); %filtered
xlabel('repetition ratio');
ylabel('magnitude');
title('spectrum: raw vs filtered');
legend('raw', 'filtered');
```



After applying the moving average filter, the DFT magnitude of the filtered data shows a clear reduction in high-frequency components compared to the raw data. This makes sense, since the filter smooths out short-term fluctuations while preserving the overall long-term trend.