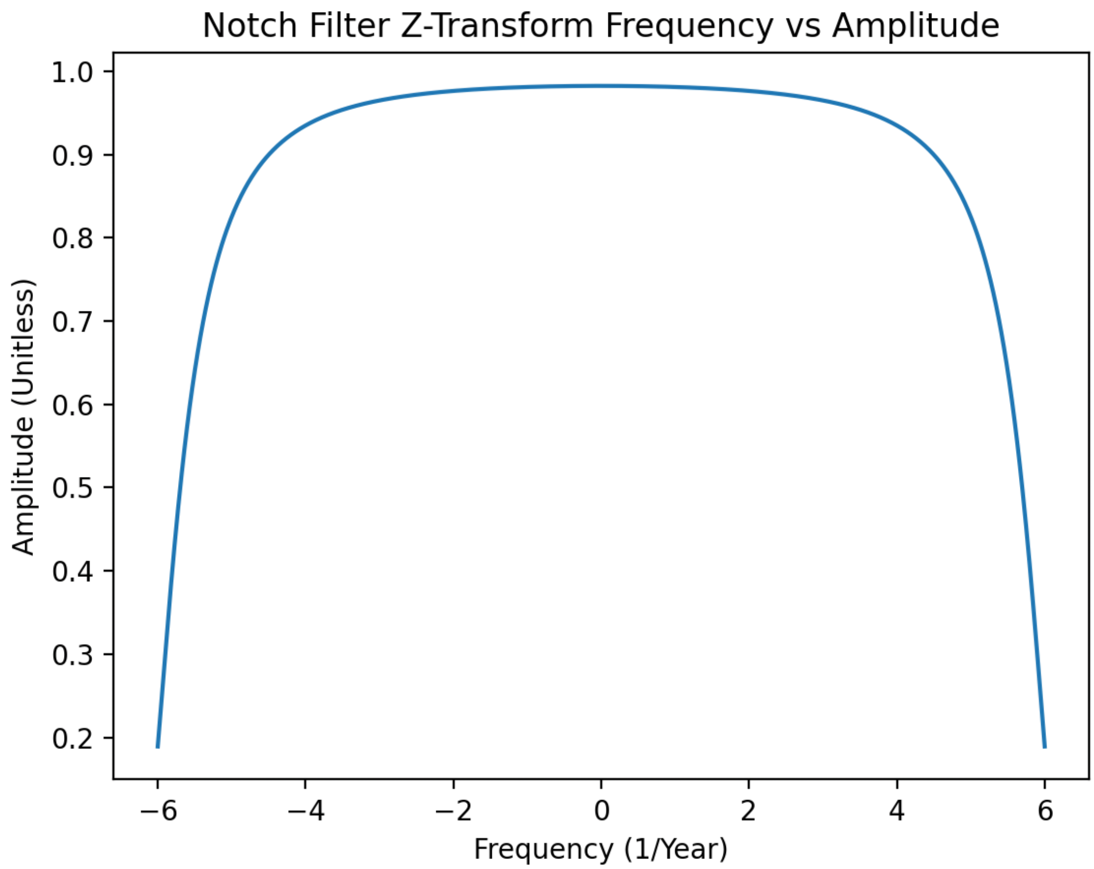
No Collaboration for any part.
Exercise 1.
1.
Zeros are located at $z = q$ and $z = q^*$ while poles are located at $z = p$ and $z = p^*$. As $q = e^{-i2\pi f_0/f_s}$ the zeros lie on the unit circle. Meanwhile since $p = (1+\epsilon)q$ and $0 < \epsilon \ll 1$, the poles lie slightly outside of the unit circle. Therefore, because the poles lie outside the unit circle, the system is stable. Finally, based on the definition of $W(z) = M\frac{z-q}{z-p}\frac{z-q^*}{z-p^*}$ we can determine that firstly $W(z) = W^*(z)$ and if $W(z) = \frac{X(z)}{Y(z)}$ then $W(-z) = \frac{X(z)+2z(q+q^*)}{Y(z)+2z(p+p^*)}$ so $W(z) \neq W^*(-z)$, so this filter is not real.

2.
```python
def W(M,z,f0,fs,e):
    q1 = np.exp(-1j*2*np.pi*(f0)/(fs))
    q2 = np.exp(1j*2*np.pi*(f0)/(fs))
    p1 = (1+e)*q1
    p2 = (1+e)*q2
    return M*((z-q1)/(z-p1))*((z-q2)/(z-p2))

M = 1.05
f0 = 1
fs = 12
e = 0.05

w = np.linspace(-fs/2,fs/2,1000)
z = np.exp(-1j*w*(1/12))

plt.plot(w, np.abs(W(M,z,f0,fs,e)**2))
plt.xlabel("Frequency (1/Year)")
plt.ylabel("Amplitude (Unitless)")
plt.title("Notch Filter Z-Transform Frequency vs Amplitude")
plt.show()
```



3.

```
for i in range(len(np.abs(W(M,z,f0,fs,e)**2))):
    if np.abs(W(M,z,f0,fs,e)**2)[i] > 0.5*np.abs(W(M,z,f0,fs,e)**2)[499]:
        print(np.abs(2*w[i]))
        break
```

The FWHM is 11.4/year, to increase the sharpness, and thereby decrease the FWHM, the epsilon value should be increased.

Exercise 2.

1.

By expanding out the equation, it can be determined that:

$a = \frac{Mqq^*}{pp^*}, b = \frac{-M(q+q^*)}{pp^*}, c = \frac{M}{pp^*}, B = \frac{-(p+p^*)}{pp^*}, C = \frac{1}{pp^*}.$

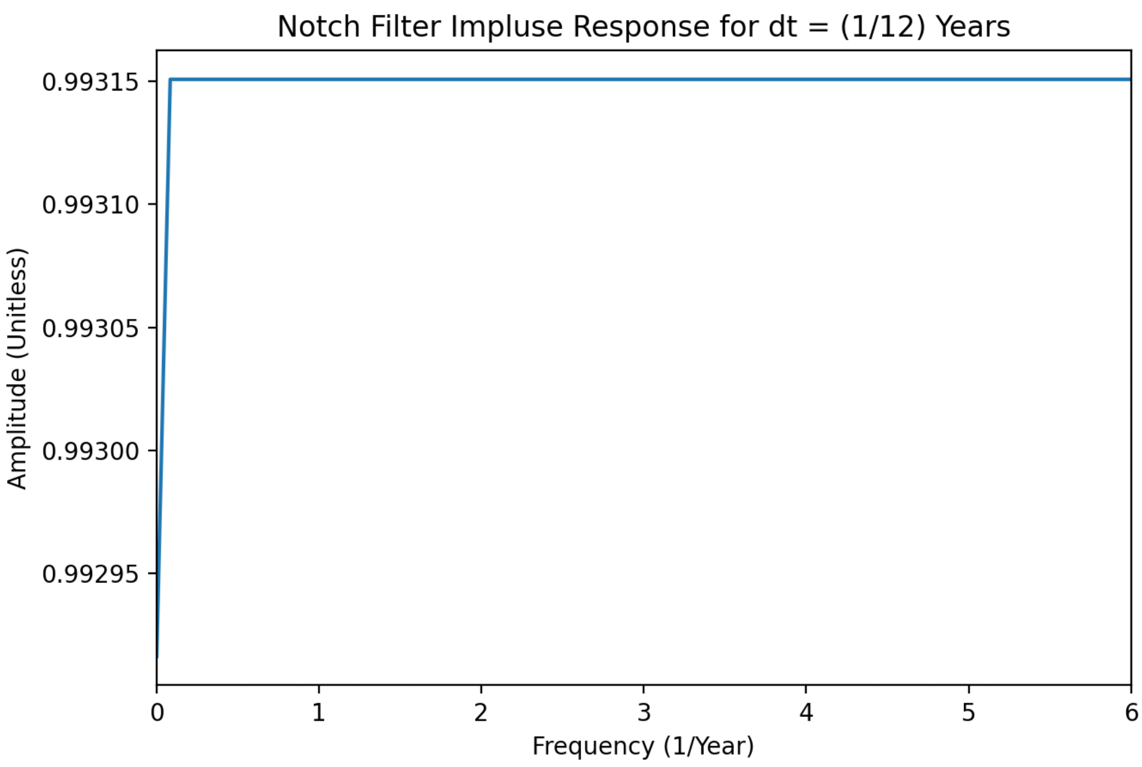
Calculating the exact values, we get:

$a = \frac{1}{1.05}, b = \frac{-1}{1.05}\left(e^{-i\frac{\pi}{2}} + e^{-i\frac{\pi}{6}}\right), c = \frac{1}{1.05}e^{-\frac{\pi}{3}}, B = \frac{-1}{1.05}\left(e^{-i\frac{\pi}{2}} + e^{-i\frac{\pi}{6}}\right), C = \frac{1}{1.05^2}e^{-\frac{\pi}{3}},$

2.

```
def ratFilter(N,D,x):
    Nx = N[0]+N[1]*x+N[2]*(x**2)
    Dx = D[0]+D[1]*x+D[2]*(x**2)
    return Nx/Dx

x = np.linspace(0,100,1200)
z = np.zeros(1200)
z[0] = (1/12)
z = np.exp(-1j*z*(1/12))
e = 0.05
N = [1/(1+e),(-1/(1+e))*(np.exp(-1j*np.pi/2)+np.exp(-1j*np.pi/6)),(1/(1+e))*np.exp(-np.pi/3)]
D = [1,(-1/(1+e))*(np.exp(-1j*np.pi/2)+np.exp(-1j*np.pi/6)),(1/(1+e)**2)*np.exp(-np.pi/3)]

plt.plot(x,ratFilter(N,D,z))
plt.xlabel("Frequency (1/Year)")
plt.ylabel("Amplitude (Unitless)")
plt.title("Notch Filter Impluse Response for dt = (1/12) Years")
plt.xlim(0,6)
plt.show()
```

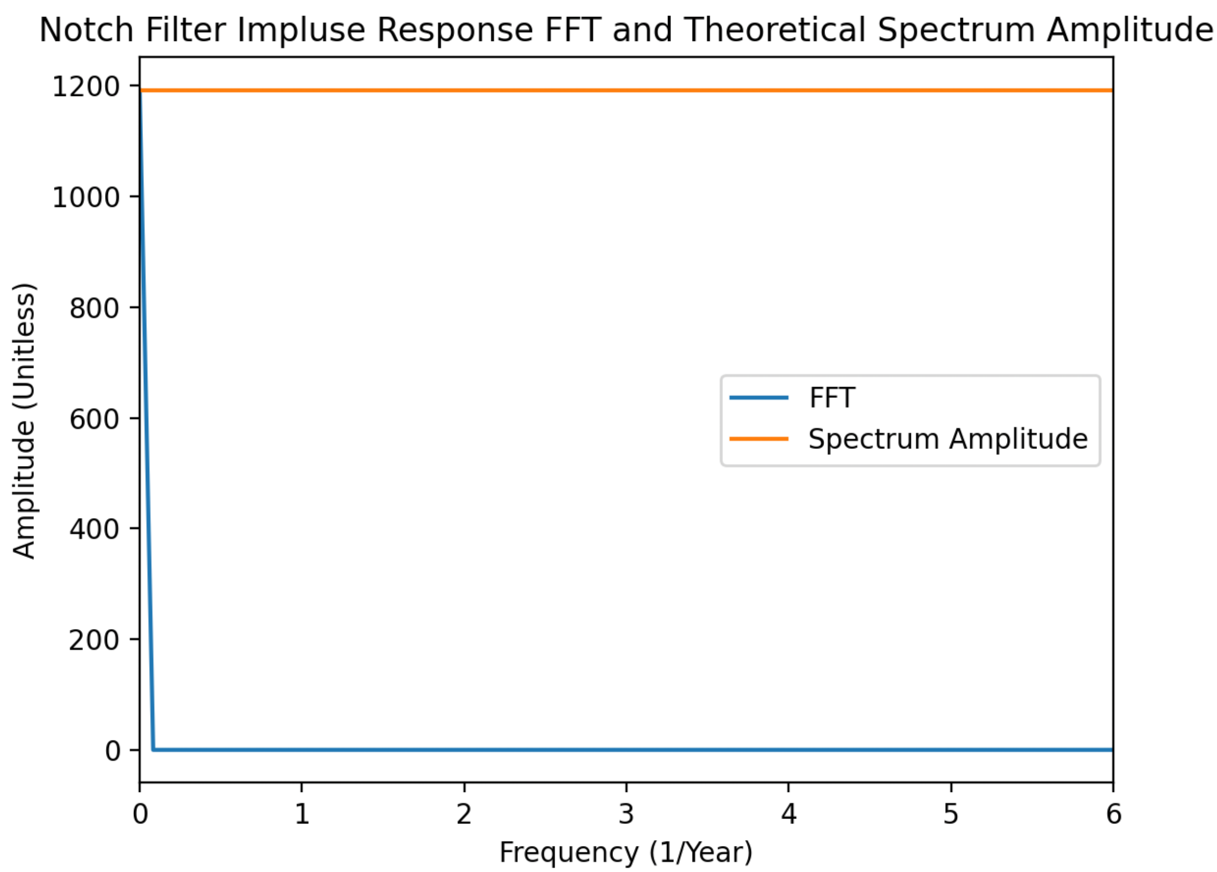
3.



Reducing the FWHM value is done by increasing the epsilon value. Doing so would result in the impulse response shifting upwards as the terms would all increase in value when $x = 0$.

4.

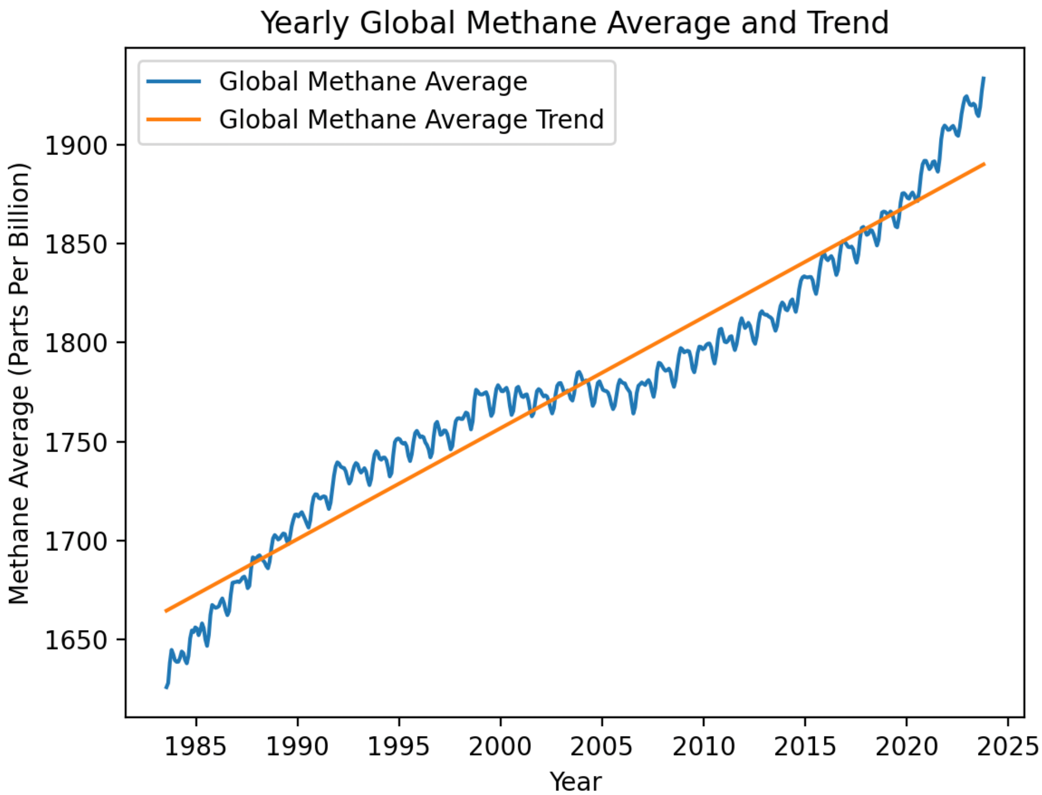
```
F = np.zeros(1200)
for i in range(len(x)):
    F[i] = np.abs(np.sum(ratFilter(N,D,z)*np.exp(-1j*x[i]*i*(1/12))))

plt.plot(x,np.fft.fft(ratFilter(N,D,z)), label = "FFT")
plt.plot(x,F, label = "Spectrum Amplitude")
plt.xlabel("Frequency (1/Year)")
plt.ylabel("Amplitude (Unitless)")
plt.title("Notch Filter Impluse Response FFT and Theoretical Spectrum Amplitude")
plt.legend()
plt.xlim(0,6)
plt.show()
```

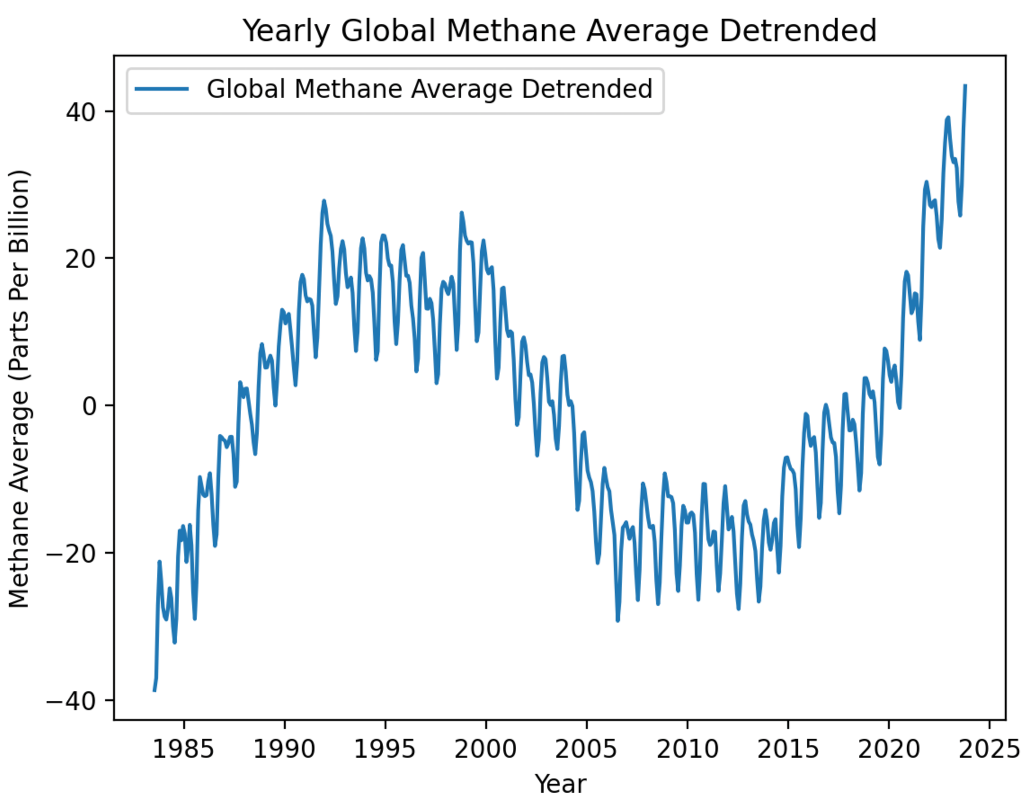


Exercise 3.

1.

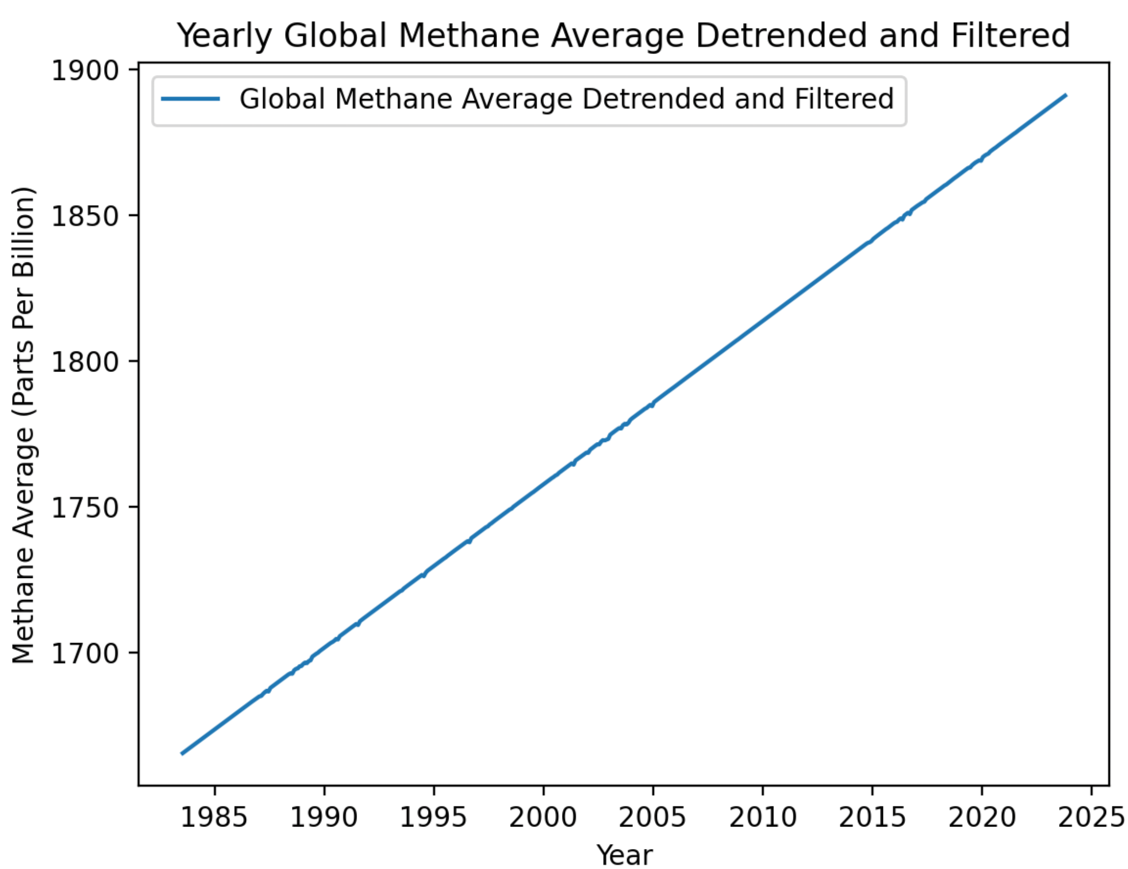
```python
file  = open("methane_global.csv",'r')
CH4 = file.read()
CH4 = CH4.split(',')

date = CH4[8:len(CH4):6]
date = [float(i) for i in date]

average = CH4[9:len(CH4):6]
average = [float(i) for i in average]

trendfit = np.polyfit(date,average,1)
trend = []
for i in range(len(date)):
    trend.append(date[i]*trendfit[0]+trendfit[1])

newaverage = []
for i in range(len(average)):
    newaverage.append(average[i]-trend[i])
plt.plot(date,average, label = "Global Methane Average")
plt.plot(date,trend, label = "Global Methane Average Trend")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average and Trend")
plt.legend()
plt.show()


plt.plot(date,newaverage, label = "Global Methane Average Detrended")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average Detrended")
plt.legend()
plt.show()
```


Yearly Global Methane Average and Trend
Global Methane Average
Global Methane Average Trend
Methane Average (Parts Per Billion)
1650
1700
1750
1800
1850
1900
1985
1990
1995
2000
2005
2010
2015
2020
2025
Year


Yearly Global Methane Average Detrended
Global Methane Average Detrended
Methane Average (Parts Per Billion)
−40
−20
0
20
40
1985
1990
1995
2000
2005
2010
2015
2020
2025
Year

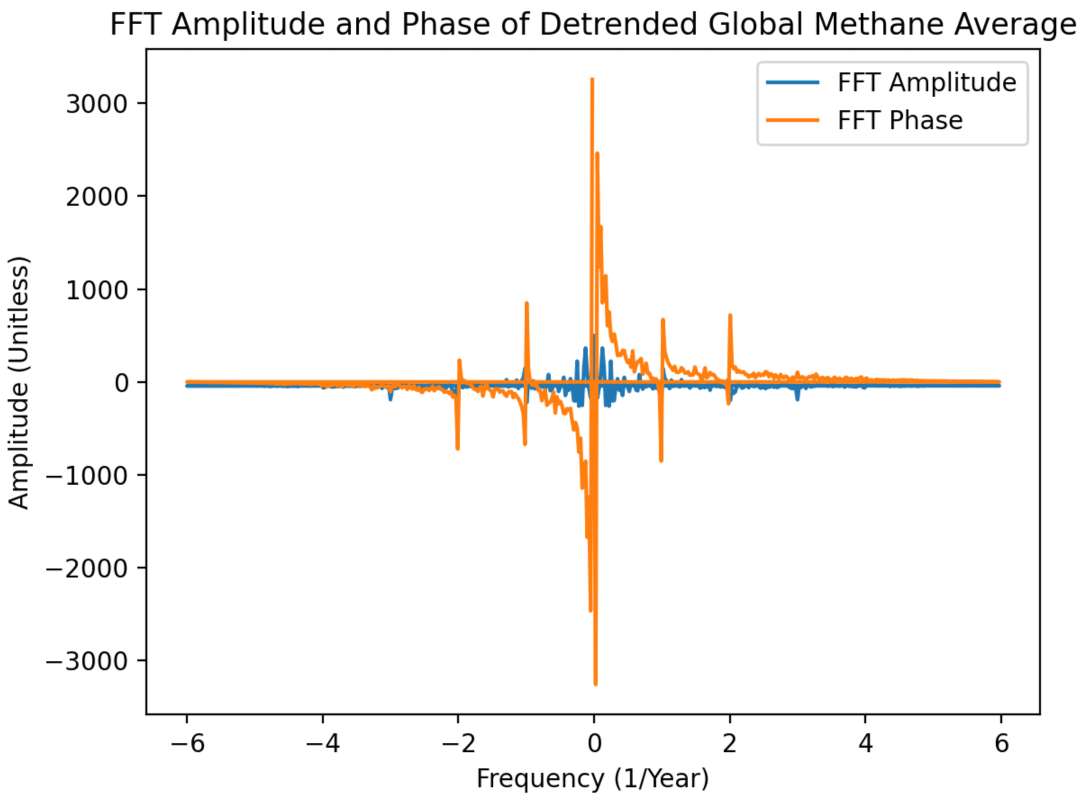
2.

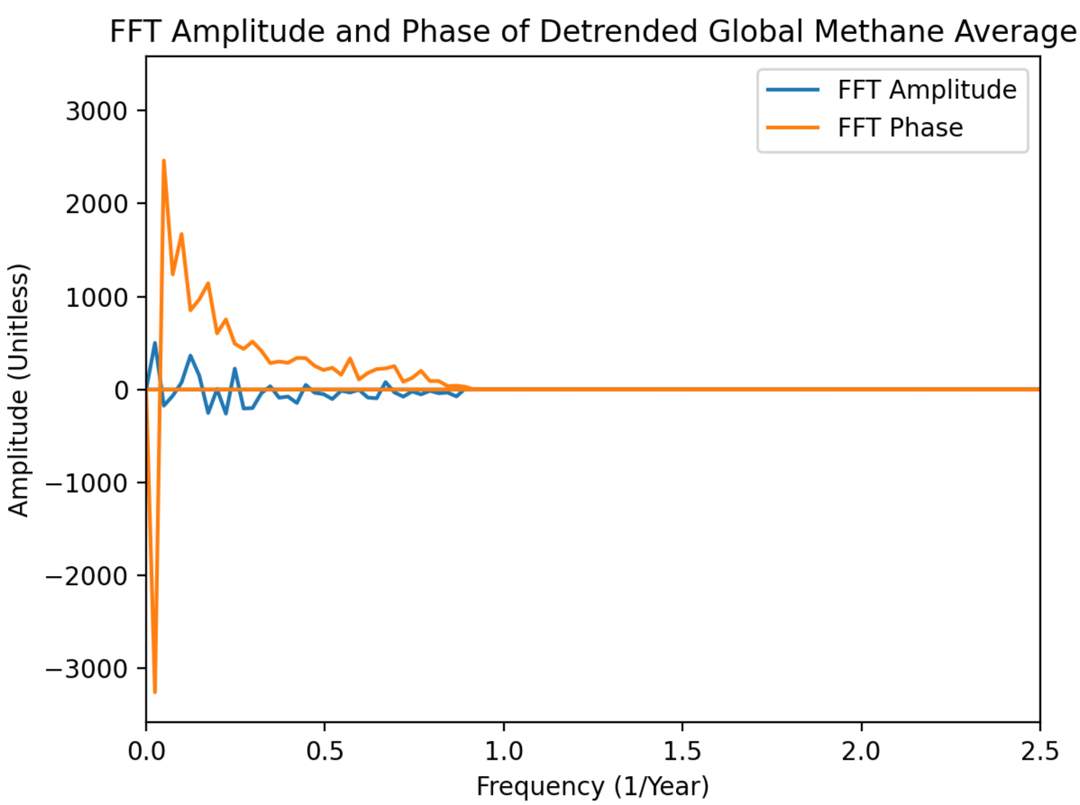
```
newaverage = np.array(newaverage)
z = np.exp(-1j*newaverage*(1/12))

def W(M,z,f0,fs,e):
    q1 = np.exp(-1j*2*np.pi*(f0)/(fs))
    q2 = np.exp(1j*2*np.pi*(f0)/(fs))
    p1 = (1+e)*q1
    p2 = (1+e)*q2
    return M*((z-q1)/(z-p1))*((z-q2)/(z-p2))

M = 1.05
f0 = 1
fs = 12
e = 0.05

# plt.plot(date,ratFilter(N,D,newaverage), label = "Global Methane Average Detrended")
plt.plot(date,W(M,z,f0,fs,e)+trend, label = "Global Methane Average Detrended and Filtered")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average Detrended and Filtered")
plt.legend()
plt.show()
```



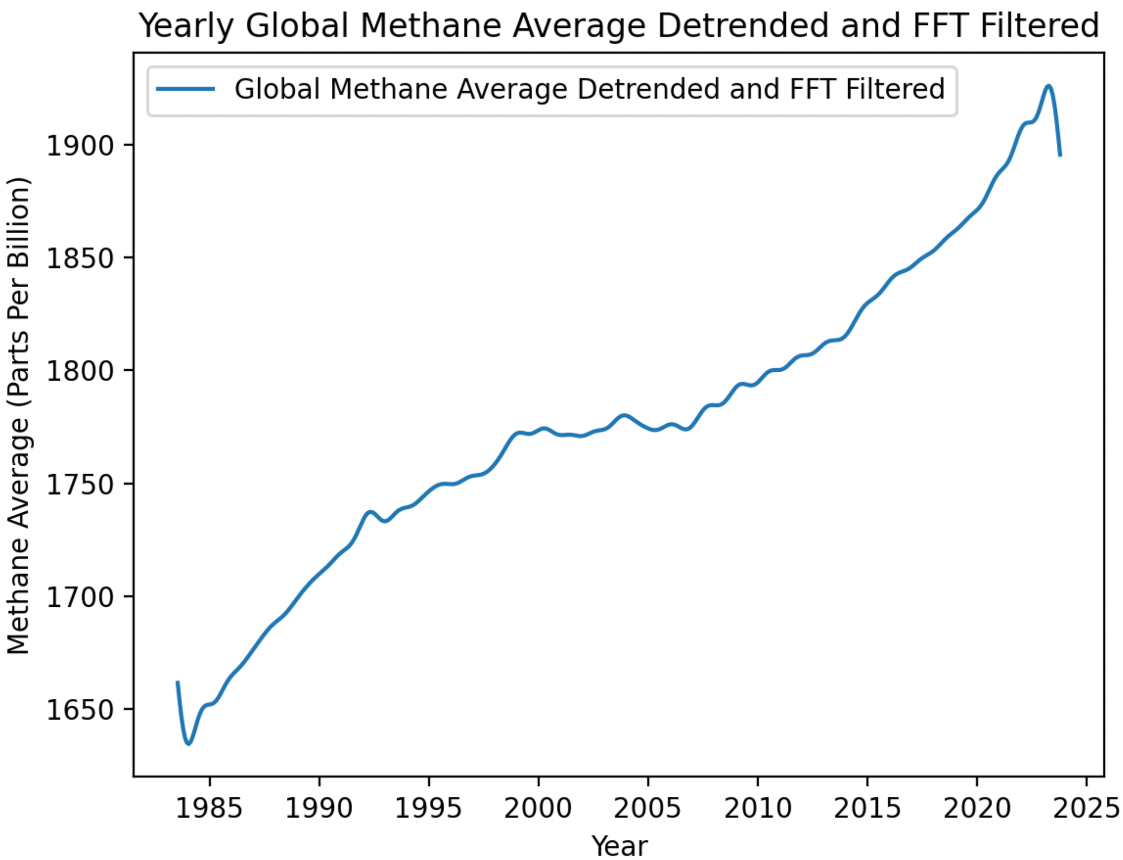
3.

```
fftnewaverage = np.fft.fft(newaverage)
length = len(date)
freq = np.fft.fftfreq(length, d=(1/12))
plt.plot(freq,np.real(fftnewaverage), label = "FFT Amplitude")
plt.plot(freq,np.imag(fftnewaverage), label = "FFT Phase")
plt.xlabel("Frequency (1/Year)")
plt.ylabel("Amplitude (Unitless)")
plt.title("FFT Amplitude and Phase of Detrended Global Methane Average")
plt.legend()
plt.show()

for i in range(len(freq)):
    if freq[i] < -0.9:
        fftnewaverage[i] = 0
    if freq[i] > 0.9:
        fftnewaverage[i] = 0

plt.plot(freq,np.real(fftnewaverage), label = "FFT Amplitude")
plt.plot(freq,np.imag(fftnewaverage), label = "FFT Phase")
plt.xlabel("Frequency (1/Year)")
plt.ylabel("Amplitude (Unitless)")
plt.title("FFT Amplitude and Phase of Detrended Global Methane Average")
plt.legend()
plt.xlim(0,2.5)
plt.show()
```


FFT Amplitude and Phase of Detrended Global Methane Average
FFT Amplitude
FFT Phase
Amplitude (Unitless)
3000
2000
1000
0
−1000
−2000
−3000
−6
−4
−2
0
2
4
6
Frequency (1/Year)

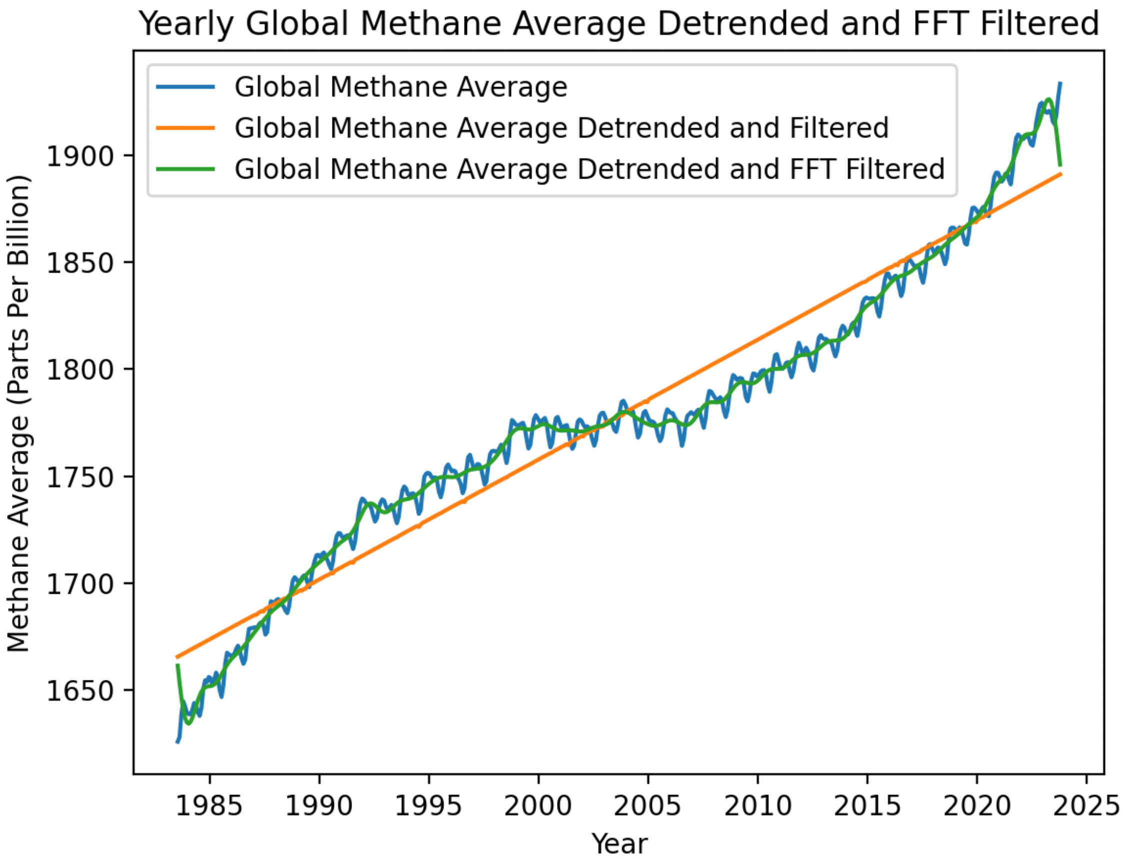

FFT Amplitude and Phase of Detrended Global Methane Average
FFT Amplitude
FFT Phase
Amplitude (Unitless)
3000
2000
1000
0
−1000
−2000
−3000
0.0
0.5
1.0
1.5
2.0
2.5
Frequency (1/Year)

```
ifftnewaverage = np.fft.ifft(fftnewaverage)

plt.plot(date,ifftnewaverage+trend, label = "Global Methane Average Detrended and FFT Filtered")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average Detrended and FFT Filtered")
plt.legend()
plt.show()
```



4.

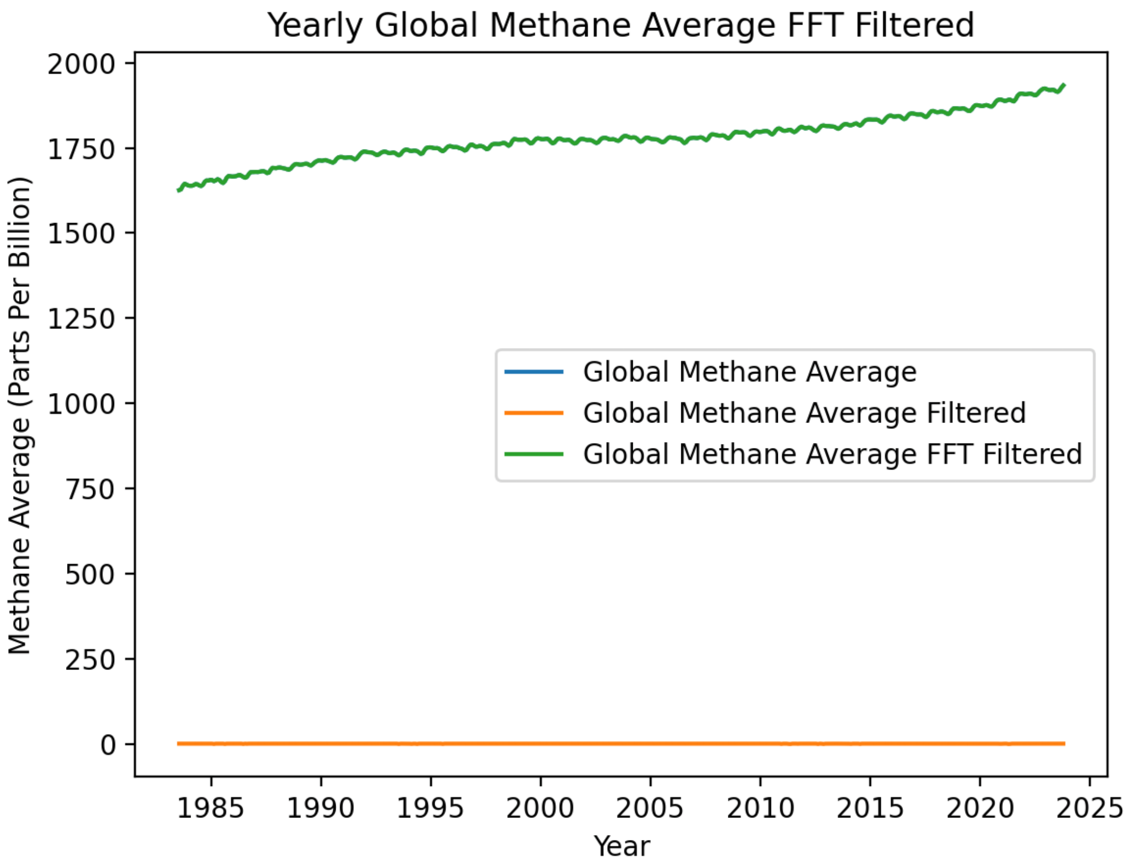
```
plt.plot(date,average, label = "Global Methane Average")
plt.plot(date,W(M,z,f0,fs,e)+trend, label = "Global Methane Average Detrended and Filtered")
plt.plot(date,ifftnewaverage+trend, label = "Global Methane Average Detrended and FFT Filtered")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average Detrended and FFT Filtered")
plt.legend()
plt.show()
```



The FFT filtering method seems to be the best balance of retaining data and filtering out the annual effects. The notch filter seems to be a very strong filter that almost matches the trend exactly, which may be useful in some cases but could lose some information.

5.

```python
average = np.array(average)
z = np.exp(-1j*average*(1/12))

fftaverage = np.fft.fft(average)
length = len(date)
freq = np.fft.fftfreq(length, d=(1/12))

for i in range(len(freq)):
    if freq[i] < -0.9:
        fftnewaverage[i] = 0
    if freq[i] > 0.9:
        fftnewaverage[i] = 0

ifftaverage = np.fft.ifft(fftaverage)

plt.plot(date,average, label = "Global Methane Average")
plt.plot(date,W(M,z,f0,fs,e), label = "Global Methane Average Filtered")
plt.plot(date,ifftaverage, label = "Global Methane Average FFT Filtered")
plt.xlabel("Year")
plt.ylabel("Methane Average (Parts Per Billion)")
plt.title("Yearly Global Methane Average FFT Filtered")
plt.legend()
plt.show()
```



It appears that the trend seems to dominate the data and prevent the filtering or FFT process from meaningfully impacting the difference.