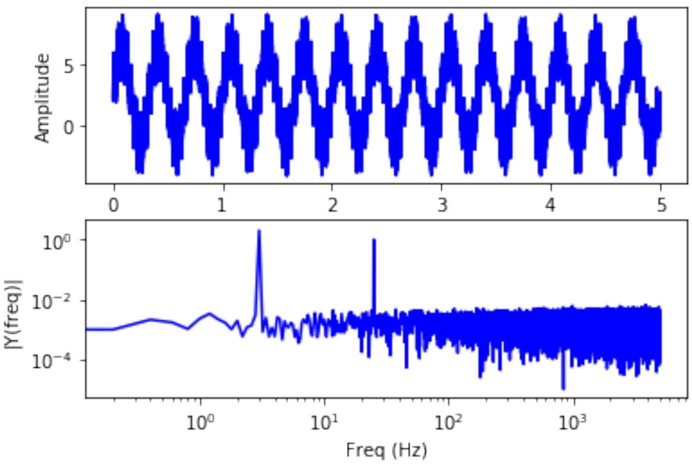
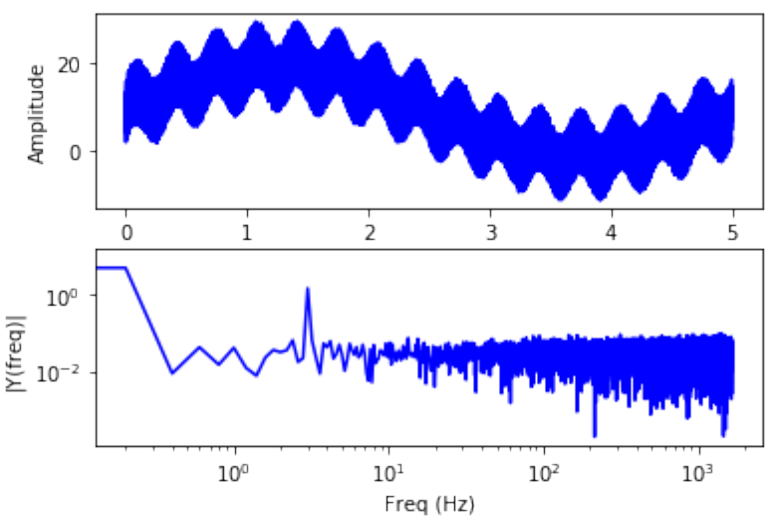
HW 2: Digital Signal Processing

4)

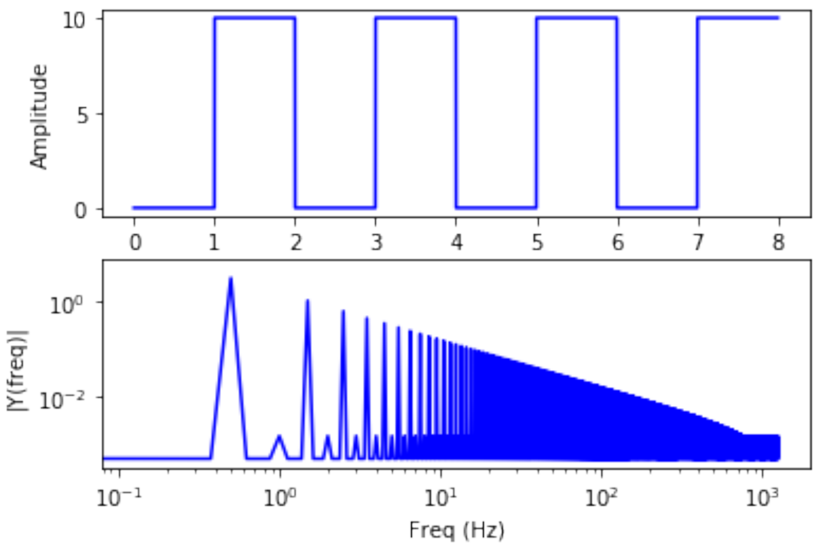
sigA:



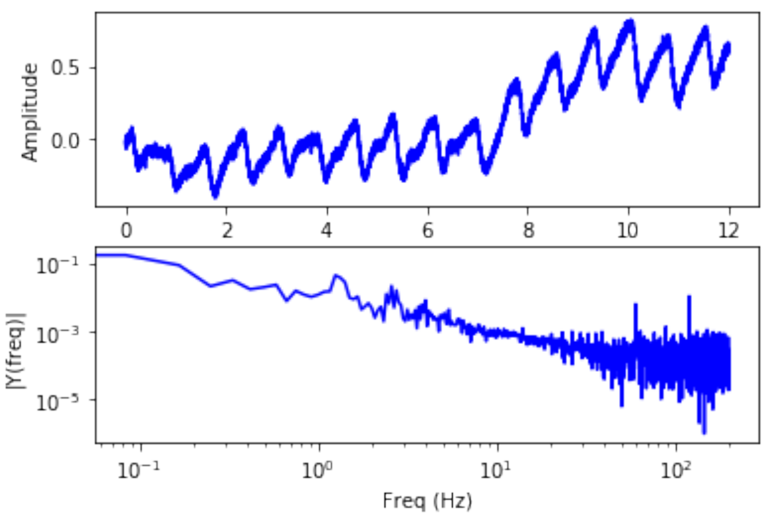
sigB:



sigC:



sigD:

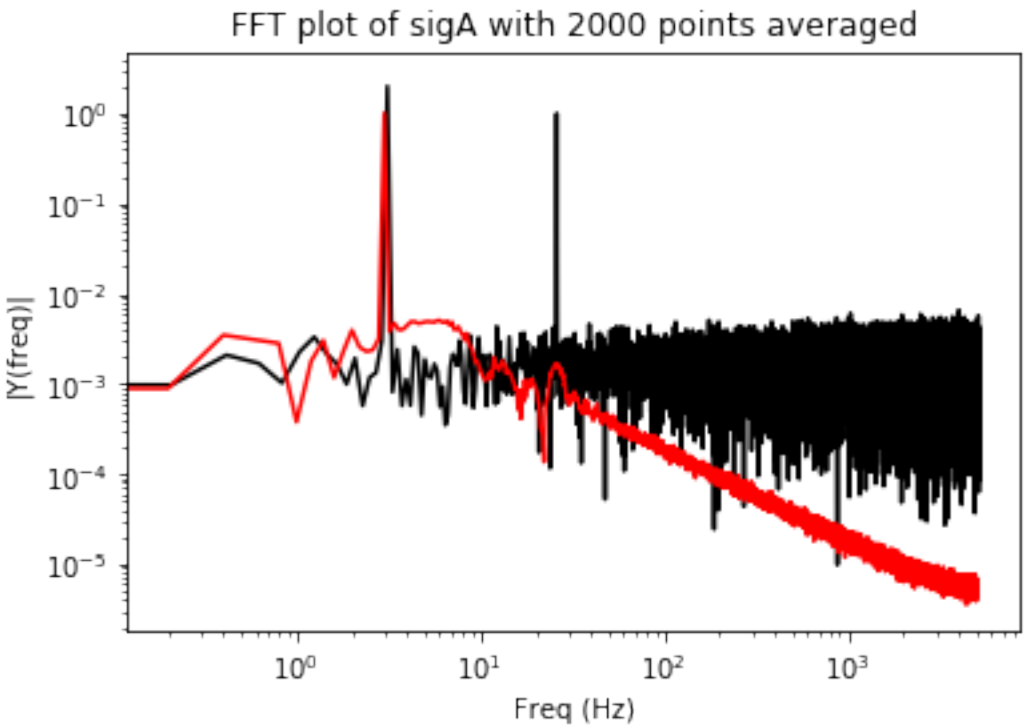


Code (same for all cases other than “sigA” being replaced with “sigB,” “sigC,” “sigD”):

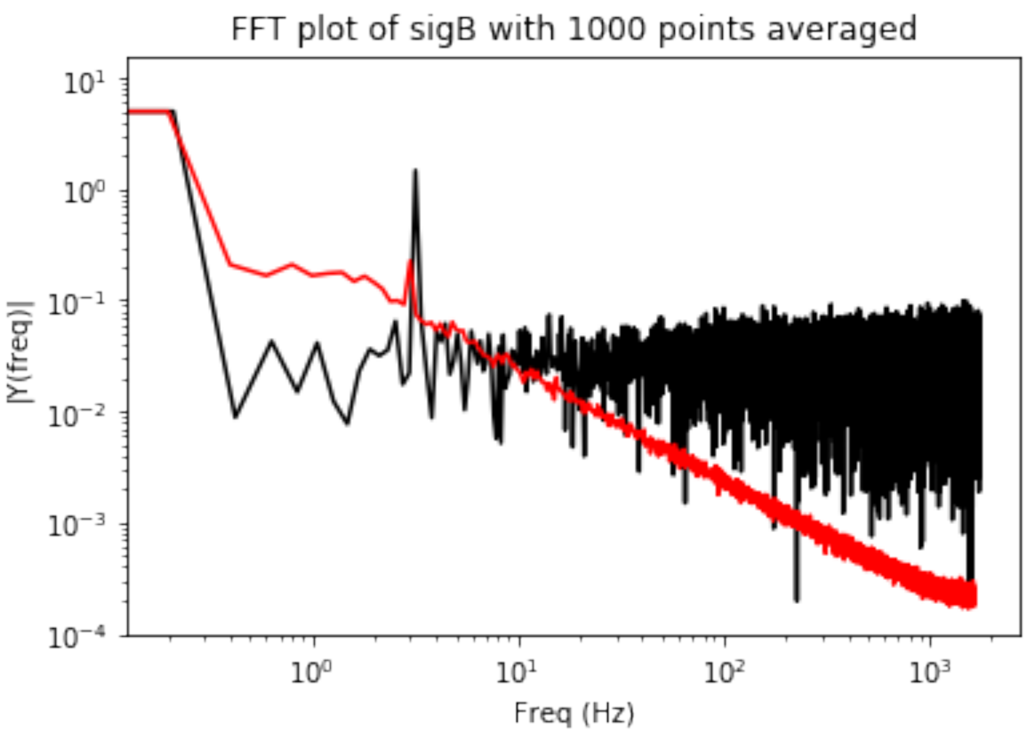
```
# FFT for sigA

t = [] # column 0
data1 = [] # column 1
data2 = [] # column 2

with open('sigA.csv') as f:
    # open the csv file
    reader = csv.reader(f)
    for row in reader:
        # read the rows 1 one by one
        t.append(float(row[0])) # leftmost column
```

```
    data1.append(float(row[1])) # second column

plt.plot(t, data1)

samplerate = len(data1)/t[-1]
print(samplerate)
print(len(data1))

# FFT for sigA

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = data1 # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]

fig, (ax1, ax2) = plt.subplots(2, 1)
ax1.plot(t,y,'b')
ax1.set_xlabel('Time')
ax1.set_ylabel('Amplitude')
ax2.loglog(frq,abs(Y),'b') # plotting the fft
ax2.set_xlabel('Freq (Hz)')
ax2.set_ylabel('|Y(freq)|')
plt.show()
```

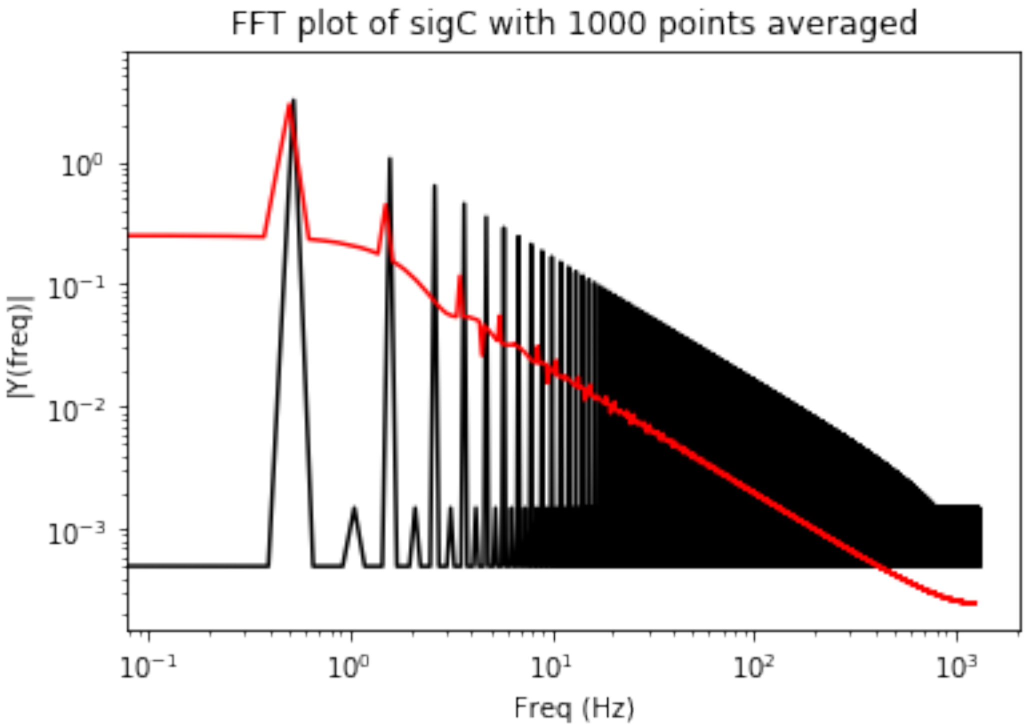
5)

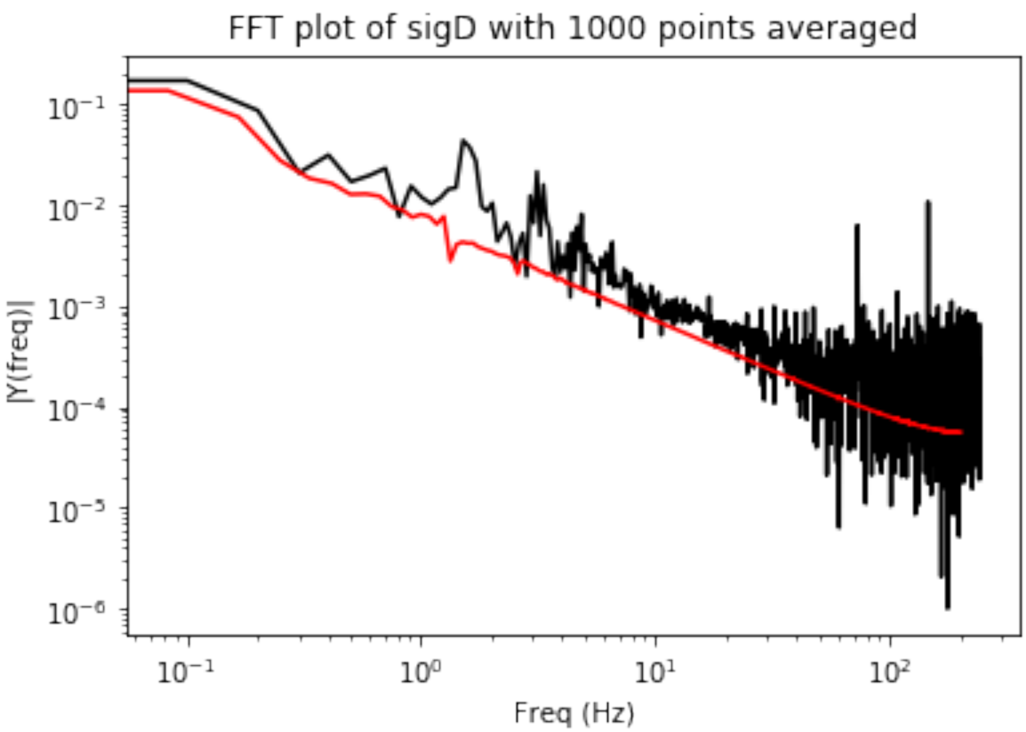


FFT plot of sigA with 2000 points averaged
|Y(freq)|
Freq (Hz)



FFT plot of sigB with 1000 points averaged
|Y(freq)|
Freq (Hz)





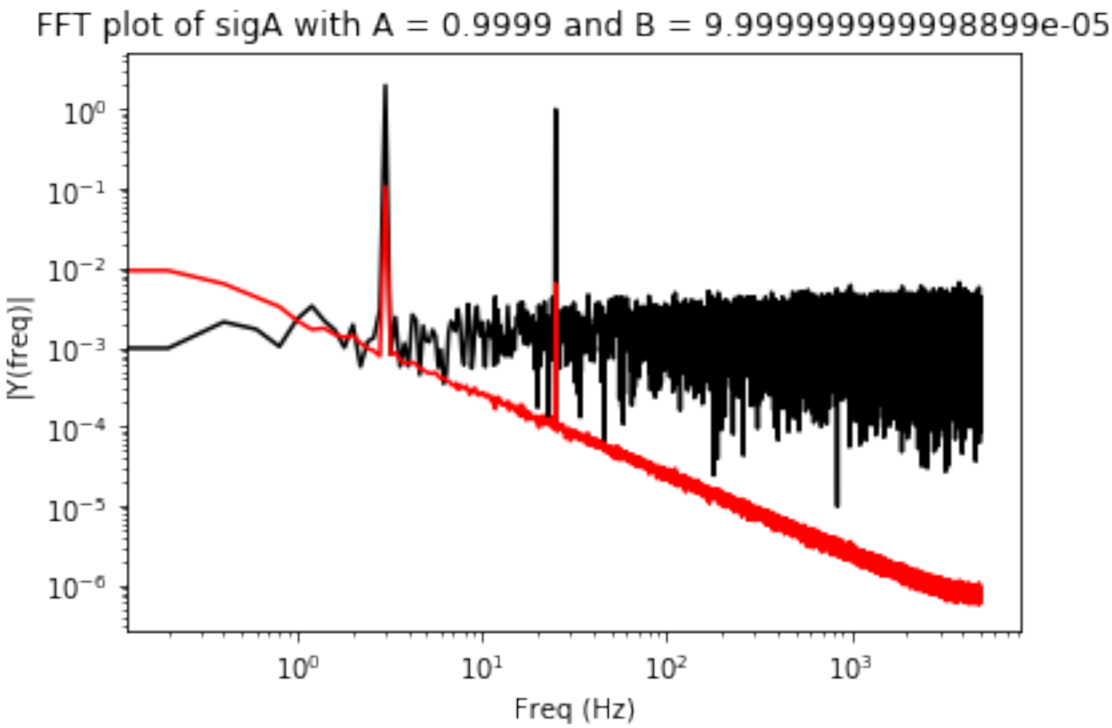
Code (same for all cases other than “sigA” being replaced with “sigB,” “sigC,” “sigD”):

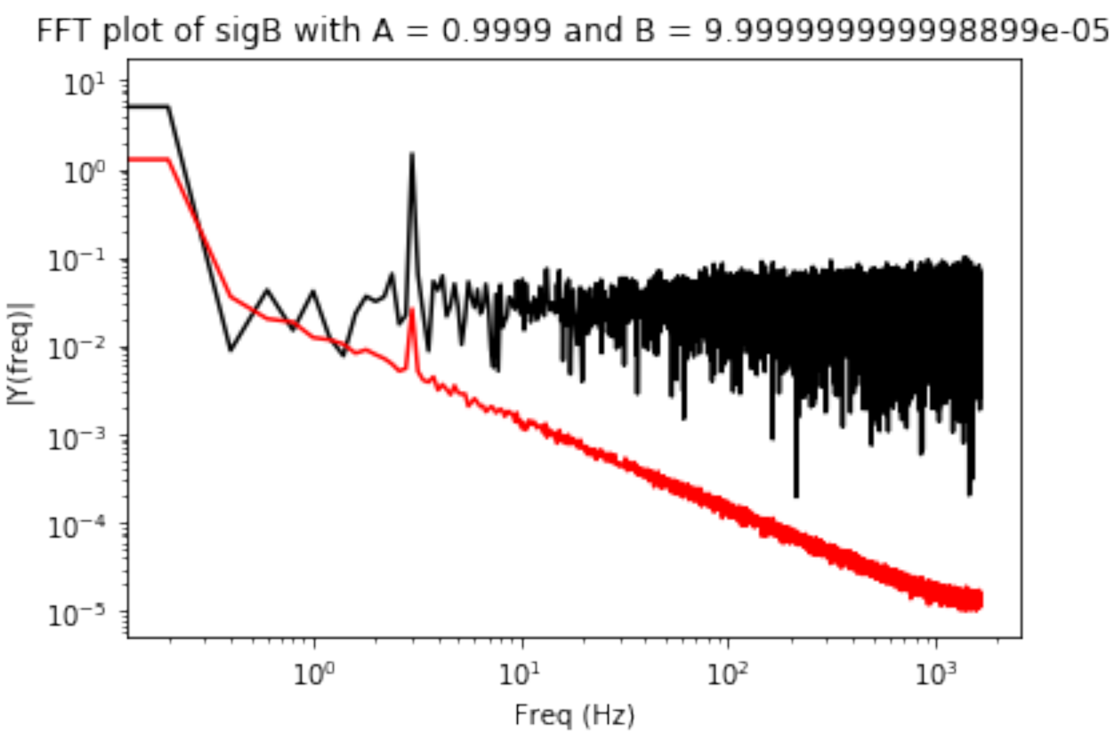
```
# Low pass filter sigA
```

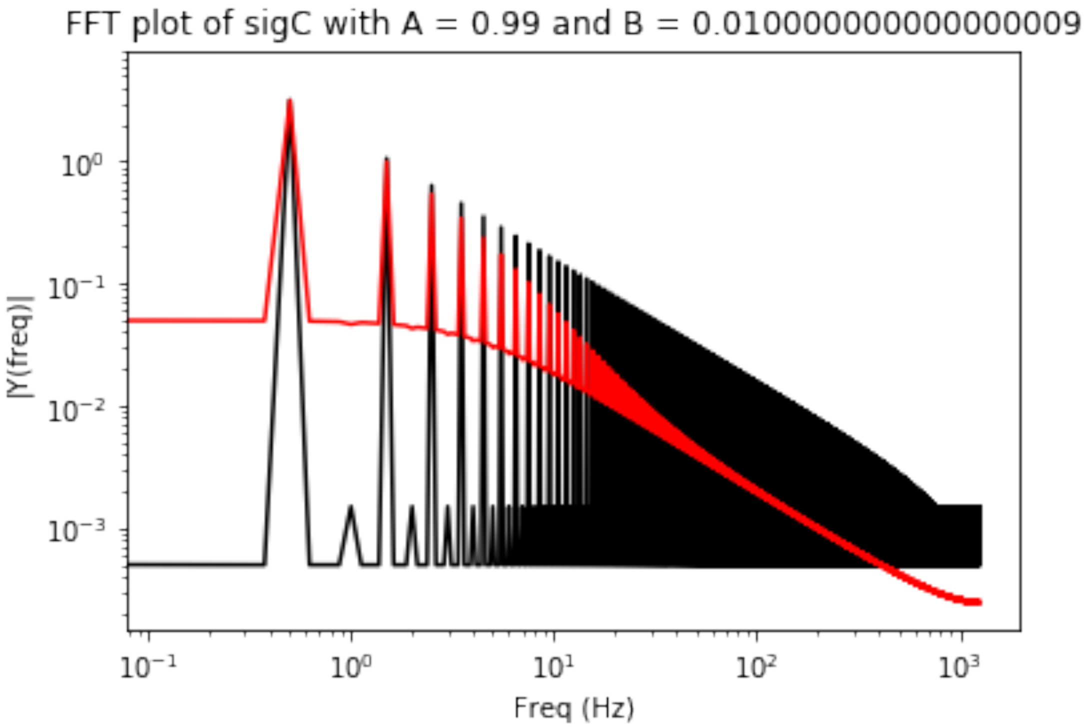
```
t = [] # column 0
data1 = [] # column 1

with open('sigA.csv') as f:
    # open the csv file
    reader = csv.reader(f)
    for row in reader:
        # read the rows 1 one by one
        t.append(float(row[0])) # leftmost column
        data1.append(float(row[1])) # second column

data = data1.copy()
x = 2000
newdata = []

for i in range(x):
    data.insert(i, 0)
for j in range(len(data)-x):
    sum = 0
    for i in range(x):
        sum = sum + data[j + x - i]
    avg = sum/x
    newdata.append(avg)

samplerate = len(data)/t[-1]
newsamplerate = len(newdata)/t[-1]


# FFT for sigA

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = data1 # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]
```

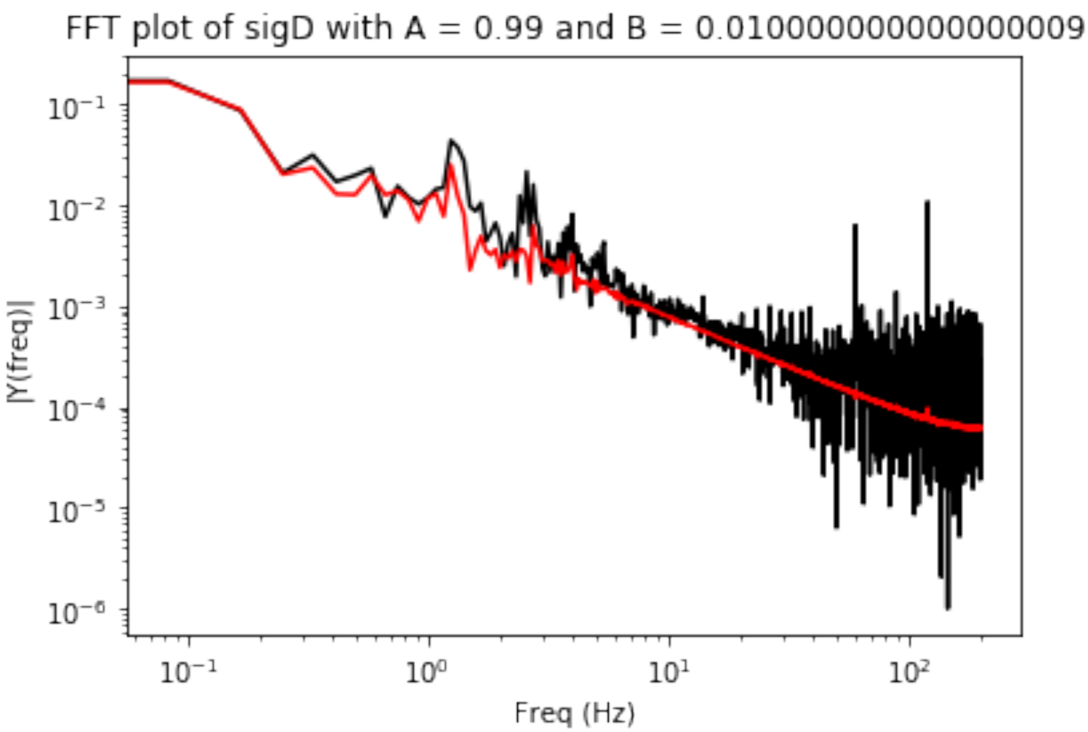
```
plt.loglog(frq,abs(Y),'black') # plotting the fft
plt.xlabel('Freq (Hz)')
plt.ylabel('|Y(freq)|')

# plt.plot(t, data1)
# plt.plot(t, newdata)


# plotting the new sample FFT
samplerate = len(newdata)/t[-1]

# FFT for sigA

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = newdata # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]


plt.loglog(frq,abs(Y),'red') # plotting the fft
plt.title('FFT plot of sigA with ' + str(x) + ' points averaged')
plt.show()
```

6)


FFT plot of sigA with A = 0.9999 and B = 9.999999999998899e-05
|Y(freq)|
Freq (Hz)


FFT plot of sigB with A = 0.9999 and B = 9.999999999998899e-05
|Y(freq)|
Freq (Hz)





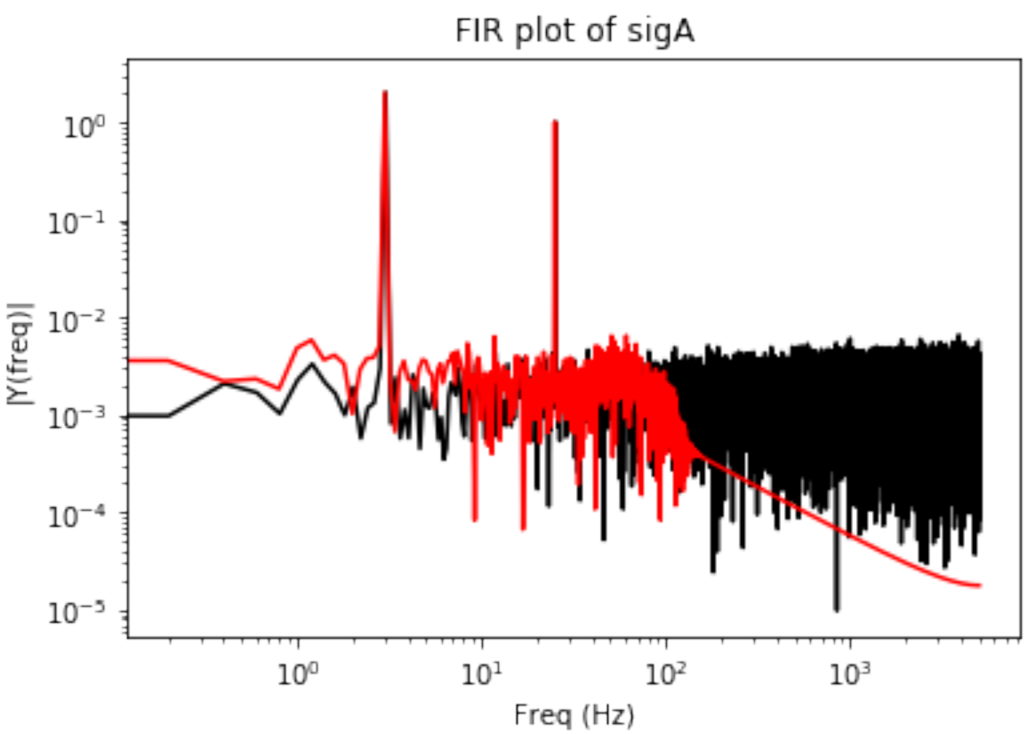
Code (same for all cases other than “sigA” being replaced with “sigB,” “sigC,” “sigD”):

```
# Low pass filtering with an IIR for sigA

t = [] # column 0
```

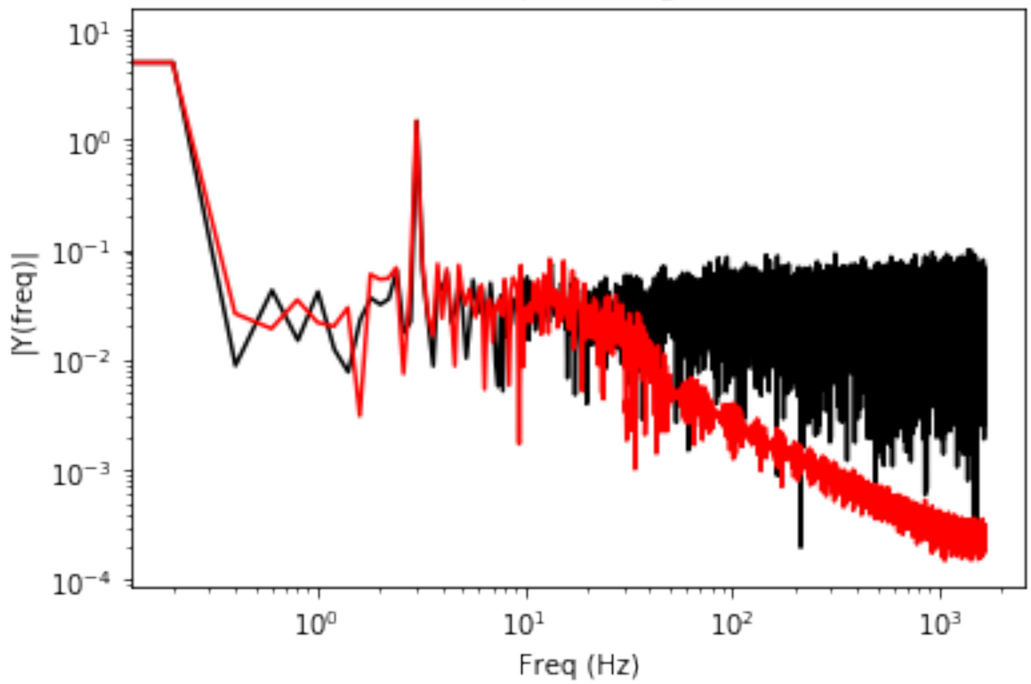
```
data1 = [] # column 1

with open('sigA.csv') as f:
    # open the csv file
    reader = csv.reader(f)
    for row in reader:
        # read the rows 1 one by one
        t.append(float(row[0])) # leftmost column
        data1.append(float(row[1])) # second column

data = data1.copy()
A = 0.9999
B = 1-A
newdata = []
newdata.append(data[0])

for i in range(1, len(data)):
    newdata.append(A*newdata[i-1] + B*data[i])

samplerate = len(data)/t[-1]
newsamplerate = len(newdata)/t[-1]


# FFT for sigA

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = data1 # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]


plt.loglog(frq,abs(Y),'black') # plotting the fft
plt.xlabel('Freq (Hz)')
plt.ylabel('|Y(freq)|')


# plotting the new sample FFT
```

```
samplerate = len(newdata)/t[-1]

# FFT for sigA

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = newdata # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]


plt.loglog(frq,abs(Y),'red') # plotting the fft
plt.title('FFT plot of sigA with A = ' + str(A) + ' and B = '+ str(B)+ '')
plt.show()
```

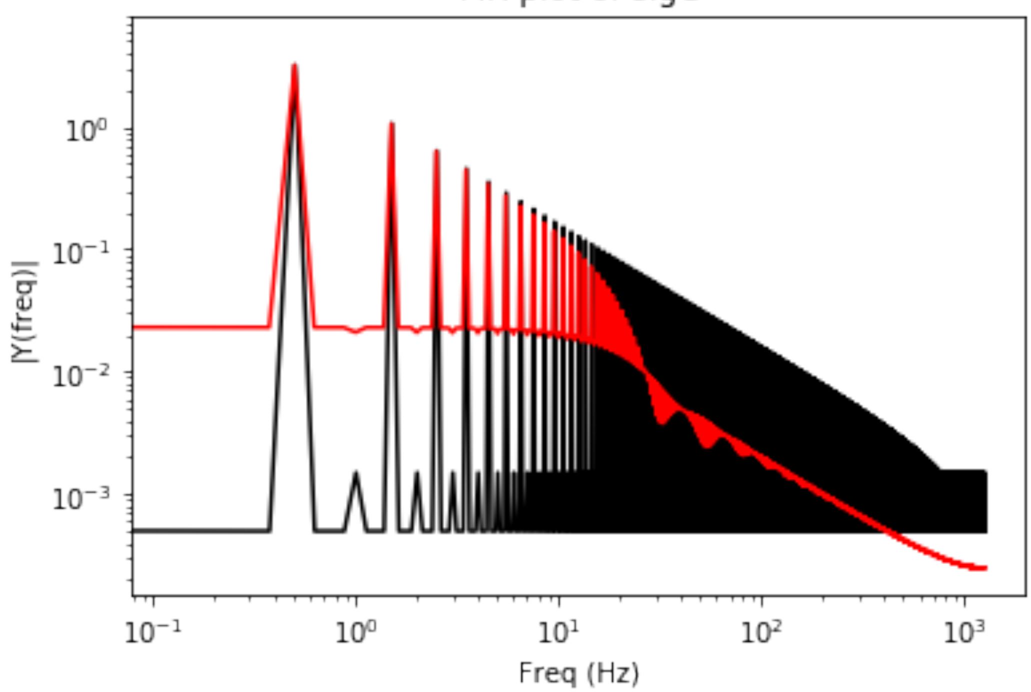
7)

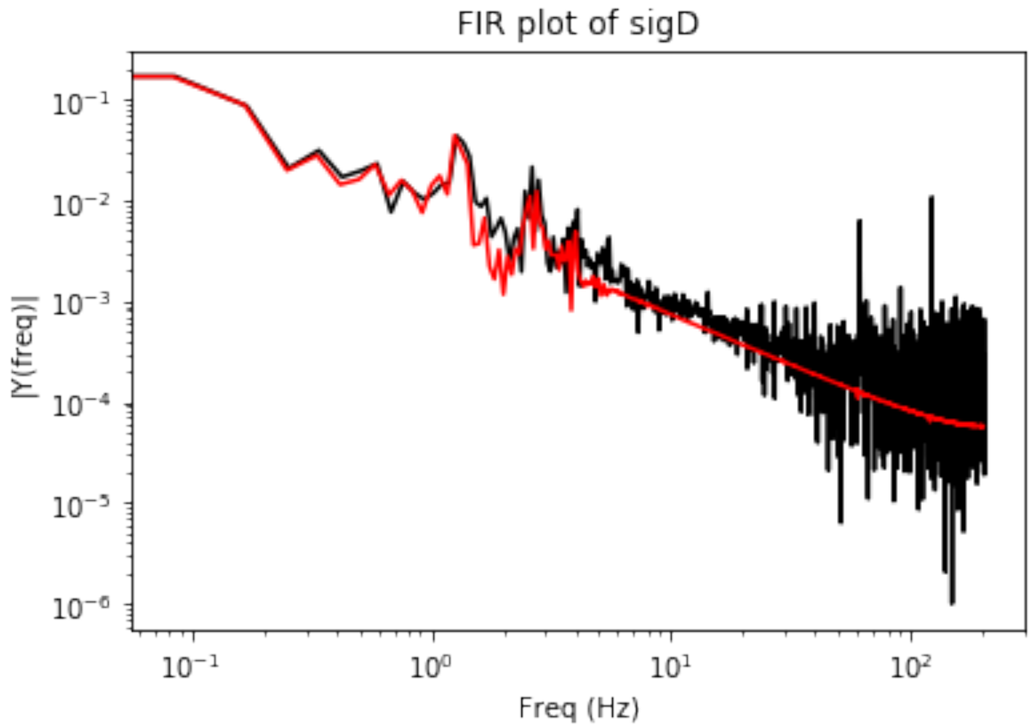


FIR plot of sigB



FIR plot of sigC





Code (same for all cases other than “sigA” being replaced with “sigB,” “sigC,” “sigD”):

NOTE: FOR CLARITY, WEIGHTS HAVE BEEN REMOVED AND REPLACED WITH “**……”** ACTUAL WEIGHTS ARE LISTED BELOW FOR EACH DATA SAMPLE

```
# Low pass filter sigA

t = [] # column 0
data1 = [] # column 1

with open('sigA.csv') as f:
   # open the csv file
   reader = csv.reader(f)
   for row in reader:
      # read the rows 1 one by one
      t.append(float(row[0])) # leftmost column
      data1.append(float(row[1])) # second column

weights = ……
data = data1.copy()
x = len(weights)
newdata = []
```

```
for i in range(x):
   data.insert(i, 0)
for j in range(len(data)-x):
   sum = 0
   for i in range(x):
      sum = sum + weights[i]*data[j + x - i]
   newdata.append(sum)

samplerate = len(data)/t[-1]
newsamplerate = len(newdata)/t[-1]

# plt.plot(t,data1)
# plt.plot(t,newdata)

# FFT for sigD

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = data1 # the data to make the fft from
n = len(y) # length of the signal
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]


plt.loglog(frq,abs(Y),'black') # plotting the fft
plt.xlabel('Freq (Hz)')
plt.ylabel('|Y(freq)|')


# plotting the new sample FFT
samplerate = len(newdata)/t[-1]

# FFT for sigD

Fs = samplerate # sample rate
Ts = 1.0/Fs; # sampling interval
ts = np.arange(0,t[-1],Ts) # time vector
y = newdata # the data to make the fft from
n = len(y) # length of the signal
```

```
k = np.arange(n)
T = n/Fs
frq = k/T # two sides frequency range
frq = frq[range(int(n/2))] # one side frequency range
Y = np.fft.fft(y)/n # fft computing and normalization
Y = Y[range(int(n/2))]


plt.loglog(frq,abs(Y),'red') # plotting the fft
plt.title('FIR plot of sigA')
plt.show()
```

Weights for sigA:

```
weights = [
    0.000000000000000000,
    0.000000022609304456,
    0.000000092137658127,
    0.000000210376981500,
    0.000000378074907886,
    0.000000594900325696,
    0.000000859416815111,
    0.000001169064300580,
    0.000001520149215920,
    0.000001907843452345,
    0.000002326192331196,
    0.000002768131812358,
    0.000003225515116128,
    0.000003689148900478,
    0.000004148839097111,
    0.000004593446468231,
    0.000005010951901591,
    0.000005388531413869,
    0.000005712640781930,
    0.000005969109667889,
    0.000006143245047260,
    0.000006219943689868,
    0.000006183813380806,
    0.000006019302503680,
    0.000005710837540961,
    0.000005242967976715,
    0.000004600518015692,
    0.000003768744460037,
    0.000002733500011260,
```

0.000001481401190987,
0.000000000000000000,
-0.000001722041638279,
-0.000003694775675826,
-0.000005926791075647,
-0.000008425037448593,
-0.000011194650419114,
-0.000014238780007746,
-0.000017558423373286,
-0.000021152263307200,
-0.000025016513915877,
-0.000029144774962341,
-0.000033527896366957,
-0.000038153854386003,
-0.000043007640996822,
-0.000048071168018069,
-0.000053323187482591,
-0.000058739229758206,
-0.000064291560877524,
-0.000069949160491469,
-0.000075677721802043,
-0.000081439674757675,
-0.000087194233709117,
-0.000092897470625088,
-0.000098502414854722,
-0.000103959180298482,
-0.000109215120710603,
-0.000114215013704846,
-0.000118901273871578,
-0.000123214195238642,
-0.000127092223121715,
-0.000130472255212696,
-0.000133289971547970,
-0.000135480192783221,
-0.000136977265978890,
-0.000137715476871664,
-0.000137629487373880,
-0.000136654796805819,
-0.000134728225127145,
-0.000131788416194798,
-0.000127776358837126,
-0.000122635923299772,
-0.000116314410389517,
-0.000108763110419883,

-0.000099937868848572,
-0.000089799655293773,
-0.000078315132425799,
-0.000065457221054363,
-0.000051205657571931,
-0.000035547539771790,
-0.000018477856937501,
0.000000000000000000,
0.000019873752518615,
0.000041121777108205,
0.000063712680780720,
0.000087604913846630,
0.000112746424287725,
0.000139074357997619,
0.000166514809072417,
0.000194982624214208,
0.000224381265158973,
0.000254602732857888,
0.000285527556926846,
0.000317024853633430,
0.000348952455414073,
0.000381157114607204,
0.000413474783751794,
0.000445730974435899,
0.000477741196287834,
0.000509311477285098,
0.000540238966114795,
0.000570312616856111,
0.000599313955772492,
0.000627017929501001,
0.000653193833411438,
0.000677606318380946,
0.000700016473693936,
0.000720182983235291,
0.000737863351600188,
0.000752815196199786,
0.000764797600901996,
0.000773572526213876,
0.000778906270490711,
0.000780570976149873,
0.000778346174378884,
0.000772020361360302,
0.000761392598594556,
0.000746274129489374,

0.000726490004004188,
0.000701880702793345,
0.000672303751986135,
0.000637635319477743,
0.000597771783385927,
0.000552631263156265,
0.000502155103676596,
0.000446309302690953,
0.000385085871786737,
0.000318504121267832,
0.000246611859321968,
0.000169486496044106,
0.000087236043089405,
0.000000000000000000,
-0.000092049881420886,
-0.000188708962926191,
-0.000289739223627016,
-0.000394868909070879,
-0.000503792332330792,
-0.000616169833229712,
-0.000731627901163730,
-0.000849759466277988,
-0.000970124362994252,
-0.001092249969091942,
-0.001215632022708469,
-0.001339735618754229,
-0.001463996385336302,
-0.001587821839857698,
-0.001710592923510197,
-0.001831665711913682,
-0.001950373298678269,
-0.002066027847682981,
-0.002177922808881408,
-0.002285335291466635,
-0.002387528587259935,
-0.002483754836236344,
-0.002573257825170752,
-0.002655275909486315,
-0.002729045047518393,
-0.002793801935577453,
-0.002848787231408712,
-0.002893248852910153,
-0.002926445338288832,
-0.002947649253212949,

-0.002956150629958684,
-0.002951260423060331,
-0.002932313965553967,
-0.002898674409562282,
-0.002849736134704403,
-0.002784928107632618,
-0.002703717175899989,
-0.002605611279351113,
-0.002490162562303985,
-0.002356970369955443,
-0.002205684112696052,
-0.002036005982363005,
-0.001847693504890926,
-0.001640561914339420,
-0.001414486333881437,
-0.001169403750025814,
-0.000905314767118358,
-0.000622285130015536,
-0.000320447003749589,
0.000000000000000001,
0.000338788058750984,
0.000695580645376159,
0.001069972311390398,
0.001461488752950066,
0.001869587163870110,
0.002293656878647479,
0.002733020307130437,
0.003186934161098913,
0.003654590971628138,
0.004135120894703703,
0.004627593801148187,
0.005131021645515233,
0.005644361107213516,
0.006166516495748513,
0.006696342910621199,
0.007232649645107787,
0.007774203821870057,
0.008319734247119286,
0.008867935468884999,
0.009417472023829232,
0.009966982856004428,
0.010515085889984220,
0.011060382739907122,
0.011601463535168466,

0.012136911842781345,
0.012665309665806283,
0.013185242496726660,
0.013695304404225379,
0.014194103131501134,
0.014680265184051732,
0.015152440884749678,
0.015609309374042002,
0.016049583533223350,
0.016472014808958144,
0.016875397917563813,
0.017258575408011220,
0.017620442063148847,
0.017959949119311452,
0.018276108285228888,
0.018567995542002971,
0.018834754706865513,
0.019075600744464558,
0.019289822810542985,
0.019476787014068648,
0.019635938885141990,
0.019766805537339234,
0.019868997514539938,
0.019942210313729766,
0.019986225576755702,
0.020000911945533145,
0.019986225576755702,
0.019942210313729766,
0.019868997514539938,
0.019766805537339234,
0.019635938885141990,
0.019476787014068648,
0.019289822810542985,
0.019075600744464561,
0.018834754706865513,
0.018567995542002971,
0.018276108285228892,
0.017959949119311456,
0.017620442063148851,
0.017258575408011220,
0.016875397917563817,
0.016472014808958144,
0.016049583533223350,
0.015609309374042002,

0.015152440884749678,
0.014680265184051732,
0.014194103131501134,
0.013695304404225379,
0.013185242496726660,
0.012665309665806286,
0.012136911842781345,
0.011601463535168468,
0.011060382739907124,
0.010515085889984224,
0.009966982856004430,
0.009417472023829233,
0.008867935468885001,
0.008319734247119286,
0.007774203821870058,
0.007232649645107790,
0.006696342910621201,
0.006166516495748514,
0.005644361107213516,
0.005131021645515233,
0.004627593801148190,
0.004135120894703703,
0.003654590971628138,
0.003186934161098913,
0.002733020307130436,
0.002293656878647479,
0.001869587163870110,
0.001461488752950066,
0.001069972311390398,
0.000695580645376159,
0.000338788058750984,
0.000000000000000001,
-0.000320447003749589,
-0.000622285130015536,
-0.000905314767118358,
-0.001169403750025814,
-0.001414486333881437,
-0.001640561914339420,
-0.001847693504890926,
-0.002036005982363005,
-0.002205684112696052,
-0.002356970369955444,
-0.002490162562303986,
-0.002605611279351114,

-0.002703717175899990,
-0.002784928107632618,
-0.002849736134704405,
-0.002898674409562284,
-0.002932313965553968,
-0.002951260423060331,
-0.002956150629958684,
-0.002947649253212949,
-0.002926445338288832,
-0.002893248852910154,
-0.002848787231408712,
-0.002793801935577453,
-0.002729045047518393,
-0.002655275909486314,
-0.002573257825170751,
-0.002483754836236344,
-0.002387528587259935,
-0.002285335291466637,
-0.002177922808881409,
-0.002066027847682983,
-0.001950373298678270,
-0.001831665711913683,
-0.001710592923510196,
-0.001587821839857699,
-0.001463996385336302,
-0.001339735618754230,
-0.001215632022708470,
-0.001092249969091943,
-0.000970124362994253,
-0.000849759466277988,
-0.000731627901163730,
-0.000616169833229712,
-0.000503792332330792,
-0.000394868909070879,
-0.000289739223627016,
-0.000188708962926191,
-0.000092049881420886,
0.000000000000000000,
0.000087236043089405,
0.000169486496044106,
0.000246611859321969,
0.000318504121267832,
0.000385085871786737,
0.000446309302690953,

0.000502155103676596,
0.000552631263156265,
0.000597771783385928,
0.000637635319477744,
0.000672303751986135,
0.000701880702793346,
0.000726490004004188,
0.000746274129489374,
0.000761392598594557,
0.000772020361360302,
0.000778346174378885,
0.000780570976149873,
0.000778906270490711,
0.000773572526213877,
0.000764797600901996,
0.000752815196199787,
0.000737863351600188,
0.000720182983235291,
0.000700016473693937,
0.000677606318380946,
0.000653193833411439,
0.000627017929501001,
0.000599313955772492,
0.000570312616856111,
0.000540238966114796,
0.000509311477285099,
0.000477741196287834,
0.000445730974435899,
0.000413474783751794,
0.000381157114607204,
0.000348952455414074,
0.000317024853633430,
0.000285527556926846,
0.000254602732857889,
0.000224381265158973,
0.000194982624214209,
0.000166514809072417,
0.000139074357997619,
0.000112746424287725,
0.000087604913846630,
0.000063712680780720,
0.000041121777108205,
0.000019873752518615,
0.000000000000000000,

-0.000018477856937501,
-0.000035547539771790,
-0.000051205657571931,
-0.000065457221054363,
-0.000078315132425799,
-0.000089799655293773,
-0.000099937868848572,
-0.000108763110419883,
-0.000116314410389517,
-0.000122635923299772,
-0.000127776358837126,
-0.000131788416194798,
-0.000134728225127145,
-0.000136654796805820,
-0.000137629487373880,
-0.000137715476871664,
-0.000136977265978890,
-0.000135480192783221,
-0.000133289971547970,
-0.000130472255212695,
-0.000127092223121715,
-0.000123214195238642,
-0.000118901273871578,
-0.000114215013704845,
-0.000109215120710603,
-0.000103959180298482,
-0.000098502414854722,
-0.000092897470625088,
-0.000087194233709117,
-0.000081439674757675,
-0.000075677721802043,
-0.000069949160491469,
-0.000064291560877524,
-0.000058739229758206,
-0.000053323187482591,
-0.000048071168018069,
-0.000043007640996822,
-0.000038153854386003,
-0.000033527896366957,
-0.000029144774962341,
-0.000025016513915877,
-0.000021152263307200,
-0.000017558423373287,
-0.000014238780007746,

```
    -0.000011194650419114,
    -0.000008425037448593,
    -0.000005926791075647,
    -0.000003694775675826,
    -0.000001722041638279,
    0.000000000000000000,
    0.000001481401190987,
    0.000002733500011260,
    0.000003768744460037,
    0.000004600518015692,
    0.000005242967976715,
    0.000005710837540961,
    0.000006019302503680,
    0.000006183813380806,
    0.000006219943689868,
    0.000006143245047260,
    0.000005969109667889,
    0.000005712640781930,
    0.000005388531413869,
    0.000005010951901591,
    0.000004593446468231,
    0.000004148839097111,
    0.000003689148900478,
    0.000003225515116128,
    0.000002768131812358,
    0.000002326192331196,
    0.000001907843452345,
    0.000001520149215920,
    0.000001169064300580,
    0.000000859416815111,
    0.000000594900325696,
    0.000000378074907886,
    0.000000210376981500,
    0.000000092137658127,
    0.000000022609304456,
    0.000000000000000000,
]
```

Weights for sigB:

```
weights = [
    0.001867519746044820,
    0.002275293677911463,
```

0.002692845467883248,
0.003119400211540369,
0.003554151391578438,
0.003996262853518041,
0.004444870878876082,
0.004899086349965267,
0.005357997000181999,
0.005820669743355830,
0.006286153075466025,
0.006753479541784039,
0.007221668262275161,
0.007689727507889525,
0.008156657320192495,
0.008621452166627776,
0.009083103623574271,
0.009540603079249766,
0.009992944448431896,
0.010439126890909214,
0.010878157525543523,
0.011309054131818317,
0.011730847830767882,
0.012142585737226980,
0.012543333575412086,
0.012932178249941664,
0.013308230364524597,
0.013670626680692492,
0.014018532509122373,
0.014351144026291085,
0.014667690509420725,
0.014967436482914990,
0.015249683769748768,
0.015513773441556675,
0.015759087661469738,
0.015985051414072162,
0.016191134117190663,
0.016376851110586923,
0.016541765016997088,
0.016685486971350718,
0.016807677714403370,
0.016908048547430839,
0.016986362145057821,
0.017042433223728004,
0.017076129063764694,
0.017087369883420026,

0.017076129063764694,
0.017042433223728004,
0.016986362145057821,
0.016908048547430839,
0.016807677714403370,
0.016685486971350718,
0.016541765016997088,
0.016376851110586923,
0.016191134117190663,
0.015985051414072162,
0.015759087661469738,
0.015513773441556675,
0.015249683769748768,
0.014967436482914990,
0.014667690509420725,
0.014351144026291085,
0.014018532509122373,
0.013670626680692492,
0.013308230364524597,
0.012932178249941664,
0.012543333575412086,
0.012142585737226980,
0.011730847830767882,
0.011309054131818317,
0.010878157525543523,
0.010439126890909214,
0.009992944448431896,
0.009540603079249766,
0.009083103623574271,
0.008621452166627776,
0.008156657320192495,
0.007689727507889525,
0.007221668262275161,
0.006753479541784039,
0.006286153075466025,
0.005820669743355830,
0.005357997000181999,
0.004899086349965267,
0.004444870878876082,
0.003996262853518041,
0.003554151391578438,
0.003119400211540369,
0.002692845467883248,
0.002275293677911463,

```
    0.001867519746044820,
]
```

Weights for sigC:

```
weights = [
    0.001867519746044820,
    0.002275293677911463,
    0.002692845467883248,
    0.003119400211540369,
    0.003554151391578438,
    0.003996262853518041,
    0.004444870878876082,
    0.004899086349965267,
    0.005357997000181999,
    0.005820669743355830,
    0.006286153075466025,
    0.006753479541784039,
    0.007221668262275161,
    0.007689727507889525,
    0.008156657320192495,
    0.008621452166627776,
    0.009083103623574271,
    0.009540603079249766,
    0.009992944448431896,
    0.010439126890909214,
    0.010878157525543523,
    0.011309054131818317,
    0.011730847830767882,
    0.012142585737226980,
    0.012543333575412086,
    0.012932178249941664,
    0.013308230364524597,
    0.013670626680692492,
    0.014018532509122373,
    0.014351144026291085,
    0.014667690509420725,
    0.014967436482914990,
    0.015249683769748768,
    0.015513773441556675,
    0.015759087661469738,
    0.015985051414072162,
    0.016191134117190663,
    0.016376851110586923,
```

0.016541765016997088,
0.016685486971350718,
0.016807677714403370,
0.016908048547430839,
0.016986362145057821,
0.017042433223728004,
0.017076129063764694,
0.017087369883420026,
0.017076129063764694,
0.017042433223728004,
0.016986362145057821,
0.016908048547430839,
0.016807677714403370,
0.016685486971350718,
0.016541765016997088,
0.016376851110586923,
0.016191134117190663,
0.015985051414072162,
0.015759087661469738,
0.015513773441556675,
0.015249683769748768,
0.014967436482914990,
0.014667690509420725,
0.014351144026291085,
0.014018532509122373,
0.013670626680692492,
0.013308230364524597,
0.012932178249941664,
0.012543333575412086,
0.012142585737226980,
0.011730847830767882,
0.011309054131818317,
0.010878157525543523,
0.010439126890909214,
0.009992944448431896,
0.009540603079249766,
0.009083103623574271,
0.008621452166627776,
0.008156657320192495,
0.007689727507889525,
0.007221668262275161,
0.006753479541784039,
0.006286153075466025,
0.005820669743355830,

```
    0.005357997000181999,
    0.004899086349965267,
    0.004444870878876082,
    0.003996262853518041,
    0.003554151391578438,
    0.003119400211540369,
    0.002692845467883248,
    0.002275293677911463,
    0.001867519746044820,
]
```

Weights for sigD:

```
weights = [
    0.001867519746044820,
    0.002275293677911463,
    0.002692845467883248,
    0.003119400211540369,
    0.003554151391578438,
    0.003996262853518041,
    0.004444870878876082,
    0.004899086349965267,
    0.005357997000181999,
    0.005820669743355830,
    0.006286153075466025,
    0.006753479541784039,
    0.007221668262275161,
    0.007689727507889525,
    0.008156657320192495,
    0.008621452166627776,
    0.009083103623574271,
    0.009540603079249766,
    0.009992944448431896,
    0.010439126890909214,
    0.010878157525543523,
    0.011309054131818317,
    0.011730847830767882,
    0.012142585737226980,
    0.012543333575412086,
    0.012932178249941664,
    0.013308230364524597,
    0.013670626680692492,
    0.014018532509122373,
    0.014351144026291085,
```

0.014667690509420725,
0.014967436482914990,
0.015249683769748768,
0.015513773441556675,
0.015759087661469738,
0.015985051414072162,
0.016191134117190663,
0.016376851110586923,
0.016541765016997088,
0.016685486971350718,
0.016807677714403370,
0.016908048547430839,
0.016986362145057821,
0.017042433223728004,
0.017076129063764694,
0.017087369883420026,
0.017076129063764694,
0.017042433223728004,
0.016986362145057821,
0.016908048547430839,
0.016807677714403370,
0.016685486971350718,
0.016541765016997088,
0.016376851110586923,
0.016191134117190663,
0.015985051414072162,
0.015759087661469738,
0.015513773441556675,
0.015249683769748768,
0.014967436482914990,
0.014667690509420725,
0.014351144026291085,
0.014018532509122373,
0.013670626680692492,
0.013308230364524597,
0.012932178249941664,
0.012543333575412086,
0.012142585737226980,
0.011730847830767882,
0.011309054131818317,
0.010878157525543523,
0.010439126890909214,
0.009992944448431896,
0.009540603079249766,

```
    0.009083103623574271,
    0.008621452166627776,
    0.008156657320192495,
    0.007689727507889525,
    0.007221668262275161,
    0.006753479541784039,
    0.006286153075466025,
    0.005820669743355830,
    0.005357997000181999,
    0.004899086349965267,
    0.004444870878876082,
    0.003996262853518041,
    0.003554151391578438,
    0.003119400211540369,
    0.002692845467883248,
    0.002275293677911463,
    0.001867519746044820,
]
```