Assignment No 3

EE18B120; Sarvani Cheruvu

February 11, 2020

1 Abstract

This experiment reads data from files and parses them. After analysing data, a function is fit to it. The effect of noise on the fitting process is studied and relevant graphs are plotted.

2 Tasks

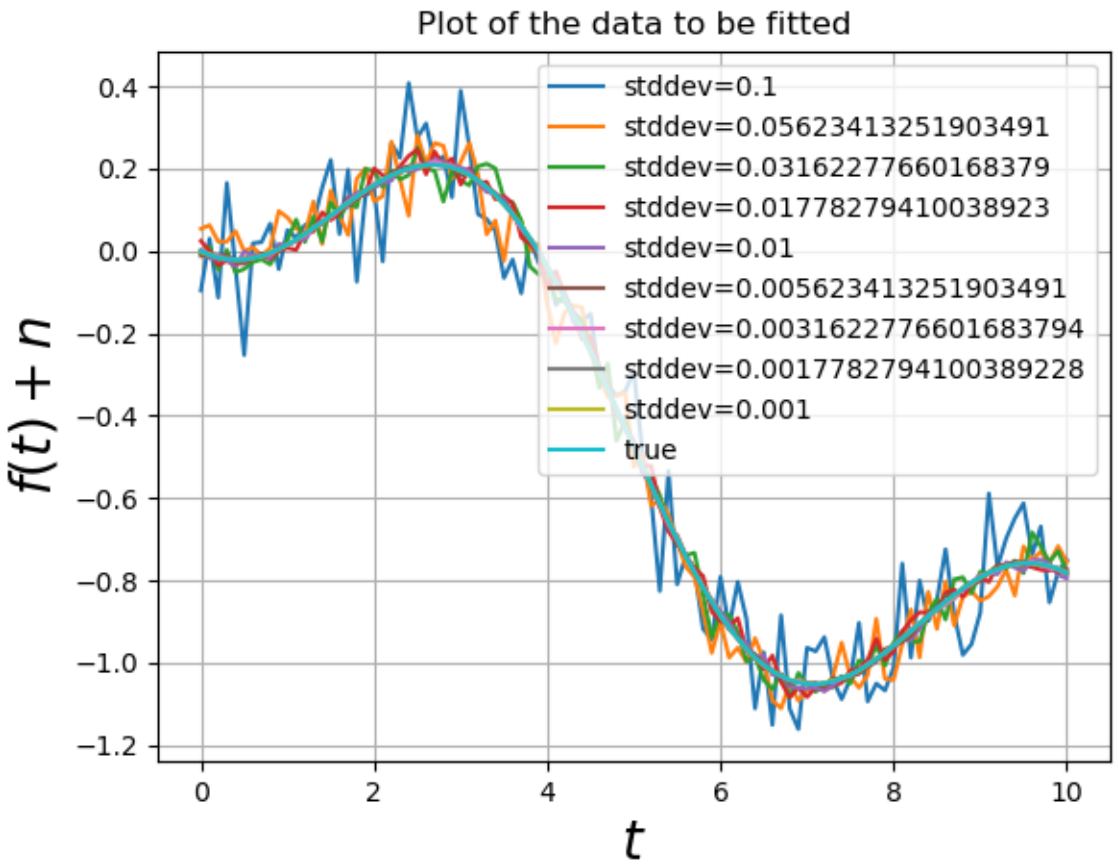
matplotlib throws an elementwise comparison error, which is yielding different plots when the same code is run on different versions of python. Please find attached, these two variants of plots.

1. Download and run the given python script.
2. The file has noisy data columns. Extract these using the following code:

```
yf=np.loadtxt("fitting.dat")
(N,k)=yf.shape
t=yf[:,0]
```

3. Plot the labelled noisy curves in Figure 0.

```
scl=logspace(-1,-3,9)
figure(0)
for i in range(1,k):
  plot(t,yf[:,i],label='$\sigma={}'.format(around(scl[i-1]),5))
xlabel(r'$t$',size=20)
ylabel(r'$f(t)+n$',size=20)
title(r'Plot of the data to be fitted')
```



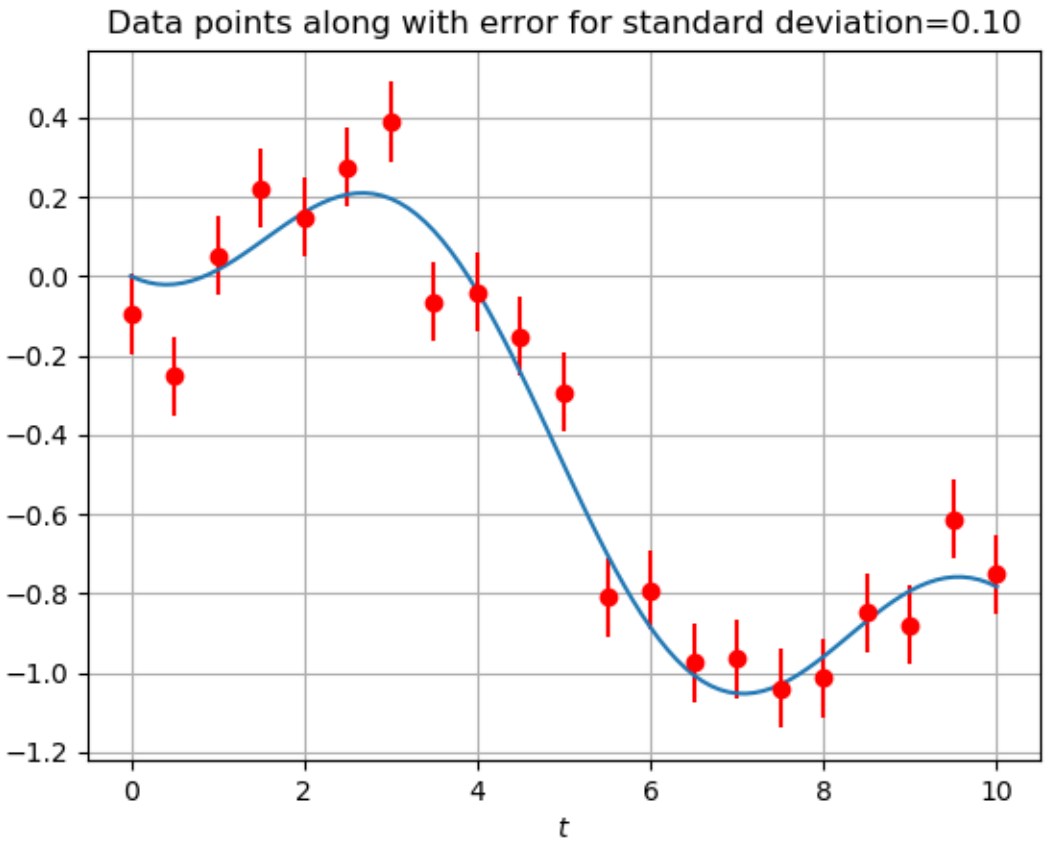
4. Define a function that fits a given function to the data. Thereafter, plot the true value (A=1.05, B=-0.105).

```
def g(tk=t,A=1.05,B=-0.105):
    return A*sp.jn(2,tk)+B*t
y=g()
plot(t,y,label='true')
legend()
grid(True)
show()
```

5. Generate a plot of the first column with errorbars, using every fifth point to do so.

```
figure(1)
plot(t,y,label='true')
```

```
errorbar(t[::5],yf[:,1][::5],scl[0],fmt='ro')
xlabel(t)
title(r'Data points along with error for
    standard deviation=0.10')
grid(True)
show()
```



6. Obtain *g(t;A;B)* as a column vector by creating a matrix equation.

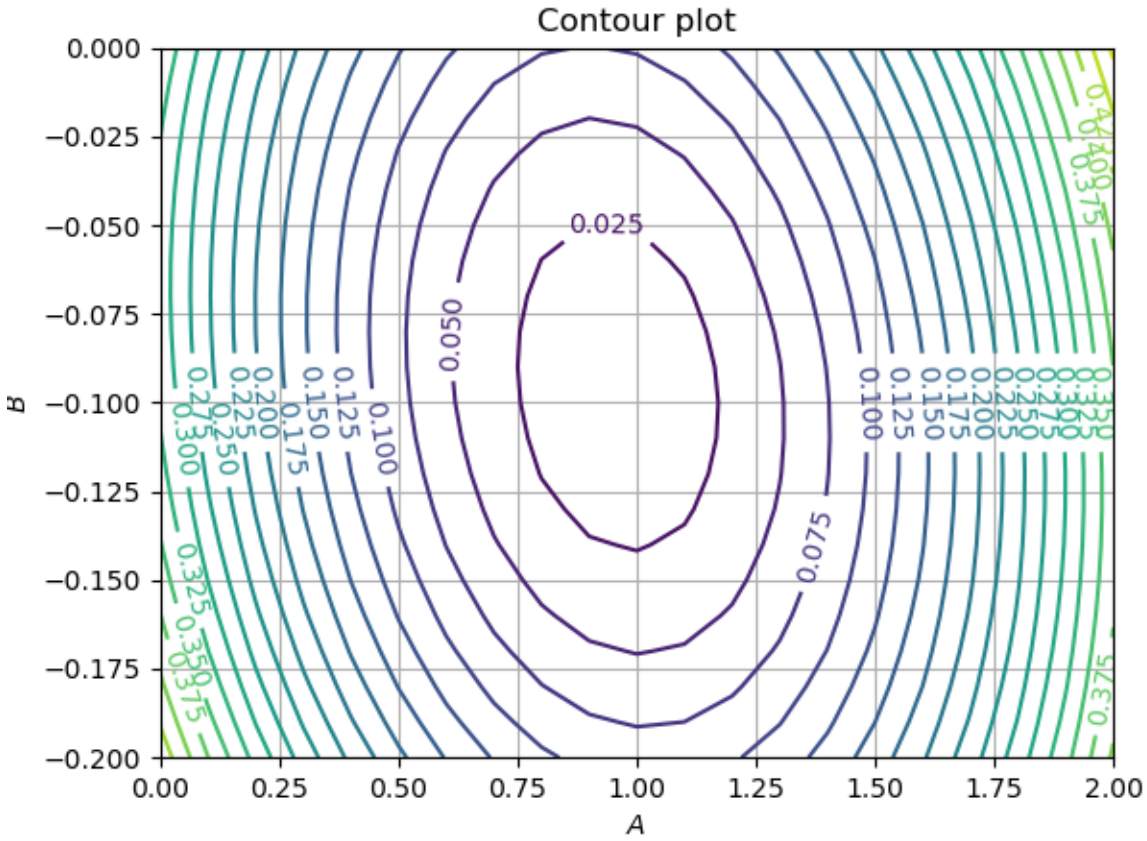
7. Compute the mean squared error between the data and the assumed model for A=0,0.1,...2 and B=-0.2,-0.19...0.

```
n=21
A=linspace(0,2,n)
B=linspace(-0.2,0,n)
eps=np.zeros((n,n))
```

```
for i in range(n):
  for j in range(n):
    eps[i][j]=mean(square(yf[:,1]-g(t,A[i],B[j])))
```

8. Plot a contour plot of ϵ_{ij}.

```
figure(2)
  pl=contour(A,B,eps,levels=20)
  xlabel(A)
  ylabel(B)
  title(r'Contour plot')
  clabel(pl)
  grid(True)
  show()
```



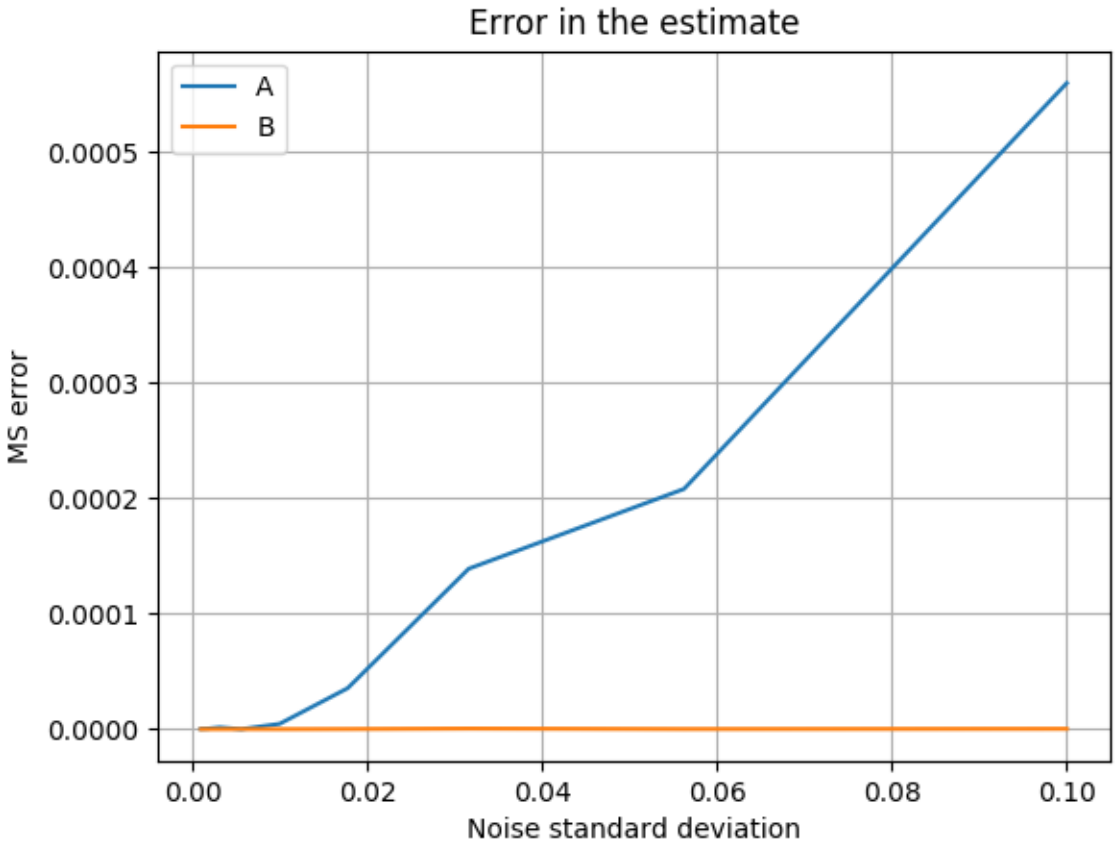
9. Obtain the best estimate of A and B using lstsq.

```
ex=np.zeros((2,1))
  ex=scipy.linalg.lstsq(M,y)[0]
```

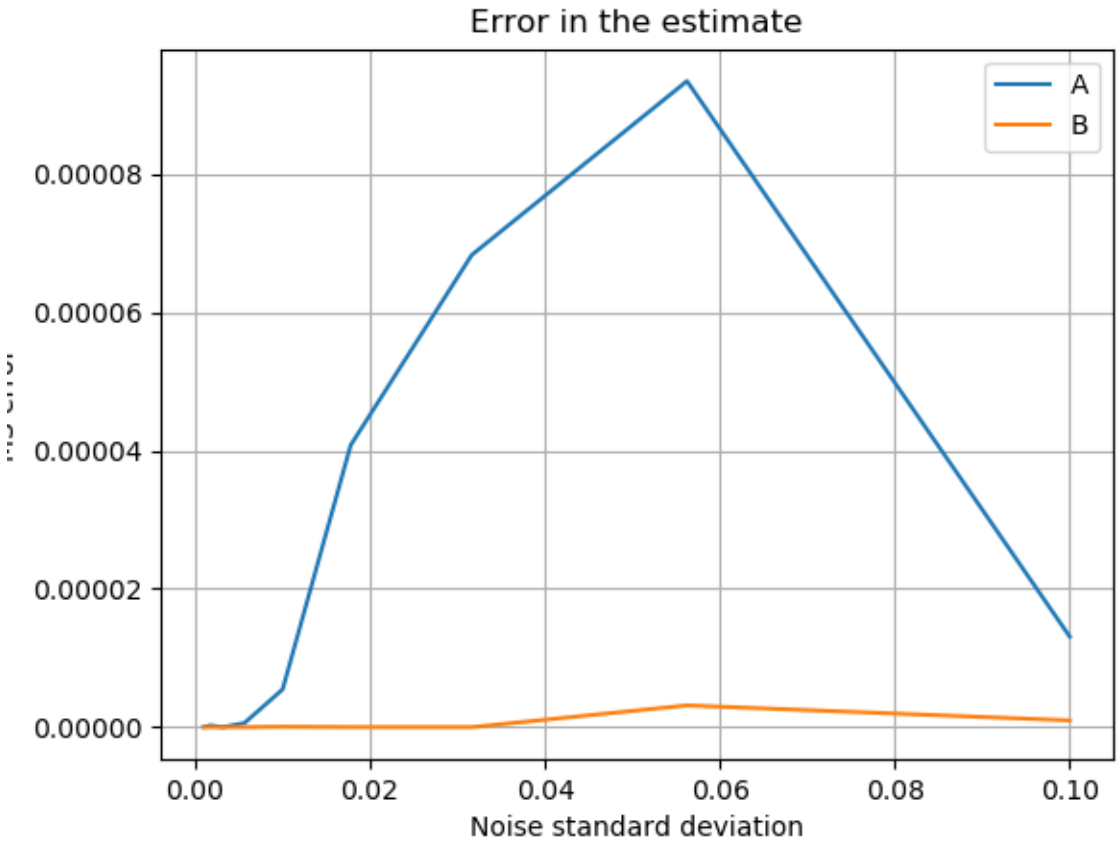
10. Plot the error in the estimate of A and B for different data files, versus the noise.

```
fit=np.zeros((k-1,2))
  for i in range(k-1):
    fit[i]=scipy.linalg.lstsq(M,yf[:,i+1])[0])
  Ae=np.zeros((k-1,1))
  Be=np.zeros((k-1,1))
  for i in range(k-1):
    Ae[i]=abs(fit[i][0]-ex[0])
    Be[i]=abs(fit[i][1]-ex[1])
  figure(3)
  plot(scl,Ae,label='A')
  plot(scl,Be,label='B')
  xlabel('Noise standard deviation')
  ylabel('MS error')
  title('Error in the estimate')
  legend()
  grid(True)
  show()
```

Graph, as plotted on ubuntu:



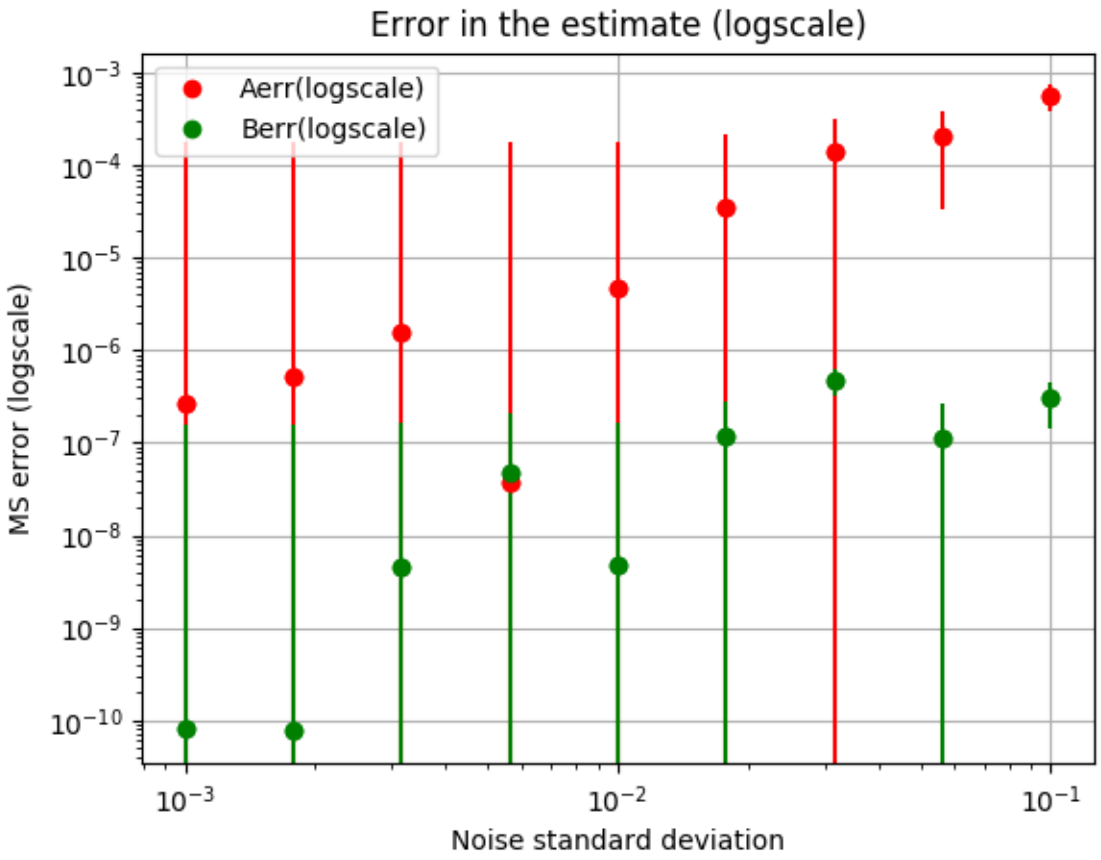
Graph, as plotted on windows:



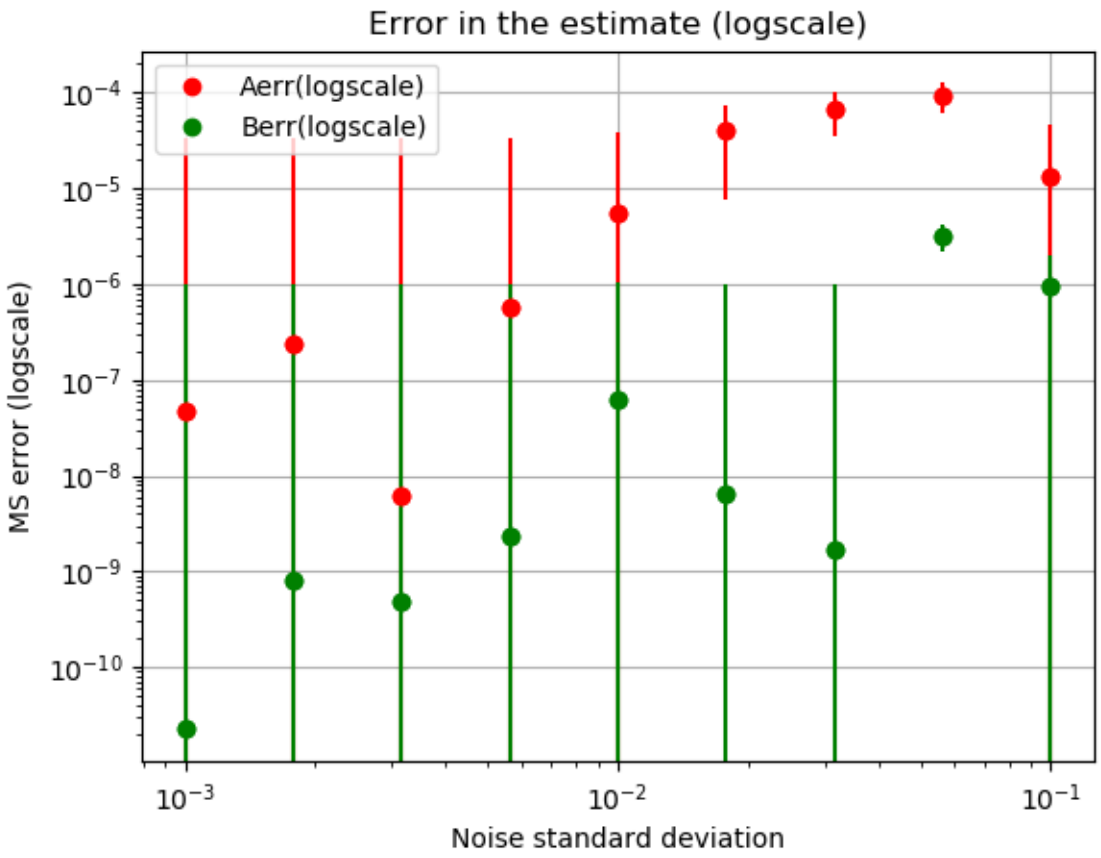
11. Replot above curves using loglog.

```
loglog(scl,Ae,'ro',label='Aerr(logscale)')
loglog(scl,Be,'go',label='Berr(logscale)')
errorbar(scl, Ae, std(Ae), fmt='ro')
errorbar(scl, Be, std(Be), fmt='go')
xlabel('Noise standard deviation')
ylabel('MS error (logscale)')
title('Error in the estimate (logscale)')
legend()
grid(True)
show()
```

Graph, as plotted on ubuntu:



Graph, as plotted on windows:



3 Conclusion

Noisy data was read from the file fitting.dat and a function was fit to it. The effect of noise on the fitting process was studied and graphs were plotted accordingly.