Calculation

August 19, 2023

0.0.1 Importing the necessary library

[102]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from uncertainties import unumpy
```

[103]:

```
df = pd.read_excel('data.xlsx')
```

[104]:

```
print(df.head())
```

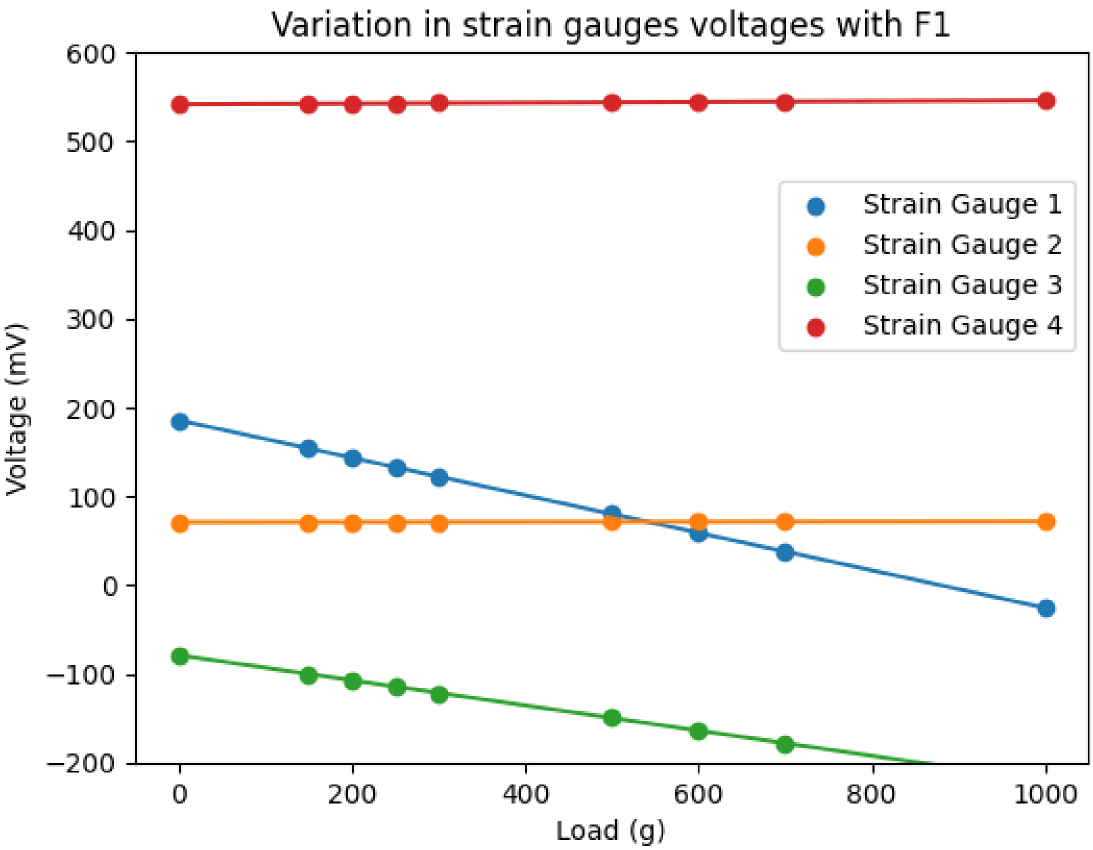
```
      F2  ch1_F2  ch2_F2  ch3_F2  ch4_F2     L2  ch1_L2  ch2_L2  ch3_L2  \
0    0.0  243.87   98.60  -31.40  587.30   20.0   243.8   111.5   -31.0
1   50.0  244.20  102.47  -30.58  593.22   40.0   243.8   116.9   -31.3
2  100.0  244.00  106.30  -30.80  599.30   60.0   243.5   122.1   -32.1
3  200.0  243.50  114.10  -31.50  611.30   80.0   242.6   127.3   -31.5
4  300.0  243.10  121.90  -31.30  622.70  100.0   242.2   132.7   -31.6

   ch4_L2   F1  ch1_F1  ch2_F1  ch3_F1  ch4_F1  L1  ch1_L1  ch2_L1  ch3_L1  \
0   619.2    0   185.7    70.9   -79.0   542.0   0   163.5    71.0  -107.0
1   624.5  150   154.0    71.0  -100.0   542.0  20   153.0    71.0  -112.0
2   629.1  200   143.5    71.0  -107.0   542.0  40   143.0    71.0  -117.5
3   635.1  250   133.0    71.0  -114.0   542.0  60   131.5    71.5  -123.0
4   640.6  300   121.5    71.0  -122.0   543.0  80   121.0    71.0  -128.0

   ch4_L1
0   541.5
1   541.8
2   542.0
3   542.0
4   542.5
```

0.0.2 Plotting the voltage changes in the strain gauges with change in F1

[115]:

```
load = np.array(df.F1).reshape(-1,1)
plt.scatter(df.F1, df.ch1_F1, label='Strain Gauge 1')
plt.scatter(df.F1, df.ch2_F1, label='Strain Gauge 2')
plt.scatter(df.F1, df.ch3_F1, label='Strain Gauge 3')
plt.scatter(df.F1, df.ch4_F1, label='Strain Gauge 4')
plt.legend(loc='center right', bbox_to_anchor=(1, 0.7))
plt.ylim([-200, 600])
plt.xlabel('Load (g)')
plt.ylabel('Voltage (mV)')
plt.title("Variation in strain gauges voltages with F1")
model = LinearRegression()

model.fit(load, df.ch1_F1)
ch1_F1_predict = model.predict(load)
ch1_F1_error = ch1_F1_predict - df.ch1_F1
k11 = model.coef_
plt.plot(load, ch1_F1_predict)

model.fit(load, df.ch2_F1)
ch2_F1_predict = model.predict(load)
ch2_F1_error = ch2_F1_predict - df.ch2_F1
k21 = model.coef_
plt.plot(load, ch2_F1_predict)

model.fit(load, df.ch3_F1)
ch3_F1_predict = model.predict(load)
ch3_F1_error = ch3_F1_predict - df.ch3_F1
k31 = model.coef_
plt.plot(load, ch3_F1_predict)

model.fit(load, df.ch4_F1)
k41 = model.coef_
ch4_F1_predict = model.predict(load)
ch4_F1_error = ch4_F1_predict - df.ch4_F1
plt.plot(load, ch4_F1_predict)
```

```
[115]: [<matplotlib.lines.Line2D at 0x283a6d89910>]
```

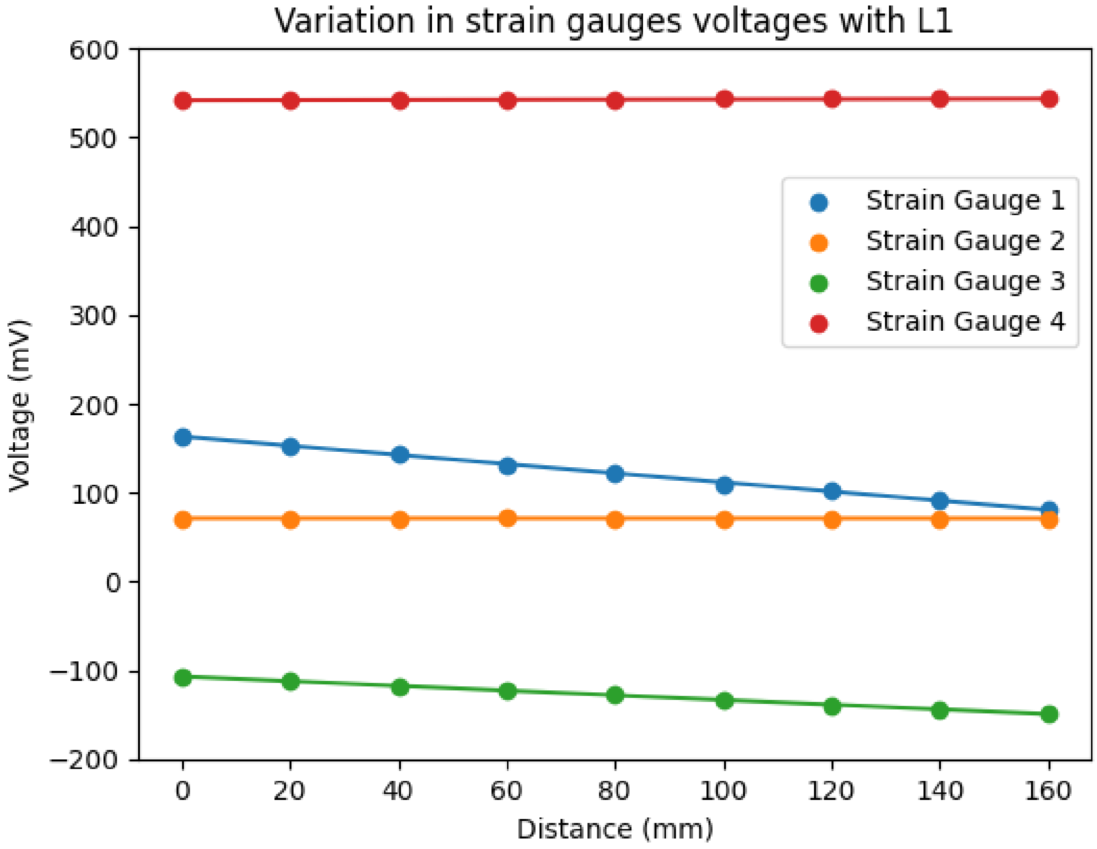


0.0.3 Plotting the voltage changes in strain gauges by changing the length L1

```
[114]: load = np.array(df.L1).reshape(-1,1)
plt.scatter(df.L1, df.ch1_L1, label='Strain Gauge 1')
plt.scatter(df.L1, df.ch2_L1, label='Strain Gauge 2')
plt.scatter(df.L1, df.ch3_L1, label='Strain Gauge 3')
plt.scatter(df.L1, df.ch4_L1, label='Strain Gauge 4')
plt.legend(loc='center right', bbox_to_anchor=(1, 0.7))
plt.ylim([-200, 600])
plt.xlabel('Distance (mm)')
plt.ylabel('Voltage (mV)')
plt.title("Variation in strain gauges voltages with L1")
model = LinearRegression()

model.fit(load, df.ch1_L1)
ch1_L1_predict = model.predict(load)
ch1_L1_error = ch1_L1_predict - df.ch1_L1
k12 = model.coef_
plt.plot(load, ch1_L1_predict)
```

```
model.fit(load, df.ch2_L1)
ch2_L1_predict = model.predict(load)
ch2_L1_error = ch2_L1_predict - df.ch2_L1
k22 = model.coef_
plt.plot(load, ch2_L1_predict)

model.fit(load, df.ch3_L1)
ch3_L1_predict = model.predict(load)
ch3_L1_error = ch3_L1_predict - df.ch3_L1
k32 = model.coef_
plt.plot(load, ch3_L1_predict)

model.fit(load, df.ch4_L1)
k42 = model.coef_
ch4_L1_predict = model.predict(load)
ch4_L1_error = ch4_L1_predict - df.ch4_L1
plt.plot(load, ch4_L1_predict)
```

```
[114]: [<matplotlib.lines.Line2D at 0x283a6ed8090>]
```



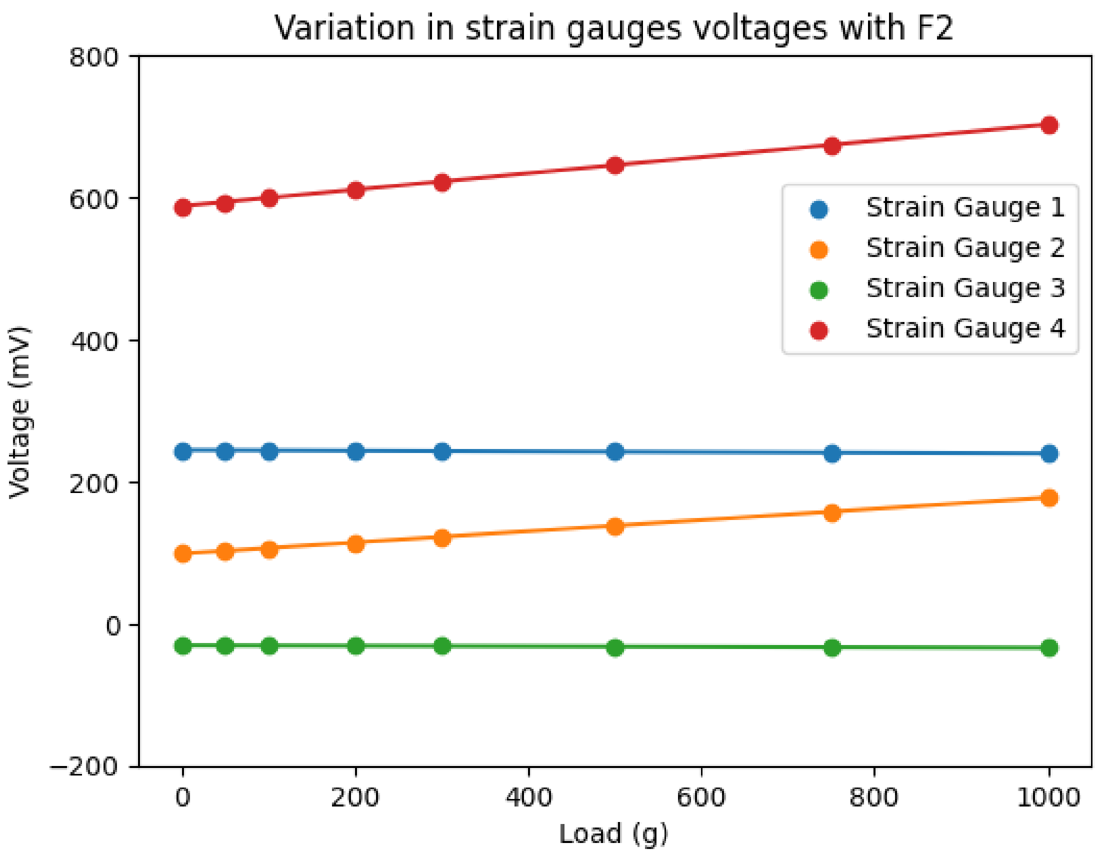
0.0.4 Plotting the voltage change in strain gauges by changing the load F2

[116]:

```
load = np.array(df.F2[0:8]).reshape(-1,1)
plt.scatter(df.F2[0:8], df.ch1_F2[0:8], label='Strain Gauge 1')
plt.scatter(df.F2[0:8], df.ch2_F2[0:8], label='Strain Gauge 2')
plt.scatter(df.F2[0:8], df.ch3_F2[0:8], label='Strain Gauge 3')
plt.scatter(df.F2[0:8], df.ch4_F2[0:8], label='Strain Gauge 4')
plt.legend(loc='center right', bbox_to_anchor=(1, 0.7))
plt.ylim([-200, 800])
plt.xlabel('Load (g)')
plt.ylabel('Voltage (mV)')
plt.title("Variation in strain gauges voltages with F2")
model = LinearRegression()

model.fit(load, df.ch1_F2[0:8])
ch1_F2_predict = model.predict(load)
ch1_F2_error = ch1_F2_predict - df.ch1_F2[0:8]
k13 = model.coef_
plt.plot(load, ch1_F2_predict)

model.fit(load, df.ch2_F2[0:8])
ch2_F2_predict = model.predict(load)
ch2_F2_error = ch2_F2_predict - df.ch2_F2[0:8]
k23 = model.coef_
plt.plot(load, ch2_F2_predict)

model.fit(load, df.ch3_F2[0:8])
ch3_F2_predict = model.predict(load)
ch3_F2_error = ch3_F2_predict - df.ch3_F2[0:8]
k33 = model.coef_
plt.plot(load, ch3_F2_predict)

model.fit(load, df.ch4_F2[0:8])
k43 = model.coef_
ch4_F2_predict = model.predict(load)
ch4_F2_error = ch4_F2_predict - df.ch4_F2[0:8]
plt.plot(load, ch4_F2_predict)
```

[116]:

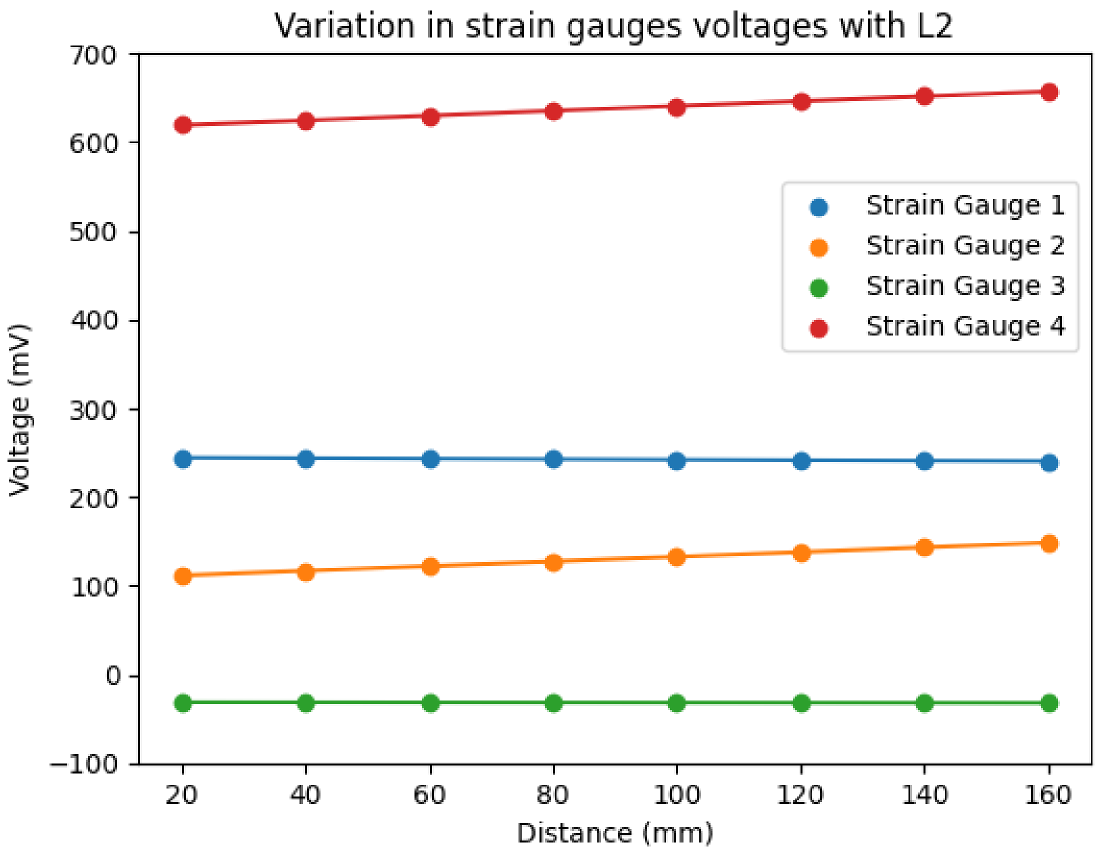
```
[<matplotlib.lines.Line2D at 0x283a6d78f10>]
```



```
[117]: load = np.array(df.L2[0:8]).reshape(-1,1)
       plt.scatter(df.L2[0:8], df.ch1_L2[0:8], label='Strain Gauge 1')
       plt.scatter(df.L2[0:8], df.ch2_L2[0:8], label='Strain Gauge 2')
       plt.scatter(df.L2[0:8], df.ch3_L2[0:8], label='Strain Gauge 3')
       plt.scatter(df.L2[0:8], df.ch4_L2[0:8], label='Strain Gauge 4')
       plt.legend(loc='center right', bbox_to_anchor=(1, 0.7))
       plt.ylim([-100, 700])
       plt.xlabel('Distance (mm)')
       plt.ylabel('Voltage (mV)')
       plt.title("Variation in strain gauges voltages with L2")
       model = LinearRegression()

       model.fit(load, df.ch1_L2[0:8])
       ch1_L2_predict = model.predict(load)
       ch1_L2_error = ch1_L2_predict - df.ch1_L2[0:8]
       k14 = model.coef_
       plt.plot(load, ch1_L2_predict)

       model.fit(load, df.ch2_L2[0:8])
```

```
ch2_L2_predict = model.predict(load)
ch2_L2_error = ch2_L2_predict - df.ch2_L2[0:8]
k24 = model.coef_
plt.plot(load, ch2_L2_predict)

model.fit(load, df.ch3_L2[0:8])
ch3_L2_predict = model.predict(load)
ch3_L2_error = ch3_L2_predict - df.ch3_L2[0:8]
k34 = model.coef_
plt.plot(load, ch3_L2_predict)

model.fit(load, df.ch4_L2[0:8])
k44 = model.coef_
ch4_L2_predict = model.predict(load)
ch4_L2_error = ch4_L2_predict - df.ch4_L2[0:8]
plt.plot(load, ch4_L2_predict)
```

```
[117]: [<matplotlib.lines.Line2D at 0x283a69a9f10>]
```



0.0.5 k_{ij} matrix magnitude

[109]:

```
kij_mag = np.array([[k11[0], k12[0], k13[0], k14[0]],[k21[0], k22[0], k23[0],␣
 ↪k24[0]], [k31[0], k32[0], k33[0], k34[0]], [k41[0], k42[0], k43[0], k44[0]]])
print(kij)
```

```
[[-2.11237229e-01 -5.15000000e-01 -4.69751037e-03 -2.73809524e-02]
 [ 7.79566130e-04 -4.16666667e-04  7.87600277e-02  2.63273810e-01]
 [-1.41645906e-01 -2.65000000e-01 -3.37067773e-03 -5.95238095e-03]
 [ 4.18474458e-03  1.05000000e-02  1.14865837e-01  2.71130952e-01]]
```

0.0.6 Error Calculation

[110]:

```
k11_err = np.std(ch1_F1_error[1:]/df.F1[1:])
k21_err = np.std(ch2_F1_error[1:]/df.F1[1:])
k31_err = np.std(ch3_F1_error[1:]/df.F1[1:])
k41_err = np.std(ch4_F1_error[1:]/df.F1[1:])

k12_err = np.std(ch1_L1_error[1:]/df.L1[1:])
k22_err = np.std(ch2_L1_error[1:]/df.L1[1:])
k32_err = np.std(ch3_L1_error[1:]/df.L1[1:])
k42_err = np.std(ch4_L1_error[1:]/df.L1[1:])

k13_err = np.std(ch1_F2_error[1:]/df.F2[1:])
k23_err = np.std(ch2_F2_error[1:]/df.F2[1:])
k33_err = np.std(ch3_F2_error[1:]/df.F2[1:])
k43_err = np.std(ch4_F2_error[1:]/df.F2[1:])

k14_err = np.std(ch1_L2_error[1:]/df.L2[1:])
k24_err = np.std(ch2_L2_error[1:]/df.L2[1:])
k34_err = np.std(ch3_L2_error[1:]/df.L2[1:])
k44_err = np.std(ch4_L2_error[1:]/df.L2[1:])

kij_error = [[k11_err, k12_err, k13_err, k14_err], [k21_err, k22_err, k23_err,␣
 ↪k24_err], [k31_err, k32_err, k33_err, k34_err], [k41_err, k42_err, k43_err,␣
 ↪k44_err]]
```

0.0.7 Final k_{ij} matrix

[113]:

```
kij = unumpy.uarray(kij_mag, kij_error)
print(kij)
```

```
[[-0.2112372288313506+/-0.0012088556127066172
  -0.5149999999999998+/-0.012022628391097068
  -0.004697510373443977+/-0.0005419882084376349
  -0.027380952380952506+/-0.002431994073693221]
 [0.000779566130160949+/-0.0002754745282922648
  -0.0004166666666666667+/-0.003024282807235162
  0.0787600276625173+/-0.000715074176637173
  0.2632738095238095+/-0.0009083558802635911]
 [-0.1416459062281316+/-0.0011732155476156724
  -0.26499999999999996+/-0.0051737862589533445
  -0.0033706777316735827+/-0.0019027888378710248
  -0.005952380952380956+/-0.004035826941759456]
 [0.004184744576627012+/-0.0011668175229557452
  0.010500000000000011+/-0.001988086054501155
  0.114865836791148+/-0.0026889564636480043
  0.2711309523809523+/-0.004490541984493112]]
```