

# Informe 2: Constante elástica del resorte

Carlos Eduardo Caballero Burgoa

200201226@est.umss.edu

21 de abril de 2021

**Grupo:** J2 (Miércoles)

**Docente:** Ing. Milka Mónica Torrico Troche

**Carrera:** Ing. Electromecánica

## Resumen

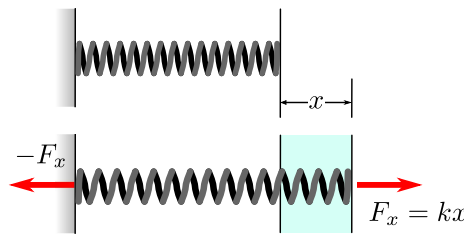
Este documento detalla el experimento realizado en simulador para hallar la constante de proporcionalidad de dos resortes a partir de la ley de *Hooke*, para esto se realizó la medición de la elongación de un resorte a diferentes cantidades de masas disponibles; y posteriormente se calculó la relación funcional con el método de mínimos cuadrados, finalmente se determinó el valor de la constante, resultando ser:  $(4.00 \pm 0.02)[N/m]$ ; 0.55 % para el primer resorte y  $(10.9 \pm 0.1)[N/m]$ ; 1.18 % para el segundo.

## 1. Introducción

Para mantener un resorte estirado una distancia  $x$  más allá de su longitud sin estirar, se debe aplicar una fuerza de igual magnitud en cada extremo como en la **Figura 1**. Si el alargamiento  $x$  no es excesivo, la fuerza aplicada al extremo derecho tiene una componente  $x$  directamente proporcional a  $x$ , esto se conoce como la ley de *Hooke*:

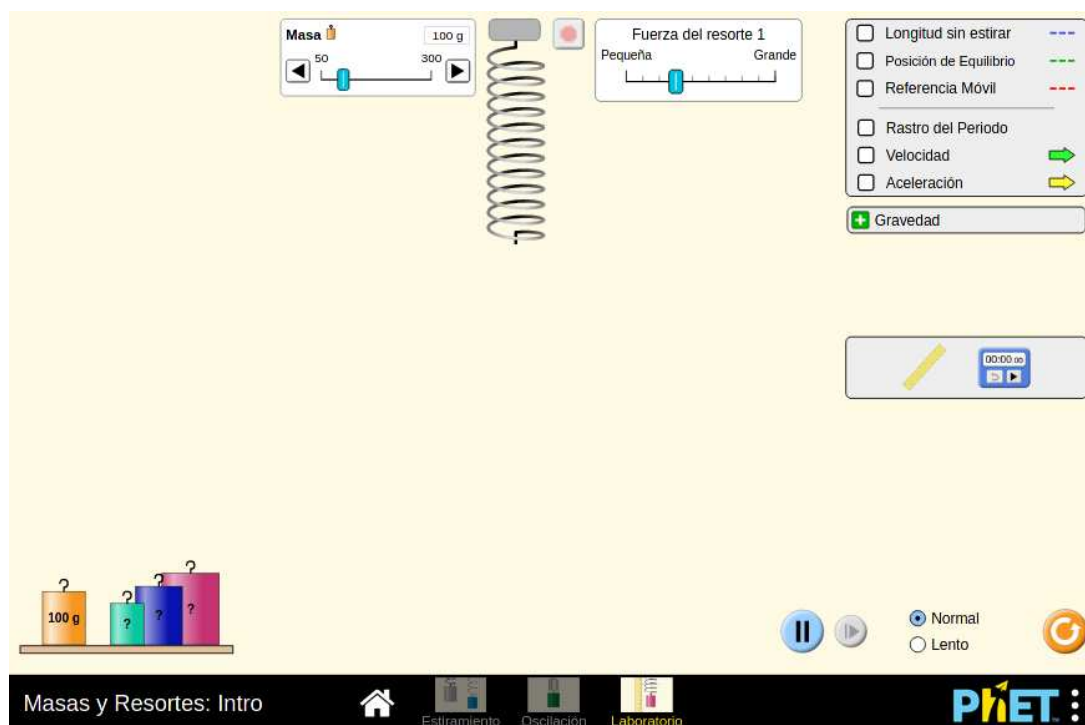
$$\vec{F}_x = -k\vec{x} \quad (1)$$

Donde  $k$  es una constante llamada **constante de fuerza** (o constante de resorte). Las unidades de  $k$  son de fuerza dividida entre distancia:  $[N/m]$  en el sistema internacional [1].



**Figura 1:** Fuerza necesaria para estirar un resorte.

**Nota:** Física Universitaria Volumen I (p. 188),  
Young, Hugh D. y Freedman, Roger A., 2013, Pearson.



**Figura 2:** Simulador de resortes.

**Nota:** Captura propia.

Para el experimento se verificará la **Ecuación 1**. A partir de la toma de datos de elongación y fuerza, se graficaron los datos, y finalmente se determinarán las constantes de los resortes usados.

## 2. Método experimental

Para la realización del experimento, se emplea el simulador de resortes de *PHET*, ubicado en la dirección web: [https://phet.colorado.edu/sims/html/masses-and-springs-basics/latest/masses-and-springs-basics\\_es.html](https://phet.colorado.edu/sims/html/masses-and-springs-basics/latest/masses-and-springs-basics_es.html), tal como se presenta en la **Figura 2**.

Para el simulador se escoge una fuerza del resorte, que se mantendrá constante durante la medición, y se registrarán diferentes valores de masa ( $m$ ) para medir su variación de longitud ( $x$ ).

Una vez medidos los datos para dos resortes con diferente fuerza del resorte, se procederá a graficar la relación fuerza vs. longitud del resorte, y con la ayuda del método de los mínimos cuadrados, se halla la relación funcional entre las variables.

### 2.1. Resorte pequeño

En el **Cuadro 1**, se pueden ver los valores tomados del experimento, tanto la masa como la longitud de la deformación resultante, para una fuerza del resorte pequeño.

$i$	$m_i[g]$	$x_i[cm]$	$i$	$m_i[g]$	$x_i[cm]$
1	0	47	10	130	79
2	50	60	11	140	82
3	60	62	12	150	84
4	70	65	13	160	86
5	80	67	14	170	89
6	90	69	15	180	91
7	100	72	16	190	94
8	110	74	17	200	96
9	120	77	18	210	99

**Cuadro 1:** Mediciones de longitud en función de la masa provista (Resorte pequeño).

**Nota:** Elaboración propia.

## 2.2. Resorte grande

En el **Cuadro 2**, pueden ver los valores tomados del experimento, tanto la masa como la longitud de la deformación resultante, para una fuerza del resorte grande.

$i$	$m_i[g]$	$x_i[cm]$	$i$	$m_i[g]$	$x_i[cm]$
1	0	47	10	220	67
2	140	60	11	230	68
3	150	61	12	240	69
4	160	61	13	250	70
5	170	62	14	260	70
6	180	63	15	270	71
7	190	64	16	280	72
8	200	65	17	290	73
9	210	66	18	300	74

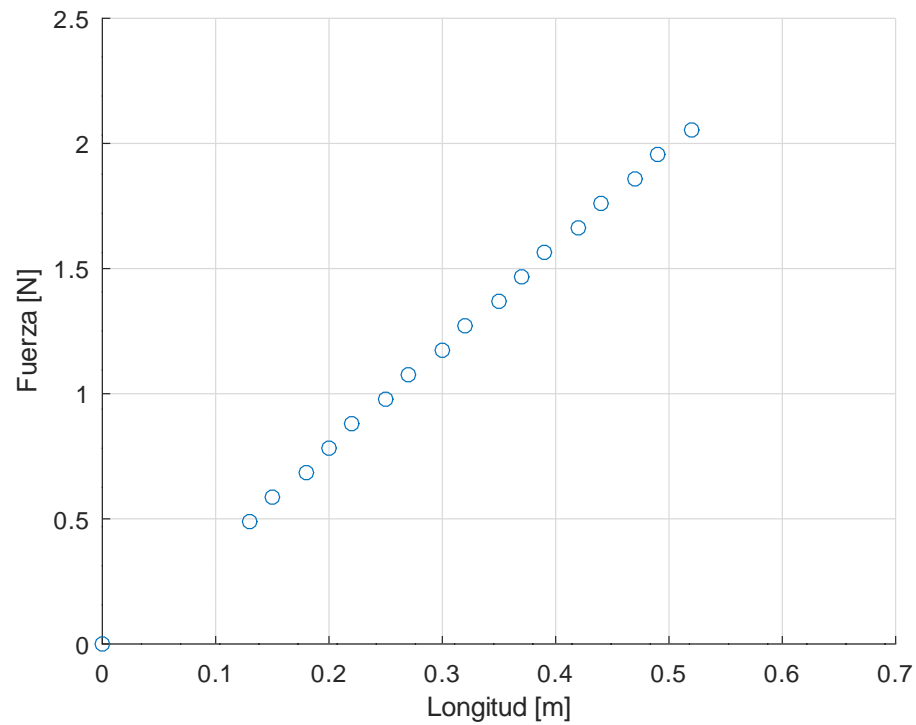
**Cuadro 2:** Mediciones de longitud en función de la masa provista (Resorte grande).

**Nota:** Elaboración propia.

## 3. Resultados

### 3.1. Resorte pequeño

A partir de los datos obtenidos se genera la gráfica de la **Figura 3**.



**Figura 3:** Gráfica de longitud vs fuerza (Resorte pequeño).

**Nota:** Elaboración propia.

Posteriormente se calculo la recta de mejor ajuste por el método de los mínimos cuadrados, resultando los siguientes valores:

$$A = (-0.016 \pm 0.007)[N]; 46.89 \%$$

$$B = (4.00 \pm 0.02)[N/m]; 0.55 \%$$

Siendo su coeficiente de correlación ( $r$ ):

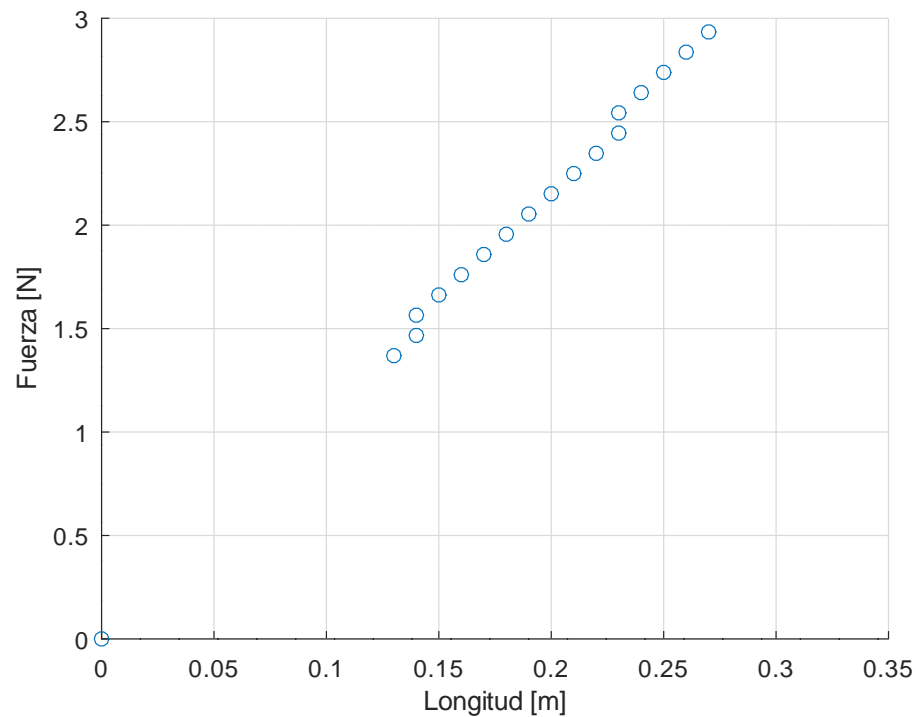
$$r = 0.9998$$

Considerando que el modelo de ajuste es:

$$F = A + Bx$$

Por tanto la relación funcional entre  $F$  y  $x$ , es:

Resultado
$F \propto x$



**Figura 4:** Gráfica de longitud vs fuerza (Resorte grande).

**Nota:** Elaboración propia.

Verificándose el comportamiento establecido por la **Ecuación 1**.

La constante elástica para el resorte pequeño del experimento es:

Resultado
$k = (4.00 \pm 0.02)[N/m]; 0.55 \%$

### 3.2. Resorte grande

A partir de los datos obtenidos se genera la gráfica de la **Figura 4**.

Posteriormente se calculo la recta de mejor ajuste por el método de los mínimos cuadrados, resultando los siguientes valores:

$$A = (-0.01 \pm 0.02)[N]; 330.46 \%$$

$$B = (10.9 \pm 0.1)[N/m]; 1.18 \%$$

Siendo su coeficiente de correlación ( $r$ ):

$$r = 0.9989$$

Considerando que el modelo de ajuste es:

$$F = A + Bx$$

Por tanto la relación funcional entre  $F$  y  $x$ , es:

Resultado
$F \propto x$

La constante elástica para el resorte grande del experimento es:

Resultado
$k = (10.9 \pm 0.1)[N/m]; 1.18 \%$

## 4. Discusión

Puede notarse que en ambos resortes la variable  $a$  es negativa, además que el error de  $b$  es pequeño, ambas son reflejo del uso de un simulador, ya que se están tratando con resortes ideales.

## 5. Conclusiones

Se halló la relación funcional entre el incremento de la longitud del resorte y la fuerza aplicada, verificándose así la ley de *Hooke*.

También se calculó el valor de la constante elástica del resorte.

## Referencias

- [1] Young, Hugh D. y Freedman, Roger A. (2013).  
Física Universitaria. Volumen 1.  
13va Edición.  
Capítulo 11.

## Apéndice A: Cálculos adicionales

### 5.1. Resorte pequeño

Conociendo  $m_i$ ,  $x_i$ , y  $g$ , se detallan los valores de la fuerza calculada ( $F$ ) y la longitud del estiramiento del resorte pequeño ( $\Delta x$ ) en el **Cuadro 3**.

$i$	$F_i[N]$	$\Delta x_i[m]$	$i$	$F_i[N]$	$\Delta x_i[m]$
1	0	0	10	1.2714	0.3200
2	0.4890	0.1300	11	1.3692	0.3500
3	0.5868	0.1500	12	1.4670	0.3700
4	0.6846	0.1800	13	1.5648	0.3900
5	0.7824	0.2000	14	1.6626	0.4200
6	0.8802	0.2200	15	1.7604	0.4400
7	0.9780	0.2500	16	1.8582	0.4700
8	1.0758	0.2700	17	1.9560	0.4900
9	1.1736	0.3000	18	2.0538	0.5200

**Cuadro 3:** Calculo de la fuerza y la longitud (Resorte pequeño).

**Nota:** Elaboración propia.

Se calculan los parámetros de la recta por el método de los mínimos cuadrados, con la ayuda de los datos presentados en el **Cuadro 4**.

$$\begin{aligned}
 n &= 18 \\
 \sum x_i &= 5.4700 \\
 \sum y_i &= 21.6138 \\
 \sum x_i^2 &= 2.0009 \\
 \sum y_i^2 &= 31.3822 \\
 \sum x_i y_i &= 7.9238 \\
 \Delta_1 &= n \sum x_i^2 - \left( \sum x_i \right)^2 = 6.0953 \\
 \Delta_2 &= n \sum y_i^2 - \left( \sum y_i \right)^2 = 97.7240 \\
 A &= \frac{\sum y_i \sum x_i^2 - \sum x_i y_i \sum x_i}{\Delta_1} = -0.0157 \\
 B &= \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\Delta_1} = 4.0031 \\
 \sum d^2 &= 0.0027
 \end{aligned}$$

$i$	$x_i^2$	$y_i^2$	$x_i y_i$	$Y_i$	$d_i$	$d_i^2 (10^{-3})$
1	0	0	0	-0.0157	0.0157	0.2475
2	0.0169	0.2391	0.0636	0.5047	-0.0157	0.245
3	0.0225	0.3443	0.0880	0.5847	0.0021	0.004
4	0.0324	0.4687	0.1232	0.7048	-0.0202	0.4091
5	0.0400	0.6121	0.1565	0.7849	-0.0025	0.0062
6	0.0484	0.7748	0.1936	0.8650	0.0152	0.2325
7	0.0625	0.9565	0.2445	0.9850	-0.0070	0.0496
8	0.0729	1.1573	0.2905	1.0651	0.0107	0.1144
9	0.0900	1.3773	0.3521	1.1852	-0.0116	0.1345
10	0.1024	1.6165	0.4068	1.2653	0.0061	0.0377
11	0.1225	1.8747	0.4792	1.3854	-0.0162	0.2610
12	0.1369	2.1521	0.5428	1.4654	0.0016	0.0025
13	0.1521	2.4486	0.6103	1.5455	0.0193	0.3733
14	0.1764	2.7642	0.6983	1.6656	-0.0030	0.0088
15	0.1936	3.0990	0.7746	1.7456	0.0148	0.2180
16	0.2209	3.4529	0.8734	1.8657	-0.0075	0.0567
17	0.2401	3.8259	0.9584	1.9458	0.0102	0.104
18	0.2704	4.2181	1.0680	2.0659	-0.0121	0.146

**Cuadro 4:** Valores para el método de mínimos cuadrados (Resorte pequeño).**Nota:** Elaboración propia.

$$\sigma^2 = \frac{\sum d_i^2}{n-2} = 1.6575 \times 10^{-4}$$

$$\sigma_A = \sqrt{\frac{\sigma^2 \sum x_i^2}{\Delta_1}} = 0.0074$$

$$\sigma_B = \sqrt{\frac{\sigma^2 n}{\Delta_1}} = 0.0221$$

Parámetros de la recta obtenida:

$$A = (-0.0157 \pm 0.0074)[N]; 46.8866 \%$$

$$B = (4.0031 \pm 0.0221)[N/m]; 0.5527 \%$$

Siendo el coeficiente de correlación:

$$R = \frac{n \sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{\Delta_1 \Delta_2}} = 0.9998$$



La ecuación de la recta resultante es:

$$y = -0.0157 + 4.0031 x$$

## 5.2. Resorte grande

Conociendo  $m_i$ ,  $x_i$ , y  $g$ , se detallan los valores de la fuerza calculada ( $F$ ) y la longitud del estiramiento del resorte grande ( $\Delta x$ ) en el **Cuadro 5**.

$i$	$F_i[N]$	$\Delta x_i[m]$	$i$	$F_i[N]$	$\Delta x_i[m]$
1	0	0	10	2.1516	0.2000
2	1.3692	0.1300	11	2.2494	0.2100
3	1.4670	0.1400	12	2.3472	0.2200
4	1.5648	0.1400	13	2.4450	0.2300
5	1.6626	0.1500	14	2.5428	0.2300
6	1.7604	0.1600	15	2.6406	0.2400
7	1.8582	0.1700	16	2.7384	0.2500
8	1.9560	0.1800	17	2.8362	0.2600
9	2.0538	0.1900	18	2.9340	0.2700

**Cuadro 5:** Calculo de la fuerza y la longitud (Resorte grande).

**Nota:** Elaboración propia.

Se calculan los parámetros de la recta por el método de los mínimos cuadrados, con la ayuda de los datos presentados en el **Cuadro 6**.

$$n = 18$$

$$\sum x_i = 3.3700$$

$$\sum y_i = 36.5772$$

$$\sum x_i^2 = 0.7005$$

$$\sum y_i^2 = 82.6020$$

$$\sum x_i y_i = 7.6059$$

$$\Delta_1 = n \sum x_i^2 - \left( \sum x_i \right)^2 = 1.2521$$

$$\Delta_2 = n \sum y_i^2 - \left( \sum y_i \right)^2 = 148.9437$$

$$A = \frac{\sum y_i \sum x_i^2 - \sum x_i y_i \sum x_i}{\Delta_1} = -0.0076$$

$i$	$x_i^2$	$y_i^2$	$x_i y_i$	$Y_i$	$d_i$	$d_i^2$
1	0	0	0	-0.0076	0.0076	0.0001
2	0.0169	1.8747	0.1780	1.4087	-0.0395	0.0016
3	0.0196	2.1521	0.2054	1.5176	-0.0506	0.0026
4	0.0196	2.4486	0.2191	1.5176	0.0472	0.0022
5	0.0225	2.7642	0.2494	1.6265	0.0361	0.0013
6	0.0256	3.0990	0.2817	1.7355	0.0249	0.0006
7	0.0289	3.4529	0.3159	1.8444	0.0138	0.0002
8	0.0324	3.8259	0.3521	1.9534	0.0026	0.0000
9	0.0361	4.2181	0.3902	2.0623	-0.0085	0.0001
10	0.0400	4.6294	0.4303	2.1713	-0.0197	0.0004
11	0.0441	5.0598	0.4724	2.2802	-0.0308	0.0009
12	0.0484	5.5093	0.5164	2.3892	-0.0420	0.0018
13	0.0529	5.9780	0.5624	2.4981	-0.0531	0.0028
14	0.0529	6.4658	0.5848	2.4981	0.0447	0.0020
15	0.0576	6.9728	0.6337	2.6071	0.0335	0.0011
16	0.0625	7.4988	0.6846	2.7160	0.0224	0.0005
17	0.0676	8.0440	0.7374	2.8250	0.0112	0.0001
18	0.0729	8.6084	0.7922	2.9339	0.0001	0.0000

**Cuadro 6:** Valores para el método de mínimos cuadrados (Resorte grande).**Nota:** Elaboración propia.

$$B = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\Delta_1} = 10.8946$$

$$\sum d^2 = 0.0183$$

$$\sigma^2 = \frac{\sum d_i^2}{n - 2} = 0.0011$$

$$\sigma_A = \sqrt{\frac{\sigma^2 \sum x_i^2}{\Delta_1}} = 0.0253$$

$$\sigma_B = \sqrt{\frac{\sigma^2 n}{\Delta_1}} = 0.1281$$

Parámetros de la recta obtenida:

$$A = (-0.0076 \pm 0.0253)[N]; 330.4569 \%$$

$$B = (10.8946 \pm 0.1281)[N/m]; 1.1758 \%$$

Siendo el coeficiente de correlación:

$$R = \frac{n \sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{\Delta_1 \Delta_2}} = 0.9989$$

La ecuación de la recta resultante es:

$$y = -0.0076 + 10.8946 x$$

## Apéndice B: Cálculos realizados en *Octave*

A continuación se presenta los cálculos realizados en el programa *Octave* para la generación de las gráficas, el calculo de los mínimos cuadrados y el valor de la constante elástica.

### 5.3. Resorte pequeño

```
# Datos importados (i1.csv):
0,47
50,60
60,62
70,65
80,67
90,69
100,72
110,74
120,77
130,79
140,82
150,84
160,86
170,89
180,91
190,94
200,96
210,99

# Comandos ejecutados (o1.m):
function graficar(t,_x,_y,eps,x,y)
    f = figure()

    title(t)
    xlabel(_x)
    ylabel(_y)

    hold on
    grid on
    plot(x, y, 'o')

    print(f,eps,'-color')
    hold off
end

function [A,sA,B,sB,R]=minimoscuadrados(x,y)
    xx = x.*x
    yy = y.*y
    xy = x.*y

    % tamaño de la muestra
    n = length(x)

    % calculo de las sumatorias
```

```

sx = sum(x)
sy = sum(y)

sxx = sum(xx)
syy = sum(yy)
sxy = sum(xy)

D1 = (n * sxx) - (sx)^2
D2 = (n * syy) - (sy)^2

% calculo de los valores de la recta
A = ( (sy * sxx) - (sxy * sx) ) / D1
B = ( (n * sxy) - (sx * sy) ) / D1

% calculo del error
Y = A + (B * x)
d = y - Y

dd = d.*d
sdd = sum(dd)
s2 = sdd / ( n - 2)

sA = sqrt( (s2 * sxx) / D1 )
sB = sqrt( (s2 * n) / D1 )

% calculando el error porcentual
EA = abs(sA / A) * 100
EB = abs(sB / B) * 100

% calculo de correlacion
R = ((n * sxy) - (sx * sy)) / sqrt( D1 * D2 )
end

clear
close all
clc

addpath(' ../../../../octave')

% leer datos previamente formateados
table = csvread('i1.csv')

g = 9.78

m = table(:,1)
l = table(:,2)
y0 = table(1,2)

% conversion a SI
m_kg = m/1000
d_cm = l-y0
d_mt = d_cm/100

% conversion F=mg y d=d-do
F = arrayfun(@(m) g*m, m_kg)
```

```
d = arrayfun(@(d) d, d_mt)

graficar(
    '',
    'Longitud [m]',
    'Fuerza [N]',
    'o1.eps',
    d,
    F
)

x = d
y = F

[A,sA,B,sB,R]=minimoscuadrados(x,y)

# Salida del programa (o1.out):
table =

    0    47
   50    60
   60    62
   70    65
   80    67
   90    69
  100    72
  110    74
  120    77
  130    79
  140    82
  150    84
  160    86
  170    89
  180    91
  190    94
  200    96
  210    99

g = 9.7800
m =

    0
   50
   60
   70
   80
   90
  100
  110
  120
  130
  140
  150
  160
  170
```

```
180
190
200
210

l =

47
60
62
65
67
69
72
74
77
79
82
84
86
89
91
94
96
99

y0 = 47
m_kg =

0
0.0500
0.0600
0.0700
0.0800
0.0900
0.1000
0.1100
0.1200
0.1300
0.1400
0.1500
0.1600
0.1700
0.1800
0.1900
0.2000
0.2100

d_cm =

0
13
15
18
20
22
```

25  
27  
30  
32  
35  
37  
39  
42  
44  
47  
49  
52

d\_mt =

0  
0.1300  
0.1500  
0.1800  
0.2000  
0.2200  
0.2500  
0.2700  
0.3000  
0.3200  
0.3500  
0.3700  
0.3900  
0.4200  
0.4400  
0.4700  
0.4900  
0.5200

F =

0  
0.4890  
0.5868  
0.6846  
0.7824  
0.8802  
0.9780  
1.0758  
1.1736  
1.2714  
1.3692  
1.4670  
1.5648  
1.6626  
1.7604  
1.8582  
1.9560  
2.0538

d =



0  
0.1300  
0.1500  
0.1800  
0.2000  
0.2200  
0.2500  
0.2700  
0.3000  
0.3200  
0.3500  
0.3700  
0.3900  
0.4200  
0.4400  
0.4700  
0.4900  
0.5200

f = 1

x =

0  
0.1300  
0.1500  
0.1800  
0.2000  
0.2200  
0.2500  
0.2700  
0.3000  
0.3200  
0.3500  
0.3700  
0.3900  
0.4200  
0.4400  
0.4700  
0.4900  
0.5200

y =

0  
0.4890  
0.5868  
0.6846  
0.7824  
0.8802  
0.9780  
1.0758  
1.1736  
1.2714  
1.3692  
1.4670

1.5648  
1.6626  
1.7604  
1.8582  
1.9560  
2.0538

xx =

0  
0.0169  
0.0225  
0.0324  
0.0400  
0.0484  
0.0625  
0.0729  
0.0900  
0.1024  
0.1225  
0.1369  
0.1521  
0.1764  
0.1936  
0.2209  
0.2401  
0.2704

yy =

0  
0.2391  
0.3443  
0.4687  
0.6121  
0.7748  
0.9565  
1.1573  
1.3773  
1.6165  
1.8747  
2.1521  
2.4486  
2.7642  
3.0990  
3.4529  
3.8259  
4.2181

xy =

0  
0.0636  
0.0880  
0.1232  
0.1565

```
0.1936
0.2445
0.2905
0.3521
0.4068
0.4792
0.5428
0.6103
0.6983
0.7746
0.8734
0.9584
1.0680

n = 18
sx = 5.4700
sy = 21.614
sxx = 2.0009
syy = 31.382
sxy = 7.9238
D1 = 6.0953
D2 = 97.724
A = -0.015732
B = 4.0031
Y =

-0.015732
0.504671
0.584733
0.704827
0.784889
0.864951
0.985044
1.065106
1.185199
1.265261
1.385354
1.465416
1.545478
1.665572
1.745634
1.865727
1.945789
2.065882

d =

1.5732e-02
-1.5671e-02
2.0666e-03
-2.0227e-02
-2.4886e-03
1.5249e-02
-7.0438e-03
1.0694e-02
-1.1599e-02
```

```
6.1389e-03
-1.6154e-02
1.5837e-03
1.9322e-02
-2.9716e-03
1.4766e-02
-7.5268e-03
1.0211e-02
-1.2082e-02

dd =

2.4750e-04
2.4559e-04
4.2709e-06
4.0911e-04
6.1931e-06
2.3254e-04
4.9615e-05
1.1436e-04
1.3454e-04
3.7686e-05
2.6096e-04
2.5080e-06
3.7332e-04
8.8302e-06
2.1805e-04
5.6652e-05
1.0427e-04
1.4597e-04

sdd = 2.6520e-03
s2 = 1.6575e-04
sA = 7.3763e-03
sB = 0.022124
EA = 46.887
EB = 0.5527
R = 0.9998
A = -0.015732
sA = 7.3763e-03
B = 4.0031
sB = 0.022124
R = 0.9998
```

#### 5.4. Resorte grande

```
# Datos importados (i2.csv):
0,47
140,60
150,61
```

```
160,61
170,62
180,63
190,64
200,65
210,66
220,67
230,68
240,69
250,70
260,70
270,71
280,72
290,73
300,74

# Comandos ejecutados (o2.m):
function graficar(t,_x,_y,eps,x,y)
    f = figure()

    title(t)
    xlabel(_x)
    ylabel(_y)

    hold on
    grid on
    plot(x, y, 'o')

    print(f,eps,'-color')
    hold off
end

function [A,sA,B,sB,R]=minimoscuadrados(x,y)
    xx = x.*x
    yy = y.*y
    xy = x.*y

    % tamaño de la muestra
    n = length(x)

    % calculo de las sumatorias
    sx = sum(x)
    sy = sum(y)

    sxx = sum(xx)
    syy = sum(yy)
    sxy = sum(xy)

    D1 = (n * sxx) - (sx)^2
    D2 = (n * syy) - (sy)^2

    % calculo de los valores de la recta
    A = ( (sy * sxx) - (sxy * sx) ) / D1
    B = ( (n * sxy) - (sx * sy) ) / D1
```

```
% calculo del error
Y = A + (B * x)
d = y - Y

dd = d.*d
sdd = sum(dd)
s2 = sdd / ( n - 2)

sA = sqrt( (s2 * sxx) / D1 )
sB = sqrt( (s2 * n) / D1 )

% calculando el error porcentual
EA = abs(sA / A) * 100
EB = abs(sB / B) * 100

% calculo de correlacion
R = ((n * sxy) - (sx * sy)) / sqrt( D1 * D2 )
end

clear
close all
clc

addpath(' ../../octave')

% leer datos previamente formateados
table = csvread('i2.csv')

g = 9.78

m = table(:,1)
l = table(:,2)
y0 = table(1,2)

% conversion a SI
m_kg = m/1000
d_cm = l-y0
d_mt = d_cm/100

% conversion F=mg y d=d-do
F = arrayfun(@(m) g*m, m_kg)
d = arrayfun(@(d) d, d_mt)

graficar(
    '',
    'Longitud [m]',
    'Fuerza [N]',
    'o2.eps',
    d,
    F
)

x = d
y = F
```

```
[A,sA,B,sB,R]=minimoscuadrados(x,y)
```

```
# Salida del programa (o2.out):
```

```
table =
```

0	47
140	60
150	61
160	61
170	62
180	63
190	64
200	65
210	66
220	67
230	68
240	69
250	70
260	70
270	71
280	72
290	73
300	74

```
g = 9.7800
```

```
m =
```

0
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290
300

```
l =
```

47
60
61
61
62
63
64

```
65
66
67
68
69
70
70
71
72
73
74

y0 = 47
m_kg =

    0
0.1400
0.1500
0.1600
0.1700
0.1800
0.1900
0.2000
0.2100
0.2200
0.2300
0.2400
0.2500
0.2600
0.2700
0.2800
0.2900
0.3000

d_cm =

    0
13
14
14
15
16
17
18
19
20
21
22
23
23
24
25
26
27

d_mt =
```



0  
0.1300  
0.1400  
0.1400  
0.1500  
0.1600  
0.1700  
0.1800  
0.1900  
0.2000  
0.2100  
0.2200  
0.2300  
0.2300  
0.2400  
0.2500  
0.2600  
0.2700

F =

0  
1.3692  
1.4670  
1.5648  
1.6626  
1.7604  
1.8582  
1.9560  
2.0538  
2.1516  
2.2494  
2.3472  
2.4450  
2.5428  
2.6406  
2.7384  
2.8362  
2.9340

d =

0  
0.1300  
0.1400  
0.1400  
0.1500  
0.1600  
0.1700  
0.1800  
0.1900  
0.2000  
0.2100  
0.2200  
0.2300

0.2300  
0.2400  
0.2500  
0.2600  
0.2700

f = 1

x =

0  
0.1300  
0.1400  
0.1400  
0.1500  
0.1600  
0.1700  
0.1800  
0.1900  
0.2000  
0.2100  
0.2200  
0.2300  
0.2300  
0.2400  
0.2500  
0.2600  
0.2700

y =

0  
1.3692  
1.4670  
1.5648  
1.6626  
1.7604  
1.8582  
1.9560  
2.0538  
2.1516  
2.2494  
2.3472  
2.4450  
2.5428  
2.6406  
2.7384  
2.8362  
2.9340

xx =

0  
0.016900  
0.019600  
0.019600  
0.022500

0.025600  
0.028900  
0.032400  
0.036100  
0.040000  
0.044100  
0.048400  
0.052900  
0.052900  
0.057600  
0.062500  
0.067600  
0.072900

yy =

0  
1.8747  
2.1521  
2.4486  
2.7642  
3.0990  
3.4529  
3.8259  
4.2181  
4.6294  
5.0598  
5.5093  
5.9780  
6.4658  
6.9728  
7.4988  
8.0440  
8.6084

xy =

0  
0.1780  
0.2054  
0.2191  
0.2494  
0.2817  
0.3159  
0.3521  
0.3902  
0.4303  
0.4724  
0.5164  
0.5624  
0.5848  
0.6337  
0.6846  
0.7374  
0.7922

```
n = 18
sx = 3.3700
sy = 36.577
sxx = 0.7005
syy = 82.602
sxy = 7.6059
D1 = 1.2521
D2 = 148.94
A = -7.6468e-03
B = 10.895
Y =
```

```
-7.6468e-03
1.4087e+00
1.5176e+00
1.5176e+00
1.6265e+00
1.7355e+00
1.8444e+00
1.9534e+00
2.0623e+00
2.1713e+00
2.2802e+00
2.3892e+00
2.4981e+00
2.4981e+00
2.6071e+00
2.7160e+00
2.8250e+00
2.9339e+00
```

```
d =
```

```
7.6468e-03
-3.9453e-02
-5.0599e-02
4.7201e-02
3.6055e-02
2.4909e-02
1.3763e-02
2.6166e-03
-8.5295e-03
-1.9676e-02
-3.0822e-02
-4.1968e-02
-5.3114e-02
4.4686e-02
3.3540e-02
2.2394e-02
1.1248e-02
1.0154e-04
```

```
dd =
```

```
5.8474e-05
1.5565e-03
```

```
2.5602e-03
2.2279e-03
1.3000e-03
6.2045e-04
1.8941e-04
6.8468e-06
7.2752e-05
3.8713e-04
9.4998e-04
1.7613e-03
2.8211e-03
1.9968e-03
1.1249e-03
5.0148e-04
1.2651e-04
1.0311e-08
```

```
sdd = 0.018262
s2 = 1.1414e-03
sA = 0.025270
sB = 0.1281
EA = 330.46
EB = 1.1758
R = 0.9989
A = -7.6468e-03
sA = 0.025270
B = 10.895
sB = 0.1281
R = 0.9989
```