

# Ajuste de datos a modelos físicos usando Mathematica

Peñuñuri, F.<sup>1</sup>, Zambrano, M.<sup>1</sup>, Acosta, C.<sup>1</sup>, Zaragoza, N.<sup>1</sup>

*Fecha de recepción: 21 de mayo de 2009 – Fecha de aceptación: 24 de marzo de 2010*

## RESUMEN

En el análisis de datos experimentales realizar ajustes a modelos físicos es de gran importancia. Con frecuencia los datos se ajusta a una expresión lineal donde los parámetros de ajuste se determinan por el método de mínimos cuadrados, este método también puede ser usado para realizar los ajustes no lineales, pero cuando se desconoce la solución del modelo, el ajuste es prácticamente imposible y no se cuenta con una manera clara de proceder.

En este trabajo se desarrolla un programa de ajuste no lineal de datos, a modelos más que a expresiones analíticas, para ello se utiliza el software Mathematica. Se analizan tres casos, una partícula sujeta a una fuerza constante, un modelo de difusión y ajustes al campo de temperaturas en el calentamiento modulado de una lámina de grafito. Por si mismo, el hecho de lograr los ajustes sin necesidad, a priori, de la solución del modelo merece mención, aunado a ello, la metodología que se usa es fácilmente aplicable a otros modelos sin requerir gran poder de cómputo ni gran cantidad de tiempo, por lo que el presente trabajo resulta de mucha utilidad para toda persona interesada en ajustes no lineales a modelos.

**Palabras clave:** ajustes no lineales, Mathematica.

## Data fit to physical models using Mathematica

## ABSTRACT

When the analysis of experimental data is required, the fit to physical model is of major importance. Frequently, the linear fit is used but this is only feasible if the phenomena under study accept a linear solution, the fit is done through the least square approach. The least square method can also be used in order to make a non-linear fit, but when the solution of the model is unknown, the non-linear fit is difficult to do if not impossible.

In this article we present a program that processes non-linear data fits. In itself, the fact of making the fit without a previous knowledge of the analytical expression deserve consideration, more over that the methodology used is easy and cheap to apply, for this reason this article will be of great help in making non-linear fits to data sets. In order to make this work, we use the Mathematica software.

We analyzed the following cases: a particle under the action of a constant force, the diffusion equation and the modulated thermal excitation of a graphite plate.

**Keywords:** non-linear fits, Mathematica.

---

<sup>1</sup>Facultad de Ingeniería de la Universidad Autónoma de Yucatán, Av. Industrias No Contaminantes por Anillo Periférico Norte S/N Apdo. Postal 150 Cordemex. Tels. (999) 941-01-68, 941-01-94, 941-01-95 Fax: (999) 941-01-89 Mérida, Yucatán, México

## INTRODUCCIÓN

La creación de modelos para la descripción de un fenómeno físico permite una mejor comprensión del problema de estudio, se tendrá una fácil manipulación de los parámetros involucrados y será posible realizar predicciones del fenómeno en cuestión.

Tales modelos por lo general se presentan en forma de ecuaciones diferenciales, integrales o integrodiferenciales. Con frecuencia, es necesario realizar ajustes de un modelo a datos experimentales, esta tarea no es difícil siempre que se conozca la forma analítica de la solución del modelo, en particular Mathematica posee muchas funciones destinada a esta labor, no es así cuando la solución es desconocida, puede verse (Wolfram, 2003) o su página web mencionada en la bibliografía.

Aún cuando existe una gran cantidad de bibliografía que aborda este problema, conocido como estimación paramétrica no lineal, problemas inversos o problemas mal condicionados, (Burden y Faires, 1985), realizar los ajustes cuando la solución del modelo es desconocida es complicado, pues está la tarea extra de encontrar la solución, que por lo general, se tendrá en forma numérica.

En este trabajo se desarrolla un método sistemático para la obtención de los ajustes, realizando en forma integral las soluciones del modelo y la elección de la mejor solución. El software utilizado es Mathematica, la economía del proceso es considerable a demás de lo amigable del software que permite una programación funcional sin mayores problemas.

El trabajo se organiza de la siguiente forma: En primer lugar se muestra la manera en que se realizan los ajustes cuando se conoce la función a ajustar. En la segunda sección se presenta la manera en que se procederá para el caso en que se desconozca la solución del modelo. En la tercera sección se realizan ajustes a tres modelos: a un tiro parabólico, a un proceso de difusión y el ajuste a un campo de temperaturas de una muestra de grafito, iluminada por una fuente con frecuencia  $f$ . Finalmente, presentamos las conclusiones y se proporciona los programas de ajuste en los apéndices.

### AJUSTES: FORMA ANALÍTICA CONOCIDA

En este trabajo se establece la diferencia entre el concepto de modelo y su solución, aunque en la literatura muchas veces se toman como sinónimos.

Se comenzará describiendo, de manera general, la forma de proceder en el ajuste de datos cuando se conoce la función a la que se desean ajustar.

En aras de la concreción, sea

$$F(x; a_1, a_2, \dots, a_n) = 0, \quad (1)$$

la función a ajustar, ésta depende de la variable independiente  $x$  y de los parámetros  $a_i$ . Se construye un vector error  $E$  cuya componente  $i$ -ésima estará definida como:

$$E_i = F(x_i; a_1, a_2, \dots, a_n) - D_i \quad (2)$$

donde  $D_i$  es el valor experimental correspondiente a la variable independiente  $x_i$ . La solución buscada resultará de encontrar el mejor conjunto de parámetros que minimice la norma del vector error.

Para realizar la minimización de la norma euclidiana, método conocido como de mínimos cuadrados, existe un gran número de software comercial. La elección de esta norma obedece a su fácil tratamiento analítico, no así por ejemplo la norma de suma ya que habrá que minimizar la suma de los valores absolutos de las componentes del vector error lo cual no es, en general, fácil de hacer. En este trabajo también se minimiza la norma euclidiana del vector  $E$ .

## METODOLOGÍA

### 2.1. AJUSTES A MODELOS

Al realizar análisis metodológico a modelos, es necesario determinar sus soluciones con el fin de proceder a los ajustes de la manera descrita en la sección anterior. Para resolver este problema, se usa el gran potencial que tiene Mathematica para realizar programación funcional. Se construye una función solución del modelo para proceder a la variación paramétrica de datos como ya se mencionó. La figura (1) muestra el procedimiento en forma esquemática.

Los puntos generales en la realización de los ajustes son:

- Decir a Mathematica cuál será la solución del modelo.
- Decir a Mathematica cuáles son las derivadas, respecto a los parámetros de ajuste de la función construida en el punto anterior.
- Ejecutar los ajustes usando por ejemplo el comando NonLinearRegress.

Es importante señalar la versatilidad de la metodología anterior aplicada en Mathematica pues no sólo sirve para modelos descritos en forma de ecuaciones diferenciales sino también integrales e

integrodiferenciales. Los comandos básicos que se utilizan para llevar a cabo esta tarea son

NonLinearRegress y NonlinearFit, (Wolfram, 2003). En los apéndices se pueden ver los detalles técnicos.

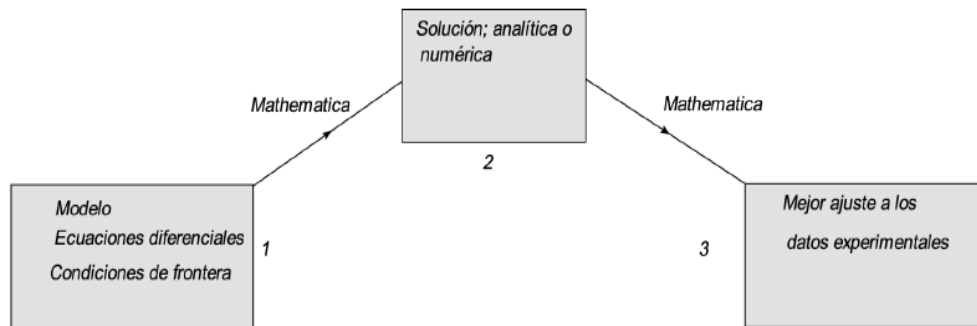


Figura 1: Diagrama del procedimiento para los ajustes.

Cabe aclarar que la manipulación numérica de las condiciones de frontera (CF), es un trabajo formidable pues no se aceptan discontinuidades que analíticamente podrían ser tratadas. Así pues, si se está trabajando con CF, éstas deben ajustarse de tal forma que sean consistentes, aquí radica un problema crucial ya que se debe asegurar que al modificar ligeramente las condiciones de frontera, para permitir el tratamiento numérico, el problema sigue siendo muy parecido al original cosa bastante plausible pero que no siempre se cumple.

## RESULTADOS Y DISCUSIÓN

Se presenta en esta sección el ajuste a tres modelos: una partícula sujeta a una fuerza constante, un proceso de difusión y a justes a un campo de temperaturas de una muestra de grafito, iluminada por una fuente con frecuencia  $f$ . Los dos primeros con el fin de ilustrar la metodología empleada pues sus soluciones son conocidas y es posible verificar la validez de los ajustes.

### PARTÍCULA SUJETA A UNA FUERZA CONSTANTE

De la segunda ley de Newton, si una partícula de masa  $m$  está sujeta a una fuerza constante entonces:

$$\ddot{\mathbf{r}}(t) = (0, -g) \quad (3)$$

éste será el modelo al cual se ajustarán los datos. Las con condiciones iniciales son:

$$\mathbf{r}(0) = (x_0, y_0) \quad (4)$$

$$\dot{\mathbf{r}}(0) = (v_{0x}, v_{0y}) \quad (5)$$

El problema acepta solución analítica,

$$\mathbf{r}(t) = \left( x_0 + v_{0x} t, y_0 + v_{0y} t - \frac{1}{2} g t^2 \right), \quad (6)$$

Con el propósito de simular datos experimentales se introduce un ruido aleatorio a la solución, para ello se le suma arbitrariamente, un numero aleatorio entre -1 y 1. En el caso del tercer ejemplo esto no es necesario pues los datos si son experimentales.

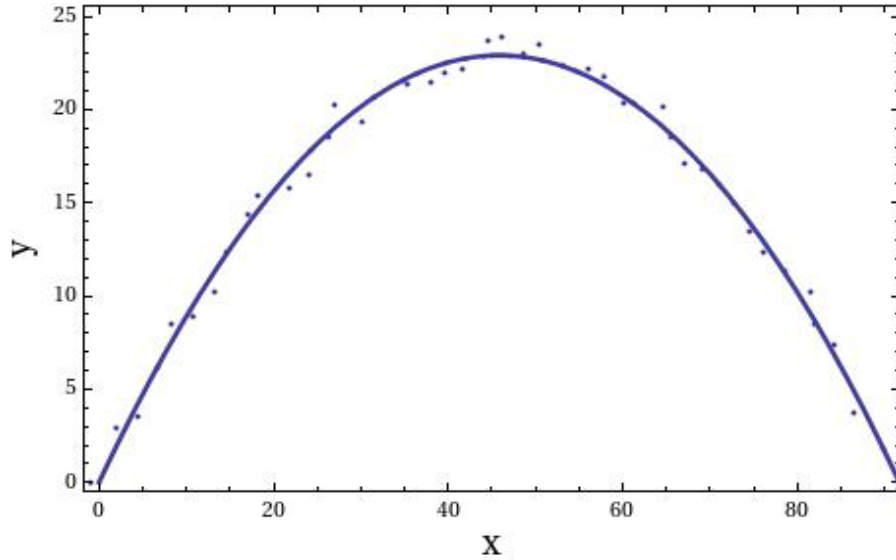
En la figura (2) se muestra el mejor ajuste y los datos simulados para un conjunto de parámetros y condiciones iniciales de  $x_0 = 0$ ,  $g = 9.8$ ,  $v_{0x} = 21.2132$ ,  $v_{0y} = 21.2132$ .

Se define en Mathematica la función:

```
f[vox_?NumberQ, voy_?NumberQ, g_?NumberQ] :=
NDSolve[{{x''[t] == 0, y''[t] == -g}, y[0] == 0, x[0] == 0,
y'[0] == voy, x'[0] == vox}, {x, y}, {t, 0, 4.5}];
```

es a esta función a la que se ajustarán los datos. El programa de ajuste se muestra en el apéndice (A), obteniendo los siguientes valores de los parámetros:  $v_{0x} = 21.168$ ,  $v_{0y} = 21.1677$ ,  $g = 9.79742$ .

Nótese que los ajustes se hacen respecto a las expresiones (3, 4, 5) no respecto a la expresión (6) puesto que en principio ésta es desconocida.



**Figura 2:** Ajuste a datos simulados.

#### AJUSTE A UN PROCESO DE DIFUSIÓN

Supóngase que se tiene un sistema que obedece la ecuación de difusión:

$$\frac{\partial u(x,t)}{\partial t} = D \frac{\partial^2 u(x,t)}{\partial x^2} \quad (7)$$

con condiciones de frontera

$$u(x,0) = x^2 \quad (8)$$

$$u(0,t) = 0 \quad (9)$$

$$u(1,t) = 0, \quad (10)$$

donde  $u$  representa la concentración de la sustancia que se difunde y  $D$  es el coeficiente de difusión.

La ecuación (7) puede resolverse por el método de separación de variables para tener:

$$u(x,t) = \sum_{n=1}^{\infty} C_n (n\pi x) e^{-D(n\pi)^2 t} \quad (11)$$

con

$$C_n = 2 \int_0^1 (n\pi x) x^2 dx. \quad (12)$$

Al igual que en el ejemplo anterior, se dará un ruido a la solución para simular datos experimentales. La suma converge rápidamente de tal forma que no hay diferencia significativa al tomar la suma hasta  $n = 3$  o  $n = 10$ , de hecho, para  $t$  y  $D$  de 0.5, la diferencia

se manifiesta más allá de la cifra decimal número 15, por mucho, más allá de la precisión experimental. Otro hecho notable es respecto a las condiciones de frontera (8) y (10), se ve que son inconsistentes aunque esta discontinuidad no afecta la solución formal de la ecuación, si afectará el tratamiento numérico de la misma.

El valor obtenido por el ajuste para el coeficiente  $D$  es de 0.501. Se muestran los detalles en el apéndice (B). Nótese el cambio en las condiciones de frontera con el fin de asegurar su continuidad, no obstante fuera del punto de discontinuidad, las CF son “prácticamente” las mismas. De nuevo, los ajustes son respecto a las expresiones (7, 8, 9, 10) y no respecto a la expresión (11).

En la figura (3), para un valor de  $D$  y  $t$  de 0.5, mostramos los datos y el ajuste correspondiente.

#### CALENTAMIENTO MODULADO DE UNA LÁMINA DE GRAFITO

En este ejemplo el ajuste no se realiza sobre un modelo que involucre alguna ecuación diferencial, sino un modelo que involucra variables complejas y que está dado por una integral.

Para este caso la solución analítica del modelo es desconocida y el ajuste de datos experimentales se realiza como se explica en la sección 2.

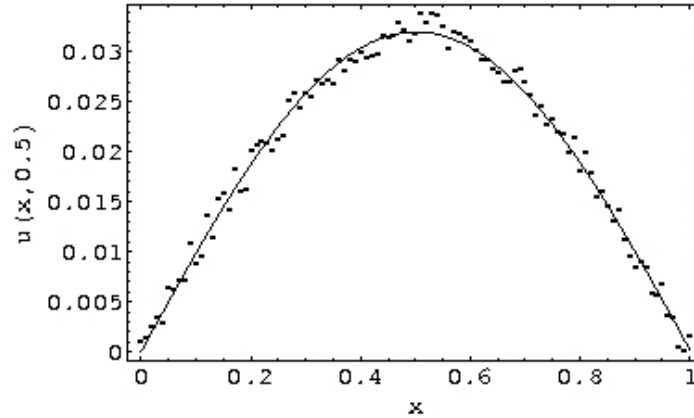


Figura 3: Ajuste y datos simulados.

En las ciencias fototérmicas frecuentemente se requiere ajustar datos al módulo o al argumento de una función compleja, por ejemplo, el campo de temperatura para una lámina de espesor  $L$  en presencia de una fuente de calor modulada a frecuencia angular  $\omega = 2\pi f$  y medida en la cara iluminada es (Salazar y Sánchez-Lavega 1998):

$$\Theta_{1D}(f) = \frac{F_0}{2k\sigma_1} \frac{1 + e^{-2\sigma L}}{1 - e^{-2\sigma L}} \quad (13)$$

donde

$$\sigma = (1 + i) \sqrt{\frac{\alpha}{\pi f}}, \quad (14)$$

$F_0$  es la cantidad de energía absorbida por el material por unidad de área por unidad de tiempo,  $\alpha$  y  $k$  son, respectivamente, la difusividad y conductividad térmica de la lámina. Este campo puede ser considerado cuando la muestra es plana, y las dimensiones y perfil de la fuente son tales que es posible suponer que el calor fluye en la dirección axial únicamente. En caso contrario y suponiendo que la fuente es un laser en su modo  $TEM_{00}$ , es necesario considerar la propagación del calor en la dirección radial teniendo en cuenta las dimensiones y el perfil gaussiano del laser. Si esto es así, el campo de temperatura en el centro del perfil gaussiano estará dado por (Fabbri y Cernuschi, 1997)

$$\Theta_{3D}(f) = \frac{F_0 R^2}{k} \int_0^\infty \frac{e^{-(\lambda R/2)^2}}{s(\lambda)} \frac{1 + e^{-2s(\lambda)L}}{1 - e^{-2s(\lambda)L}} \lambda d\lambda, \quad (15)$$

donde

$$s(\lambda) = \sqrt{\lambda^2 + \sigma^2}, \quad (16)$$

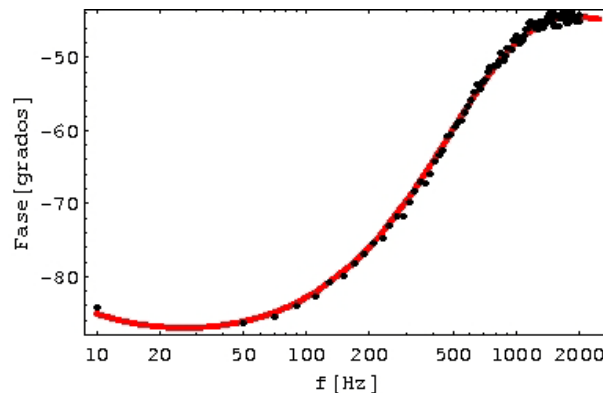


Figura 4: Fase de la señal IR de una lámina de grafito de  $188 \mu m$  como función de la frecuencia. La línea sólida representa el mejor ajuste a los datos experimentales usando la ecuación 15.

$R$  es el radio efectivo del láser y  $\lambda$  es la variable del espacio de Hankel. Es precisamente a la fase de esta función, Ec. 15, a la que se desea ajustar los datos experimentales. La figura (4) muestra la fase de la señal IR tomada de una lámina de grafito de  $188\mu m$  de espesor como función de la frecuencia de modulación. La línea representa el mejor ajuste a los datos experimentales.

Este ajuste se realizó en las versiones 5 y 6 de Mathematica, pero la versión 5 resultó mucho más eficiente (el ajuste se realizó en 3 minutos), mientras que en la versión 6 tomó cerca de una hora. Los cálculos se realizaron en una PC de escritorio con un procesador Intel Pentium Dual-Core de 2 gigas de Ram.

En el apéndice (C) presentamos el programa que

realiza el ajuste.

### **CONCLUSIONES**

Se presentó una metodología para la realización de ajustes a modelos usando Mathematica. La eficiencia y simplicidad de los ajustes mostrados es considerable. No se requiere el conocimiento previo de la solución del modelo, por lo que el trabajo presentado es de gran utilidad práctica.

Se efectuaron ajustes a tres fenómenos físicos, dos de ellos muy sencillos presentados con fines explicativos, simulando datos experimentales y un tercero con datos obtenidos de las mediciones en la forma del campo de temperaturas en una placa de grafito, para la cual no se tenía la forma analítica a la cual ajustar.

### **REFERENCIAS BIBLIOGRÁFICAS**

Página web de WOLFRAM RESEARCH.

<http://documents.wolfram.com/applications/eda/FittingDataToLinearModelsByLeast-SquaresTechniques.html>

Burden, R. L. y Faires, J. D. (1985). *Análisis Numérico*. Grupo Editorial Iberoamericana. España;

Woodbury, K. A. (2003). *Inverse Engineering Handbook*. CRC Press.

S. Wolfram, *The Mathematica Book*, 5a. Edición, Wolfram Media 2003.

A. Salazar, A. Sánchez-Lavega, J. M. Terrón. *Effective thermal diffusivity of layered materials measured by modulated photothermal techniques*, J. Appl. Phys., 84 (6), 3031, (1998).

L. Fabbri, F. Cernuschi. *Finite laser beam size effects in thermal wave interferometry*, J. Appl. Phys., 82(11), 5305, (1997).

## APÉNDICE

A. Lanzamiento parabólico. La versión de Mathematica usada es la 6.

```
(*Esta parte simula datos experimentales*)
```

```
xt = 30. Cos[45 °] t;
```

```
yt = 30. Sin[45 °] t -  $\frac{g t^2}{2}$  /. g → 9.8;
```

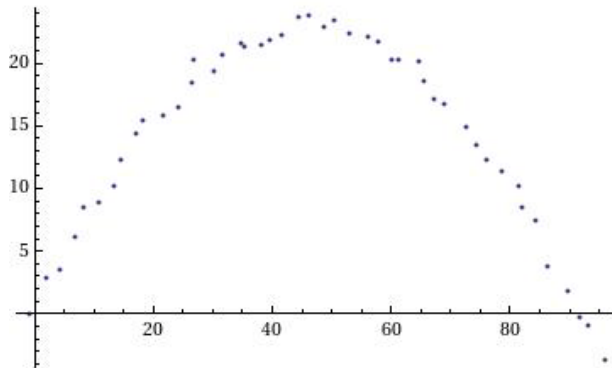
```
r[xt_, yt_] = {xt, yt};
```

```
datax = Table[{t, xt + RandomReal[{-1, 1}]}, {t, 0, 4.5, 0.1}];
```

```
datay = data = Table[{t, yt + RandomReal[{-1, 1}]}, {t, 0, 4.5, 0.1}];
```

```
randdat = Thread[{datax[[All, 2]], datay[[All, 2]]};
```

```
lp = ListPlot[randdat, AxesOrigin → {0, 0}]
```



```
(*Solución de la ED (modelo)*)
```

```
f[vox_?NumberQ, voy_?NumberQ, g_?NumberQ] :=
```

```
NDSolve[{x'[t] == 0, y'[t] == -g}, y[0] == 0, x[0] == 0, y'[0] == voy, x'[0] == vox},  
{x, y}, {t, 0, 4.5}];
```

```
S[vox_?NumberQ, voy_?NumberQ, g_?NumberQ, t_] := {x[t], y[t]} /. f[vox, voy, g][[1]];
```

```
Sx[vox_?NumberQ, t_] := S[vox, 0, 0, t][[1]];
```

```
Sy[voy_?NumberQ, g_?NumberQ, t_] := S[0, voy, g, t][[2]];
```

```
(*Paquete para el cálculo numérico*)
```

```
<< NumericalCalculus`
```

```
(*Cálculo de las derivadas de la solución*)
```

```
Sxvox[vox_?NumberQ, t_] := ND[Sx[aux, t], {aux, 1}, vox, Terms → 10];
```

```
Syvoy[voy_?NumberQ, g_?NumberQ, t_] := ND[Sy[aux, g, t], {aux, 1}, voy, Terms → 10];
```

```
Syg[voy_?NumberQ, g_?NumberQ, t_] := ND[Sy[voy, aux, t], {aux, 1}, g, Terms → 10];
```

```
(*Asignación de derivadas utilizadas por NonlinearRegress*)
```

```
Derivative[1, 0][Sx] = Sxvox;
```

```
Derivative[1, 0, 0][Sy] = Syvoy;
```

```
Derivative[0, 1, 0][Sy] = Syg;
```

```
(*Paquete para ajustes no lineales*)
```

```
<< NonlinearRegression`
```

```
(*Ajuste-x*)
```

```
ajx = NonlinearRegress[datax, Sx[vox, t], {{vox, 15.1}}, t] // First
```

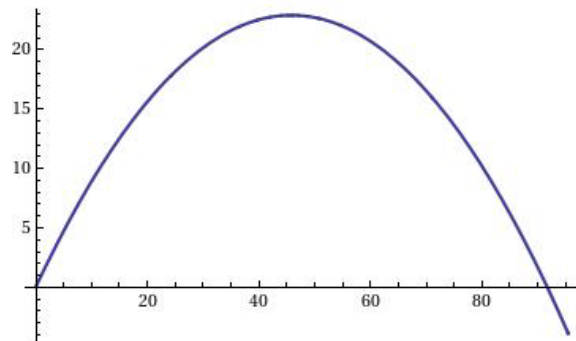
```
BestFitParameters -> {vox -> 21.2147}
```

```
(*Ajuste-y*)
```

```
ajy = NonlinearRegress[datay, Sy[voy, g, t], {{voy, 15.1}, {g, 7}}, t] // First
```

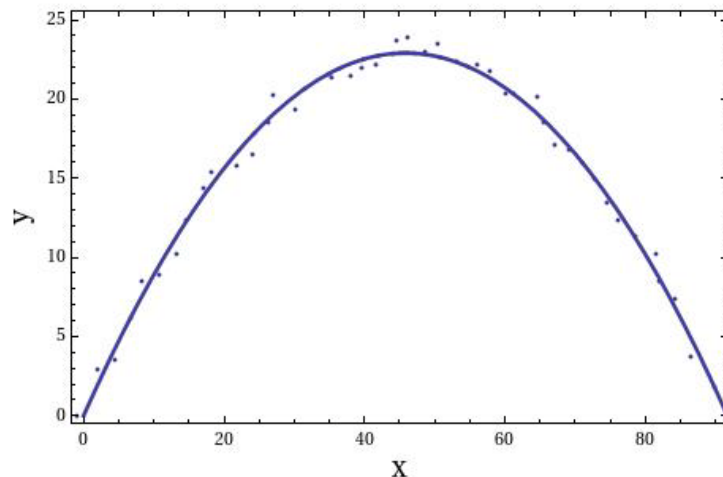
```
BestFitParameters -> {voy -> 21.1809, g -> 9.79845}
```

```
grap = ParametricPlot[S[ajx[[2]][1, 2], ajy[[2]][1, 2], ajy[[2]][2, 2], k], {k, 0, 4.5},
  AspectRatio ->  $\frac{1}{\text{GoldenRatio}}$ , PlotStyle -> {Thickness[0.006]}, AxesOrigin -> {0, 0}]
```



```
(*Gráfica de los datos y el ajuste*)
```

```
amgr = Show[lp, grap, Frame -> True, AspectRatio ->  $\frac{1}{\text{GoldenRatio}}$ , PlotRange -> {{0, 90}, {0, 25}},
  FrameLabel -> {"X", "Y"}]
```





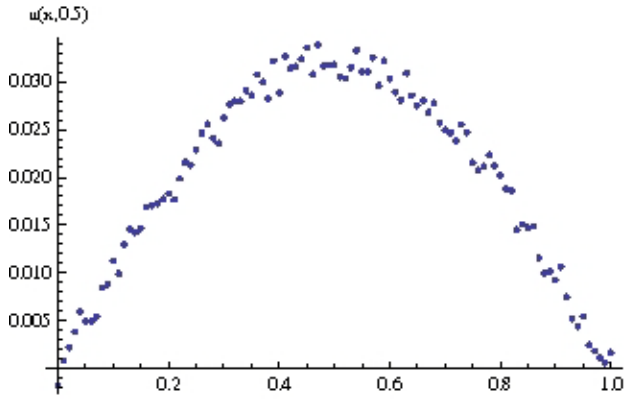
B. Proceso de difusión. La versión de Mathematica usada es la 5.

```
(*Generación de los datos*)
sol[n_] = (2) Integrate[(x^2) Sin[n Pi x], {x, 0, 1}];
fdc[x_] = Sum[sol[j] Exp[-(D) (j Pi)^2 t] Sin[j Pi x], {j, 1, 10}] /. {t -> 1/2, D -> 1/2};

dat2 = Table[{k, fdc[k] + Random[Real, {-0.002, 0.002}]}, {k, 0, 1, .01}];

(*Paquete para ajustes no lineales*)
<< Statistics'NonlinearFit'

(*Gráfica de los datos*)
g2 = ListPlot[dat2, PlotJoined -> False, AxesLabel -> {x, "u(x,0.5)"}]



(*Solución*)
fajus[d_?NumberQ] := NDSolve[{D[u[x, t], t] == (d) D[u[x, t], {x, 2}],
  u[x, 0] == x^2, u[0, t] == 0, u[1, t] == Exp[-10000 t]}, u, {x, 0, 1}, {t, 0, 1}]

f[d_?NumberQ][x_, t_] := u[x, t] /. fajus[d][[1]]

(*Paquete necesario para cálculo numérico*)
<< NumericalMath'NLimit'
```

```
fd[d_?NumberQ][x_, t_] := ND[f[aux][x, t], {aux, 1}, d, Terms -> 10]
```

```
(*Asignación de derivadas que usará NonlinearRegress*)
f/: ∂d( f[d][x_, t_] ) := fd[d][x, t]
```

```
result = NonlinearRegress[dat2, f[d][x, .5], x, {d, .3}]
```

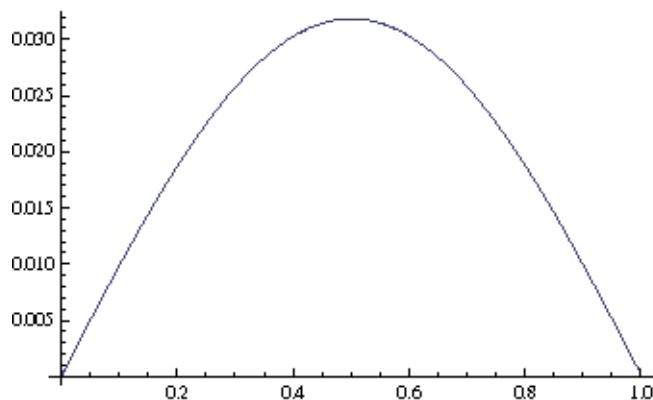
```
{BestFitParameters -> [d -> 0.502451], ParameterCITable ->
      Estimate      Asymptotic SE      CI
      d      0.502451      0.000869272      [0.500726, 0.504175]}
```

	Model	DF	SumOfSq	MeanSq
EstimatedVariance -> $1.12851 \times 10^{-6}$ , ANOVATable ->	Error	100	0.000112851	$1.12851 \times 10^{-6}$ ,
	Uncorrected Total	101	0.0510523	
	Corrected Total	100	0.0101387	

	Curvature
AsymptoticCorrelationMatrix -> ( 1. ), FitCurvatureTable -> Max Intrinsic	0
	0
	0
	95. % Confidence Region
	0.504039

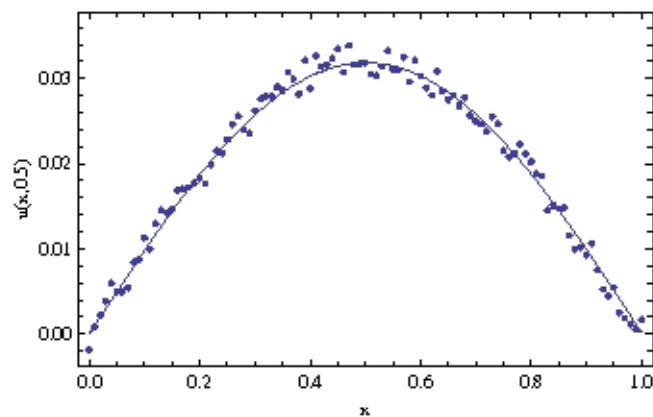
```
(*Gráfica de la solución obtenida*)
```

```
gpe1 = Plot[f[0.502451][x, .5], {x, 0, 1}]
```



```
(*Gráficas de la solución y los datos*)
```

```
gdataj = Show[g2, gpe1, Frame -> True, Background -> RGBColor[1, 1, 1],
  FrameLabel -> {" x ", "u(x,0.5)", None, None}, RotateLabel -> True,
  AspectRatio -> 1/GoldenRatio, AxesOrigin -> {0, 0}, PlotRange -> {-0.003, .037}]
```

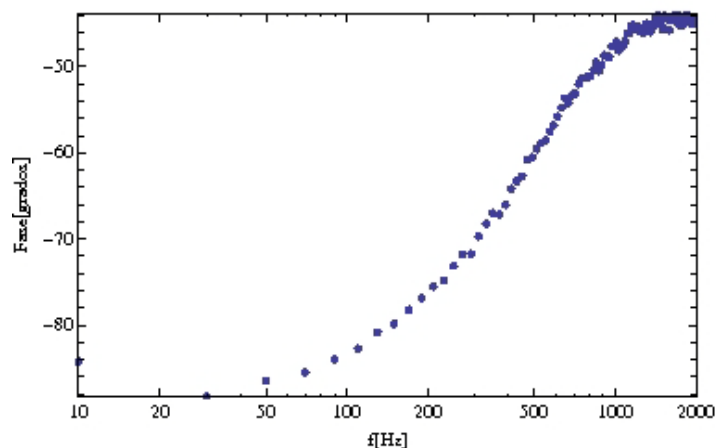


C. Calentamiento modulado de una lámina de grafito. La versión de Mathematica usada es la 5.0:

```
(*Función solución, se desea ajustar a su argumento*)
TS[r_?NumberQ, α_?NumberQ, R_?NumberQ, L_?NumberQ, f_?NumberQ, a_?NumberQ] :=
NIntegrate[
  Exp[-λ^2 R^2 / 4]
  Sqrt[λ^2 + 2 π I f / α]
  1 + Exp[-2 Sqrt[λ^2 + 2 π I f / α] L] BesselJ[0, λ r] BesselJ[1, λ a], {λ, 0, ∞}]
  1 - Exp[-2 Sqrt[λ^2 + 2 π I f / α] L]

(*Datos experimentales*)
DataL188 = {{10, -84.24'}, {30, -88.27'}, {50, -86.44'}, {70, -85.5'}, {90, -83.99'},
{110, -82.73'}, {130, -80.83'}, {150, -79.852'}, {170, -78.20'}, {190, -76.84'},
{210, -75.5072'}, {230, -74.8027'}, {250, -73.108'}, {270, -71.801'}, {290, -71.71'},
{310, -69.71'}, {330, -68.25'}, {350, -66.98'}, {370, -67.14'}, {390, -66.01'},
{410, -64.16'}, {430, -63.29'}, {450, -62.71'}, {470, -60.77'}, {490, -60.51'},
{510, -59.55'}, {530, -58.88'}, {550, -58.56'}, {570, -57.5'}, {590, -56.8'}, {610, -55.76'},
{630, -54.74'}, {650, -53.66'}, {670, -54.23'}, {690, -53.38'}, {710, -53.1'},
{730, -52.04'}, {750, -51.39'}, {770, -51.31'}, {790, -51.19'}, {810, -51.03'},
{830, -50.36'}, {850, -49.49'}, {870, -50.52'}, {890, -49.73'}, {910, -48.64'},
{930, -48.76'}, {950, -48.91'}, {970, -47.66'}, {990, -47.63'}, {1010, -47.1'},
{1030, -48.12'}, {1050, -47.42'}, {1070, -47.57'}, {1090, -47.06'}, {1110, -46.18'},
{1130, -46.13'}, {1150, -45.24'}, {1170, -45.16'}, {1190, -45.7'}, {1210, -45.26'},
{1230, -45.88'}, {1250, -45.56'}, {1270, -46.12'}, {1290, -45.84'}, {1310, -45.01'},
{1330, -45.7'}, {1350, -45.96'}, {1370, -45.57'}, {1390, -44.91'}, {1410, -44.43'},
{1430, -45.11'}, {1450, -44.04'}, {1470, -44.86'}, {1490, -45.26'}, {1510, -45.72'},
{1530, -44.11'}, {1550, -44.43'}, {1570, -44.19'}, {1590, -45.77'}, {1610, -44.65'},
{1630, -44.19'}, {1650, -44.55'}, {1670, -44.67'}, {1690, -43.84'}, {1710, -44.53'},
{1730, -44.55'}, {1750, -45.27'}, {1770, -45.29'}, {1790, -44.61'}, {1810, -44.83'},
{1830, -44.12'}, {1850, -43.88'}, {1870, -44.64'}, {1890, -44.57'}, {1910, -44.7'},
{1930, -45.04'}, {1950, -44.7'}, {1970, -44.97'}, {1990, -45.05'}, {2010, -44.32'}};
```

```
(*Paquete para graficar usando escala
logarítmica y gráfica de los datos experimentales*)
<< Graphics`Graphics`
G1 = LogLinearListPlot[DataL188, {PlotRange → All, PlotStyle → {PointSize[0.0125]},
Frame → True, FrameLabel → {"f[Hz]", "Fase[grados]"}, AspectRatio →  $\frac{1}{\text{GoldenRatio}}$ }]
```



```
(*Cálculo numérico*)
<< NumericalMath`NLimit`

(*Para evitar warnings de deletreo*)
Off[General::spell1]
```

```

(*Función a ajustar(menos una constante, F0)*)
TP[r_?NumberQ, α_?NumberQ, R_?NumberQ, L_?NumberQ, f_?NumberQ, a_?NumberQ] :=
  180/Pi Arg[TS[r, α, R, L, f, a]]

(*Derivadas de la función anterior,
en realidad solo se requiere de la derivada respecto a α y a R*)
TPr[r_?NumberQ, α_?NumberQ, R_?NumberQ, L_?NumberQ, f_?NumberQ, a_?NumberQ] :=
  ND[TP[aux, α, R, L, f, a], {aux, 1}, r, Terms→10]

TPα[r_?NumberQ, α_?NumberQ, R_?NumberQ, L_?NumberQ, f_?NumberQ, a_?NumberQ] :=
  ND[TP[r, aux, R, L, f, a], {aux, 1}, α, Terms→10]

TPR[r_?NumberQ, α_?NumberQ, R_?NumberQ, L_?NumberQ, f_?NumberQ, a_?NumberQ] :=
  ND[TP[r, α, aux, L, f, a], {aux, 1}, R, Terms→10]

(*Asignación de las derivadas, aqui se instruye a Mathematica
sobre derivadas que requerirá para hacer los ajustes*)
Derivative[0, 1, 0, 0, 0, 0][TP] = TPα;
Derivative[0, 0, 1, 0, 0, 0][TP] = TPR;

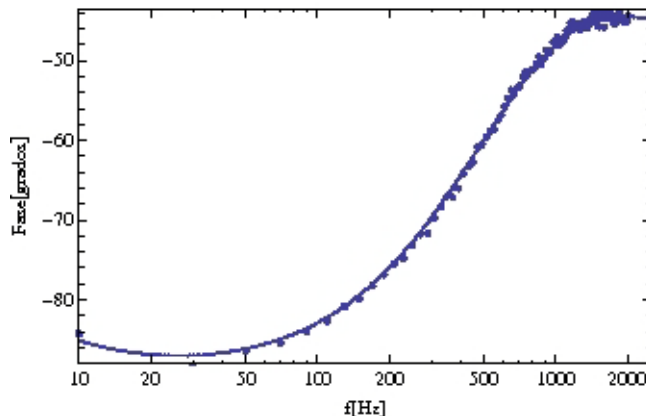
(*Paquete para realizar ajustes*)
<< Statistics'NonlinearFit'

bfp = NonlinearRegress[DataL188,
  TP[0, α, R, 0.0188, f, .05] + F0, f, {{α, .2}, {R, .2}, {F0, 1}}] // First
BestFitParameters → {α → 0.551304, R → 0.612911, F0 → -1.05859}

(*Gráfica del ajuste y los datos experimentales*)
GFittl1 = LogLinearPlot[ $\frac{180}{\pi}$  Arg[TS[0, .55, .612, .0188, f, .05]] - 1.05,
  {f, 10, 2500}, PlotStyle → {Thickness[.01], Hue[1]}, Frame → True,
  FrameLabel → {"f[Hz]", "Fase[grados]"}, AspectRatio →  $\frac{1}{\text{GoldenRatio}}$ ];

GrafitoL188 = Show[GFittl1, G1]

```



Este documento debe citarse como:

Peñuñuri, F., Zambrano, M., Acosta, C., Zaragoza, N. (2010). Ajuste de datos a modelos físicos usando **Mathematica**. Ingeniería, Revista Académica de la FI-UADY, 14-1, pp 55-66, ISSN: 1665-529-X.