



Wolfram
Language™



☛ Patrocinado por: **Wolfram Research Europe Ltd.**
Contacto para España y Portugal: cjordan@wolfram.com
<http://www.wolfram.com>

Wolfram Mathematica en la enseñanza, la investigación y la empresa

J.Guillermo Sánchez

(http://www.wolfram.com/training/instructors/guillermo_sanchez.html)

Parte del material procede del libro: *Mathematica Beyond Mathematics. The Wolfram Language in the Real World*. CRC Press

Fuentes muy recomendables: <http://mathworld.wolfram.com> y <http://functions.wolfram.com>

¿Qué es Mathematica?

Wolfram Mathematica (wolfram language) inicialmente surgió orientado al cálculo simbólico, pero desde hace años es un entorno integral de programación que puede utilizarse prácticamente en cualquier campo y área de conocimiento (lingüística, demografía, finanzas, química o lo que Ud pueda imaginar.). Permite utilizar lenguaje natural, accede en tiempo real a datos depurados de infinidad de campos, puede ejecutarse en la nube, dispone de funciones optimizadas orientadas al Machine Learning, Big data y mucho más.

¿Qué vamos a hacer?

- Algunas novedades de Mathematica aplicada a las Matemáticas
- Cómo elaborar documentos y presentaciones en Mathematica
- El formato libre y otras formas para empezar a utilizar Mathematica rápidamente. Wolfram Alpha
- Datos computable: Obteniendo y tratando datos de distintos campos (Ciencias de la vida, meteorología, economía, física y química, astronomía, etc).
- Ideas básicas de programación
- Calculo simbólico, numérico, gráficos.
- Imágenes.
- Creando aplicaciones interactivas. Geometric Region
- Probabilidad y Estadística
- Machine Learning
- Paralelización
- Aplicaciones para la nube
- WOLFRAM ENTERPRISE PRIVATE CLOUD

Algunas novedades de Mathematica aplicada a las matematicas

Evaluación de derivadas a partir de la definición

```
In[8]:= g[x_] := x^2 e^x
In[9]:= dq1[x_] = DifferenceQuotient[g[x], {x, h}]
Out[9]= 
$$\frac{e^x (e^h h^2 + e^h x^2 + 2 e^h h x - x^2)}{h}$$

In[10]:= Limit[dq1[x], h -> 0]
Out[10]=  $e^x x (x + 2)$ 
```

Ecuaciones diferenciales parciales

(<https://www.wolfram.com/language/11/partial-differential-equations/?product=mathematica>)

Generación de oscilaciones en una membrana circular

```
In[11]:= eqn = r  $\frac{\partial^2 u(r, t)}{\partial t^2} = \frac{\partial}{\partial r} \left( r \frac{\partial u(r, t)}{\partial r} \right);$ 
In[12]:= bc = u(1, t) == 0;
In[13]:= ic = {u(r, 0) == 0, u^(0,1)(r, 0) == 1};
(dsol = DSolve[{eqn, bc, ic}, u[r, t], {r, t}])
Out[14]/TraditionalForm= 
$$\left\{ \left\{ u(r, t) \rightarrow \sum_{K[1]=1}^{\infty} \frac{2 J_0(r J_{0,K[1]}) J_1(J_{0,K[1]}) \sin(t J_{0,K[1]})}{(J_0(J_{0,K[1]})^2 + J_1(J_{0,K[1]})^2) (J_{0,K[1]})^2} \right\} \right\}$$

```

Asymptotics. Nonlinear Differential Equations

Compute a polynomial approximation for a **nonlinear first-order ODE**:

```
In[1]:= eqn = {3 y'[x]^2 + 4 x y'[x] - y[x] + x^2 == 0, y[0] == 1};
In[2]:= sol = Quiet[AsymptoticDSolveValue[eqn, y[x], {x, 0, 37}]]
Out[2]= 
$$-\frac{x^2}{4} - \frac{x}{\sqrt{3}} + 1$$

```

Verify that this is an **exact solution** of the ODE:

```
In[3]:= Simplify[eqn /. {y -> Function[{x}, Evaluate[sol]]}]
Out[3]= {True, True}
```

Metaalgoritmos

<https://www.wolfram.com/algorithmbase/>

FormulaData

```
FormulaLookup["kinetic energy"]
{KineticEnergy, KineticEnergyRelativistic}
```

FormulaData["KineticEnergy"]

$$K = \frac{1}{2} m v^2$$

**In[29]:= Column[{FormulaData[{"PlanckRadiationLaw", "Wavelength"}],
FormulaData[{"PlanckRadiationLaw", "Frequency"}]}**

$$L(\lambda) = \frac{2 h c^2}{\lambda^5 \left(\exp\left(\frac{h c}{T \lambda}\right) - 1 \right)}$$

$$\text{Out[29]= } L(\nu) = \frac{\nu^3 (2 h / c^2)}{\exp\left(\frac{\nu (h / k)}{T}\right) - 1}$$

MathematicalFunctionData

**In[30]:= MathematicalFunctionData[
incomplete elliptic integral $E(z|m)$ MATHEMATICAL FUNCTION, "ArgumentSimplifications"]**

guide/TheoremProving FindEquationalProof

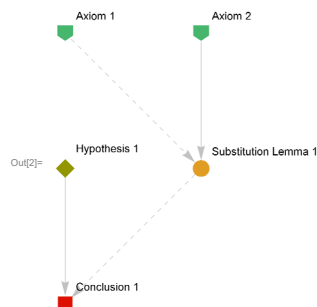
Prove an elementary theorem in propositional logic:

In[1]:= proof = FindEquationalProof[a == c, {a == b, b == c}]

Out[1]= ProofObject  **Logic: EquationalLogic** **Steps: 5**
Theorem: a == c

Show the abstract proof network:

In[2]:= proof["ProofGraph"]



Show the complete list of proof steps as a Dataset object:

In[3]:= proof["ProofDataset"]

		Statement	Proof
Axiom	1	$b == a$	
Axiom	2	$b == c$	
Hypothesis	1	$a == c$	
SubstitutionLemma	1	$a == c$	Input Position Construct Orientation 6 total >
Conclusion	1	True	Input Position Construct Orientation 6 total >

Prove a theorem involving universal quantification in first-order logic:

```
In[1]:= proof = FindEquationalProof[ForAll[x, f[g[x]] == g[f[x]]], ForAll[x, f[x] == g[x]]]
```

```
Out[1]= ProofObject[ Logic: EquationalLogic Steps: 4  
Theorem:  $\forall x, f[g[x]] == g[f[x]]$ ]
```

Typeset a natural language argument:

```
In[2]:= proof["ProofNotebook"]
```

```
Out[2]= 
Axiom 1
We are given that:
 $f[x] == g[x]$ 
Hypothesis 1
We would like to show that:
 $f[g[x]] == g[f[x]]$ 
Substitution Lemma 1
It can be shown that:
 $f[g[x]] == f[f[x]]$ 
PROOF
We start by taking Hypothesis 1, and apply the substitution:
 $g[x_] \rightarrow f[x]$ 
which follows from Axiom 1.
Conclusion 1
We obtain the conclusion:
True
PROOF
Take Substitution Lemma 1, and apply the substitution:
 $g[x_] \rightarrow f[x]$ 
which follows from Axiom 1.
```

Build a function that performs the steps of the proof:

```
In[3]:= pf = proof["ProofFunction"]
```

```
Out[3]= Function[{ },
Block[{EquationalProof`Axiom, EquationalProof`Hypothesis, EquationalProof`CriticalPairLemma,
EquationalProof`SubstitutionLemma, EquationalProof`Conclusion}, Axiom1 = f[x] == g[x];
Hypothesis1 = f[g[x]] == g[f[x]];
SubstitutionLemma1 = MapAt[Replace[g[x_] → f[x]], Hypothesis1, 2];
Conclusion1 = MapAt[Replace[g[x_] → f[x]],
SubstitutionLemma1, FirstPosition[SubstitutionLemma1, g[x_]]];
Conclusion1]
```

```
In[4]:= pf[]
```

```
Out[4]= True
```

Algunas cosas que quizás nunca imaginó que Mathematica podía hacer:

<https://www.wolfram.com/language/11/multilingual-functionality/?product=mathe->

matica

```
In[23]:= string = "Привет, как дела?";
```

```
In[24]:= origin = LanguageIdentify[string]
```

```
Out[24]= Russian
```

```
In[26]:= Transliterate[string, origin -> "Spanish"]
```

```
Out[26]= Privet, kak dela?
```

Edición de documentos. Estilos

Cada *notebook* de *Mathematica* es un documento completo e interactivo que combina texto, tablas, gráficos, cálculos y otros elementos, que permite editar documentos con una excelente calidad, listos para ser publicados (algunas revistas admiten directamente los documentos en formato de *Mathematica*).

Desde la paleta **Witting Assistant** o desde la barra de menu: **Format ► Text Color** podemos seleccionar la fuente, **aspecto**, tamaño, **color**, etc. Podemos usar caracteres especiales como $\frac{1}{2}$.

Cuando vaya a escribir un nuevo *notebook* elija el estilo desde el principio con **Format ► Stylesheet ► Stylesheet Chooser...** o utilizando la paleta **Witting Assistant**. Pulse en la miniatura con plantilla con el estilo elegido (New), podrá utilizar esta plantilla para ir la modificando adaptando el contenido a sus necesidades.

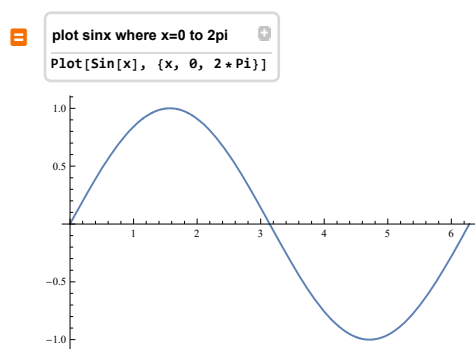
Si va a preparar un documento para una presentación puede elegir entre distintos estilos **Format ► Stylesheet ► SlideShow**.

El formato libre y otras formas para empezar a utilizar Mathematica rápidamente

Recuerde: Para ejecutar una celda *Input* seleccione la celda (poniendo el cursor del ratón en la marca de celda) y pulse `[⇧]+ ⏏` o **[Entrar]** en el teclado numérico o en el menú contextual seleccione "Evaluate Cell". Para acceder directamente a WolframAlpha escriba en una celda tipo input un doble igual, esto es ==

El formato libre

Escriba lo que quiere (formato lingüístico) (EN UNA CELDA TIPO INPUT use doble == o Shift+Enter). Tome la salida como plantilla y adapte la a sus necesidades. Puede probar con otras expresiones equivalentes (*graph of the sine of x, visualize the sine of x, ..*) y obtendrá el mismo resultado.



Pulse sobre la salida y obtendrá el input de *Mathematica* con la sintaxis adecuada.

integral of 1/(1 + x^3)

```
Integrate[1 / (1 + x^3), x]
```

$$\frac{\text{ArcTan}\left[\frac{-1+2x}{\sqrt{3}}\right]}{\sqrt{3}} + \frac{1}{3} \log[1+x] - \frac{1}{6} \log[1-x+x^2]$$

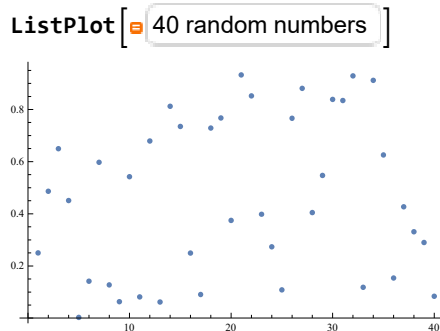
Puede usarlo mas allá de las matemáticas.

melting point of caffeine

```
ChemicalData["Caffeine", "MeltingPoint"]
```

235.25 °C

Con **Insert-> Inline Free-form** podemos usar el formato lingüístico en expresiones. Por ejemplo escribe: “40 random numbers” dentro de una función standard de *Mathematica* como ListPlot



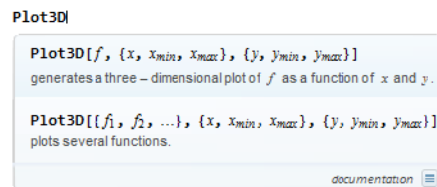
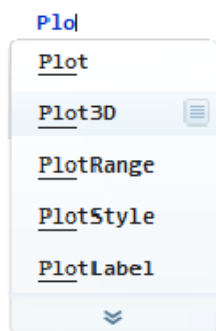
Entradas con el asistente de entrada contextual

Tal cual escribimos una entrada se van mostrando las funciones más frecuentes que empiezan por las mismas letras. Para una misma función se nos muestra varias plantillas relacionadas (Crt+May+K).

Auto completion

Function Templates

Dynamic Highlighting



`Plot3D[Sin[x y],
{x, 0, 2 π}, {y, 0, 10}]`

Un código de colores permite detectar errores de sintaxis (según vaya escribiendo vera como le va desplegando la ayuda)

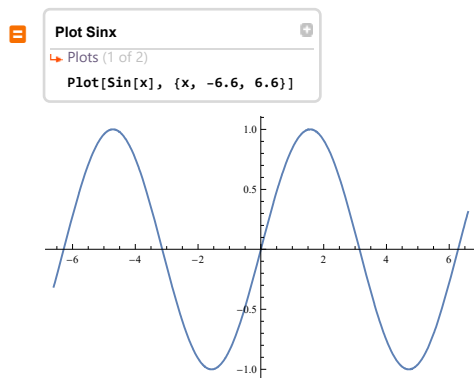
`Plot3D[Sin[], {x, xmin, xmax}, {y, ymin, ymax}, Opti]`

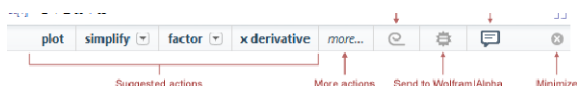
Interface predictiva (IP)

Al ejecutar una sentencia la interface predictiva muestra tras el output una barra de sugerencias

<http://reference.wolfram.com/mathematica/guide/WolframPredictiveInterface.html>

Escribiendo = al inicio de una celda podemos utilizar **formato lingüístico** que podemos combinar con la interface predictiva (IP)





Utilizar ejemplos incluidos (ExampleData)

Descargamos el “Origen de las especies” de Darwin (se incluye como ejemplo pero puede obtener desde varios sitios de Internet).

```
txt = ExampleData[{"Text", "OriginOfSpecies"}];
```

Nos preguntamos cuantas veces aparece la palabra “evolution” y la comparamos con las veces que aparece “selection”

```
StringCount[txt, "evolution"]
```

```
4
```

```
StringCount[txt, "selection"]
```

```
351
```

El uso de peletas (Palettes) es de gran ayuda, especialmente cuando queremos que las entradas se parezcan a la notación habitualmente utilizada en matematicas

Como ejemplo, en una celda input escriba (ayúdese de una paleta para transcribir el contenido) en la forma estándar lo siguiente y ejecútela.

$$\partial_{x,y} \sin[x \sqrt{y}]$$

$$\frac{\cos[x \sqrt{y}]}{2 \sqrt{y}} - \frac{1}{2} x \sin[x \sqrt{y}]$$

El cálculo anterior también podemos mostrarlo en notación clásica de las matemáticas. En *Mathematica* se denomina *TraditionalForm*.

Copie la celda *input* de arriba en una nueva celda y en la barra de menú eleja **Cell**

► **Convert To ► TraditionalForm** o añada `// TraditionalForm` para indicar que el resultado también se nos muestre en notación tradicional. Observe que la marca de celda aparece con un corchete serpenteado. El mismo menú podrá utilizarlo para conversión a otras formas.

$$\frac{\partial^2 \sin(x \sqrt{y})}{\partial x \partial y}$$

$$\frac{\cos(x \sqrt{y})}{2 \sqrt{y}} - \frac{1}{2} x \sin(x \sqrt{y})$$

La forma tradicional anterior ocupa muchos recursos de ordenador. Además puede que le dé problemas de conversión si el fichero es leído en una versión anterior de *Mathematica*. Es recomendable usar la forma *StandardForm* o *InputForm* y reservar la tradicional para aquellos casos en los que se requiera editar el documento en alta calidad.

Wolfram Alpha

In[27]:=  Integrate x e^x

Indefinite integrals:

Hide steps +

$$\int x e^x dx = e^x (x - 1) + \text{constant}$$

Possible intermediate steps:

Take the integral:

$$\int e^x x dx$$

For the integrand $e^x x$, integrate by parts, $\int f dg = fg - \int g df$, where

$$f = x, \quad dg = e^x dx,$$

$$df = dx, \quad g = e^x:$$

$$= e^x x - \int e^x dx$$

The integral of e^x is e^x :

$$= e^x x - e^x + \text{constant}$$

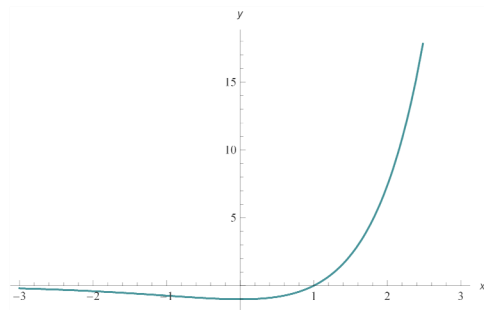
Which is equal to:

Answer:

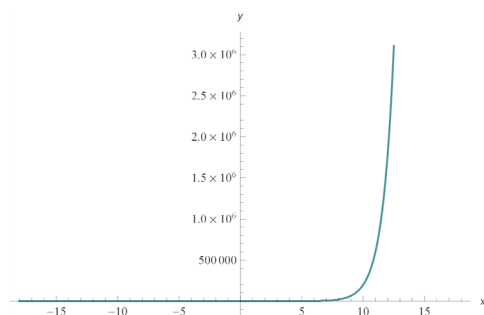
$$= e^x (x - 1) + \text{constant}$$

Plots of the integral:

+



min  max 



min  max 

Expanded form of the integral:

Step-by-step solution +

$$e^x x - e^x + \text{constant}$$

Series expansion of the integral at x=0:

$$-1 + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{8} + O(x^5)$$

(Taylor series)

[Big-O notation »](#)

Definite integral:

$$\int_{-\infty}^0 e^x x dx = -1$$

WolframAlpha 

Datos computables: Obteniendo y tratando datos de distintos campos (Ciencias de la vida, meteorología, economía, física y química, astronomía, etc).

In[28]= ? *Data

Información en tiempo real

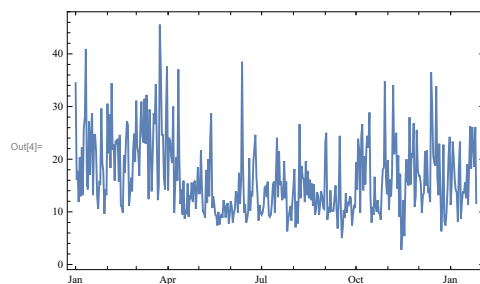
Existen funciones que acceden a datos en tiempo real, por ejemplo (es la temperature del lugar (ciudad) que estamos:

```
AirTemperatureData[]
```

20. °C

Podemos elegir un lugar concreto y analizar datos historicos

```
In[4]= DateListPlot[
  WeatherData[Santander CITY ☒, "WindSpeed", {{2018, 1, 1}, {2019, 01, 31}, "Day"}]]
```



Ideas básicas de programación: Reglas fundamentales

1. Funciones: Mayúsculas en los nombres de funciones: Cos, Sin.

Mathematica dispone de miles de funciones, todas empiezan por una letra mayúscula. Si no conocemos el nombre exacto de una función pero sabemos las letras por las que empieza tecleelas en una celda tipo *Input* y el asistente de ayuda contextual desplegará las instrucciones de *Mathematica* que empiezan por esas letras, pulse sobre la función elegida y se le mostrará una plantilla. También puede utilizar la entrada en formato lingüístico o la barra de herramientas.

2. [] : Argumentos, lo que se desea calcular

```
D[E^Cos[2 Pi x], x]
-2 e^Cos[2 π x] π Sin[2 π x]
```

La misma función escrita utilizando la paleta (utilizando el formato estandar de *Mathematica*) :

```
∂x eCos[2 π x]
-2 eCos[2 π x] π Sin[2 π x]
```

Para escribir π , e se utilizan símbolos especiales (e no es lo mismo que e , e es una letra sin otro significado) para lo que emplearemos la paletas o podemos escribir Pi por π y $\text{E}^{\text{}}$ o $\text{Exp}[]$ por e

3. { } - listas o rangos

```
list = {3, Pi/2, 4 Pi};
```

Si una función es listable podrá introducir dentro de ellas una lista y la función se aplicará a cada uno de los elementos de la lista

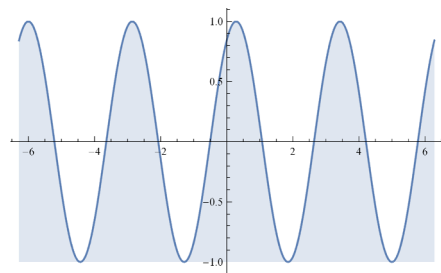
```
Attributes[Sin]
{Listable, NumericFunction, Protected}
Sin[list]
{Sin[3], 1, 0}
```

4. Opciones

Las funciones incluyen tienen opciones (si no se indican el programa usa las que tiene por defecto). Una forma de ver directamente las opciones de una función es

```
Options[func]
Options[Plot]
```

`Plot[ Sin[2 x + 1], {x, - 2 Pi, 2 Pi}, Filling → Bottom]`



Funciones (o instrucciones)

Mathematica dispone de miles de funciones que a su vez pueden ser combinadas para construir funciones más complejas

Mathematica permite definir funciones de forma muy parecida al uso habitual en matemáticas. La sintaxis habitual en el caso de una función de una variable es: $f[x_] := \text{expr}(x)$. El *blank* ("_") junto a la variable sirve para especificar lo que sería la variable independiente de la función.

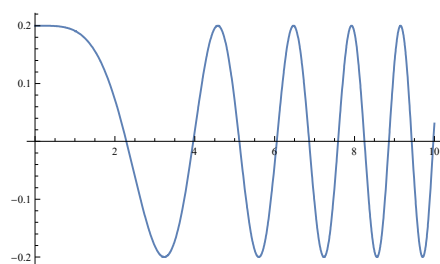
```
f[x_] := 0.2 Cos[0.3 x^2]
```

Ahora podemos asignar un valor a la variable independiente y obtendremos el valor de la función.

```
f[3]
-0.180814
```

Aquí la utilizamos para representarla

```
Plot[f[x], {x, 0, 10}]
```

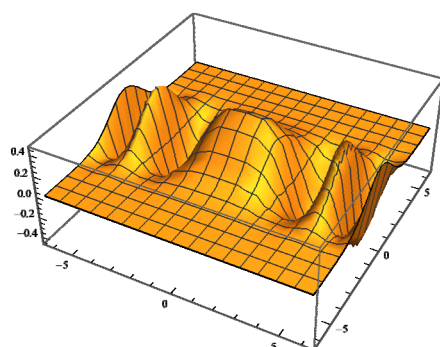


El mismo concepto puede generalizarse a varias variables. En este ejemplo utilizamos la función $f(x)$ antes definida.

```
g[x_, y_] := f[x] 2.3 Exp[-0.3 y^2]
```

Ahora podemos presentar el resultado en un gráfico 3D para lo que podemos utilizar `Plot3D`. Utilizamos la opción `PlotRange->All` para mostrar los puntos del rango para el que se calcula la función, elimine la opción para comprobar lo que sucede.

```
Plot3D[g[x, y], {x, -2 Pi, 2 Pi}, {y, -2 Pi, 2 Pi}, PlotRange -> All]
```



☞ Función

$$f(x_, y_) = \sqrt{\log(y - x^2)} ;$$

☞ Dominio

`FunctionDomain[f[x, y], {x, y}]`

$$y > 0 \wedge \log(y - x^2) \geq 0 \wedge x^2 - y < 0$$

Cálculos, ...

Solve que da las soluciones mas habituales.

```
sol1 = Solve[{a x + b y == 3, 2 x + 3 y == 7}, {x, y}]
```

$$\left\{ \left\{ x \rightarrow -\frac{-9 + 7b}{3a - 2b}, y \rightarrow -\frac{6 - 7a}{3a - 2b} \right\} \right\}$$

```
sol1[[1, 1]]
```

$$x \rightarrow -\frac{-9 + 7b}{3a - 2b}$$

Si pretendemos obtener todas las soluciones posibles utilizaremos **Reduce**.

A veces no es posible encontrar una solución general y entonces es necesario establecer el dominio para la solución buscada, en el ejemplo se establece como dominio los números reales.

```
Reduce[ $\sqrt{x} \sqrt{27^{2x-1}} == \sqrt{9^{2x-1}}$ , x, Reals]
```

$$x == \frac{1}{2} \mid \mid x == 3$$

Soluciones geométricas

```
Solve[{x, y} ∈ InfiniteLine[{{0, 0}, {2, 1}}] && {x, y} ∈ Circle[], {x, y}]
```

$$\left\{ \left\{ x \rightarrow -\frac{2}{\sqrt{5}}, y \rightarrow -\frac{1}{\sqrt{5}} \right\}, \left\{ x \rightarrow \frac{2}{\sqrt{5}}, y \rightarrow \frac{1}{\sqrt{5}} \right\} \right\}$$

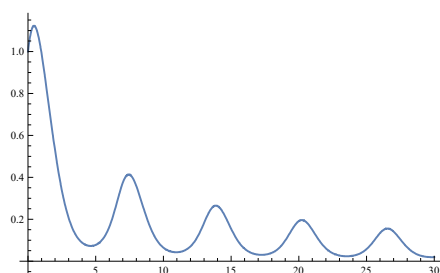
Cálculo numérico

La potencia en cálculo numérico es enorme, como ejemplo veamos alguna ecuación diferencial resuelta numericamente.

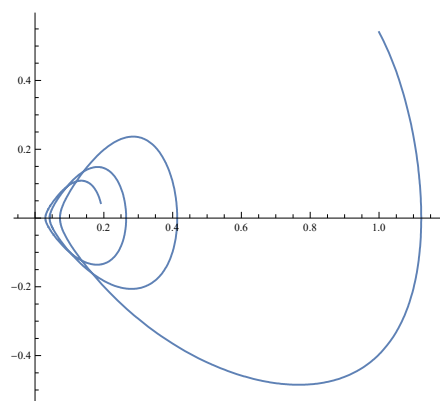
```
s = NDSolve[{y'[x] == y[x] Cos[x + y[x]], y[0] == 1}, y, {x, 0, 30}]
```

```
{ {y -> InterpolatingFunction[ Domain: {{0, 30}} Output: scalar] ] }
```

```
Plot[Evaluate[y[x] /. s], {x, 0, 30}, PlotRange -> All]
```



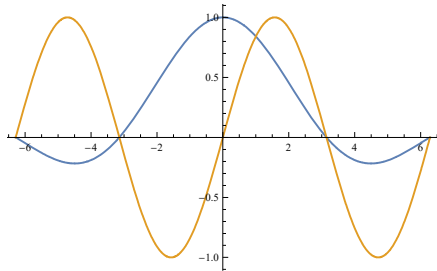
```
ParametricPlot[Evaluate[{y[x], y'[x]} /. s], {x, 0, 20}]
```



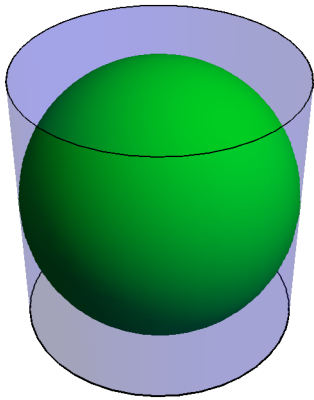
Aprovecha las posibilidades gráficas

Muestre varios gráficos

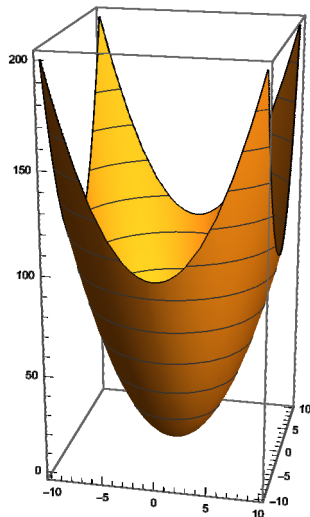
```
Plot[{Sin[x], Sin[x]/x}, {x, -2 Pi, 2 Pi}]
```



```
Graphics3D[{{Opacity[0.2], Blue, Cylinder[]}, {Green, Sphere[]}}, Boxed → False]
```

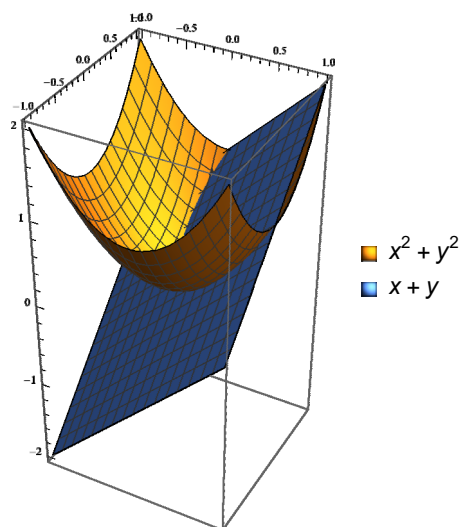


$$f(x, y) = x^2 + y^2;$$



Utilice leyendas

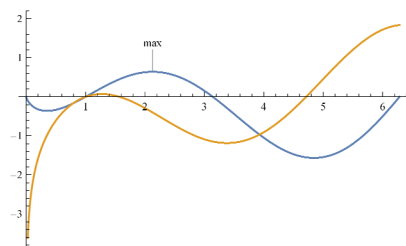
```
Plot3D[{x^2 + y^2, x + y}, {x, -1, 1}, {y, -1, 1},
BoxRatios -> Automatic, PlotLegends -> "Expressions"]
```



Etiquetas

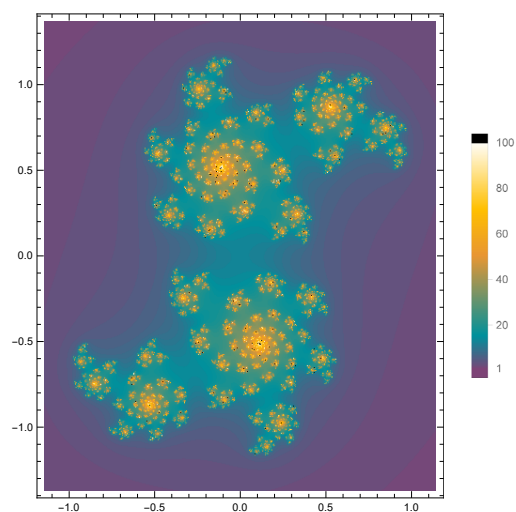
Coloque etiquetas donde quiera

```
Plot[{Callout[Log[x] Sin[x], "max", Above], Log[x] Cos[x]}, {x, 0, 2 π}]
```

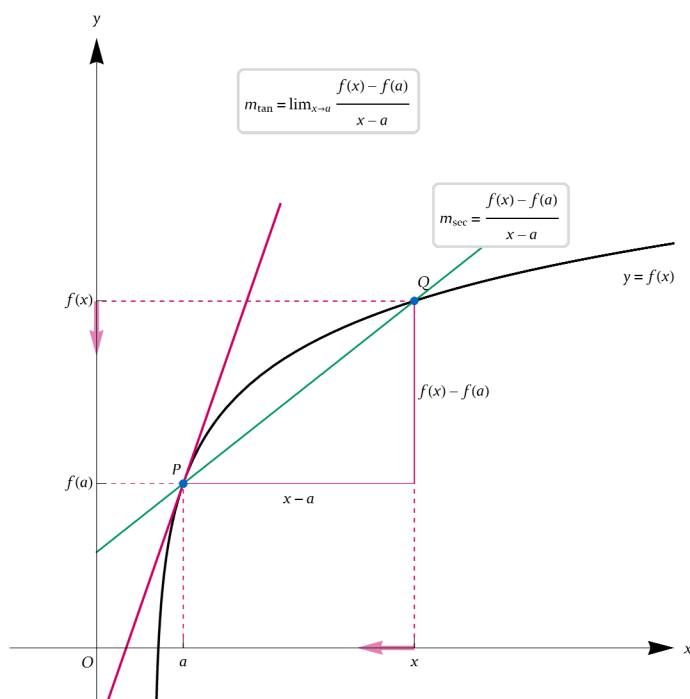


Graficos especificos

```
JuliaSetPlot[0.365 - 0.4 i, PlotLegends -> Automatic]
```



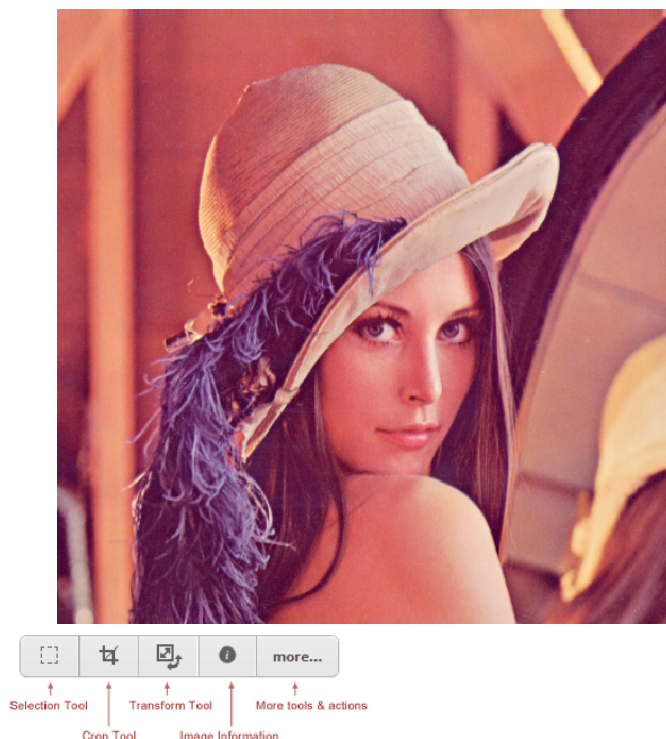
Puede utilizar Control D para elaborar un gráfico sofisticado



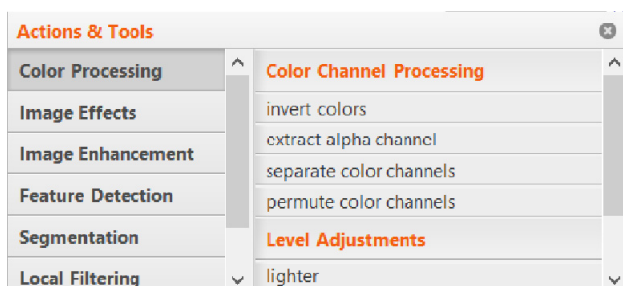
Visualizacion: Interface predictiva e imágenes

El tratamiento de imágenes también ha experimentado una clara mejora en las últimas versiones de *Mathematica*. Desde *Mathematica* 9 cuando pulse en una imagen verá que justo debajo de la misma se muestra un menu que le dará numerosas posibilidades de actuar sobre la imagen.

```
ExampleData[{"TestImage", "Lena"}]
```



Pulse en los diferentes iconos. Por ejemplo, para **more...** se le desplegará el siguiente menu:



ImageRotate[% , Right]



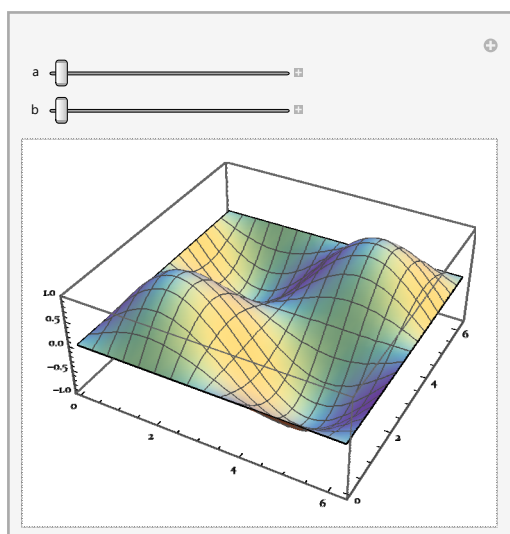
MODELIZACION CON Manipulate

(<http://demonstrations.wolfram.com>)

Ejemplo básico

En este caso creamos un gráfico mas complejo en el que dejamos que el usuario pueda modificar los parámetros a y b

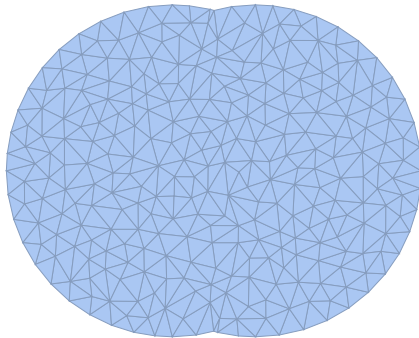
```
Manipulate[  
  Plot3D[Sin[x + a] Sin[y + b^2], {x, 0, 2 Pi},  
    {y, 0, 2 Pi}, PlotRange -> 1, PlotPoints -> 20, Mesh -> True],  
  {a, 0, 2 Pi}, {b, 0, 2 Pi}]
```



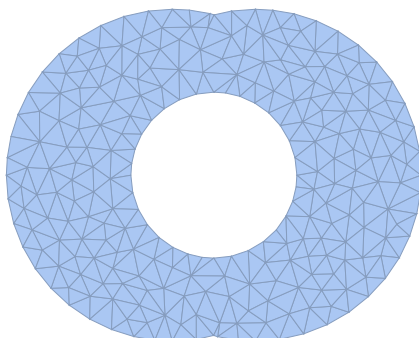
Funcionalidades: Operando con “Geometric Regions”

Una de las últimas funcionalidades es el tratamiento geométrico. Aquí obtenemos la región formada por la unión de dos círculos

```
 $\mathcal{R} = \text{RegionUnion}[\text{Disk}[\{0, 0\}, 2], \text{Disk}[\{1, 0\}, 2]]; \\ \text{DiscretizeRegion}[\mathcal{R}]$ 
```

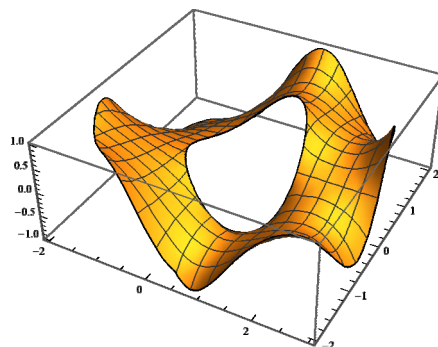


```
 $\mathcal{R1} = \text{RegionDifference}[\mathcal{R}, \text{Disk}[\{1/2, 0\}, 1]]; \\ \text{DiscretizeRegion}[\mathcal{R1}]$ 
```



En este ejemplo vamos a resolver una ecuación diferencial limitada a una región específica definida geométricamente.

```
 $\nabla_{\{x,y\}}^2 u[x, y] \\ u^{(0,2)}[x, y] + u^{(2,0)}[x, y] \\ \text{sol} = \text{NDSolveValue}[ \\ \{ \nabla_{\{x,y\}}^2 u[x, y] == 0, \text{DirichletCondition}[u[x, y] == \text{Sin}[x y], \text{True}] \}, u, \{x, y\} \in \mathcal{R1}]; \\ \text{Plot3D}[\text{sol}[x, y], \{x, y\} \in \mathcal{R1}] // \text{Quiet}$ 
```



Probabilidad y Estadística

Ademas de las aplicaciones basicas comunes eh muchos programas estadisticos Mathematica incluye entre otras :

- Funciones de probabilidad (guide/ParametricStatisticalDistributions).- De acuerdo con la información proporcionada en la página del fabricante el número de funciones es sustancialmente superior a las que contienen otros conocidos programas de cálculo estadístico como R, SAS, SPSS, etc.
- Distribuciones derivadas (guide/DerivedDistributions).- Permite construir funciones de probabilidad que son combinación de otras distribuciones.
- Distribuciones no paramétricas (guide/NonparametricStatisticalDistributions).- Permite obtener funciones de distribución no paramétricas a partir de datos empíricos.
- Cálculo automatizado de probabilidades (Probability) y esperanza matemática (Expectation).
- Se amplían las propiedades asociadas a las distribuciones: PDF (distribución de densidad), CDF (función de distribución), SurvivalFunction (función de supervivencia), Moment (momentos), EstimatedDistribution, ...
- Se aumentan las posibilidades de generación de números aleatorios (RandomVariate), de amplio uso en simulación Montecarlo y en numerosos análisis estadísticos, de acuerdo a la función de distribución que se especifique.
- Ajustes de datos para varias variables: regresión no lineal (NonlinearModelFit), modelo lineal generalizado (GeneralizedLinearModelFit) y modelo de regresión logístico (ProbitModelFit).
- Gráficos estadísticos específicos: DiscretePlot3D, Histogram, PairedHistogram, QuantilePlot, ProbabilityPlot, BoxWhiskerChart, DistributionChart, ...
- Series temporales: guide/TimeSeriesProcesses.
- Análisis de *clusters* mediante el comando FindClusters y otros que se detallan en: guide/DistanceAndSimilarityMeasures.
- Análisis de supervivencia y fiabilidad.
- Procesos estocásticos (guide/RandomProcesses), incluida ecuaciones diferenciales estocásticas.

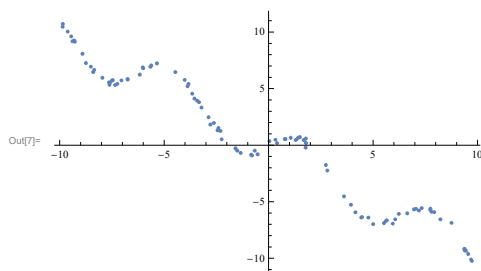
FindFormula

Generate data with normally distributed noise:

```
In[5]:= SeedRandom[4311]
In[6]:= data = Table[{x, -x + 2 Sin[x] + RandomVariate[NormalDistribution[0, 0.2]]},
  {x, RandomReal[{-10, 10}, 100]}];
```

Visualize the data:

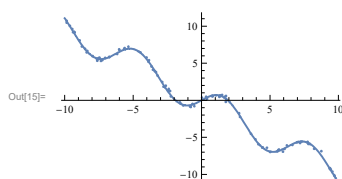
```
In[7]:= ListPlot[data]
```



```
In[14]:= fit = FindFormula[data, x, 1]
```

```
Out[14]= 2.0155 sin(x) - 0.996833 x
```

```
In[15]:= Show[ListPlot[data], Plot[fit, {x, -10, 10}, PlotRange -> All]]
```



FindDistribution

Generate data sampled from an exponential distribution:

```
In[16]:= D = ExponentialDistribution[1];
```

```
data = RandomVariate[D, 1000];
```

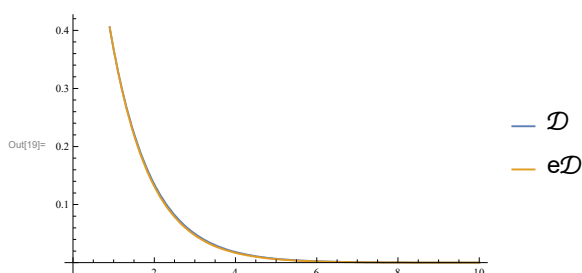
Find the best distribution from the data:

```
In[18]:= estimatedD = FindDistribution[data]
```

```
Out[18]= ExponentialDistribution[1.02784]
```

Compare the PDFs for the original and estimated distributions:

```
In[19]:= Plot[{PDF[D, x], PDF[estimatedD, x]}, {x, 0, 10}, PlotLegends -> {"D", "eD"}]
```



Machine Learning

Simple

```
In[2]:= digit = Classify[
  {2→2, 5→5, 4→8, 0→0, 2→2, 7→7, 5→5, 1→1, 3→3, 0→0, 3→3, 9→9, 6→6,
    2→2, 8→8, 2→2, 0→0, 6→6, 6→6, 1→1, 1→1, 7→7, 8→8, 5→5, 0→0,
    4→4, 7→7, 6→6, 0→0, 2→2, 5→5, 3→3, 1→1, 5→5, 6→6, 7→7, 5→5,
    4→4, 1→1, 9→9, 3→3, 6→6, 8→8, 0→0, 9→9, 3→3, 0→0, 3→3, 7→7, 4→4,
    4→4, 3→3, 8→8, 0→0, 4→4, 1→1, 3→3, 7→7, 6→6, 4→4, 7→7, 2→2,
    7→7, 2→2, 5→5, 2→2, 0→0, 9→9, 8→8, 9→9, 8→8, 1→1, 6→6, 4→4, 8→8,
    5→5, 8→8, 0→0, 6→6, 7→7, 4→4, 5→5, 8→8, 4→4, 3→3, 1→1, 5→5,
    1→1, 9→9, 9→9, 9→9, 2→2, 4→4, 7→7, 3→3, 1→1, 9→9, 2→2, 9→9, 6→6}]
```

```
Out[2]= ClassifierFunction[
   Input type: Image
  Number of classes: 10
]
```

```
In[3]:= digit[{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}]
```

```
Out[3]= {0, 1, 2, 3, 4, 1, 6, 7, 8, 9}
```

ImageIdentify

```
In[4]:= ImageIdentify[
  
]
```

```
Out[4]= English cocker spaniel
```

Paralelizacion.

Pretendemos factorizar los siguientes enteros (46 digitos):

```
In[1]:= largeIntegers = {
  3 614 639 327 625 022 357 498 716 997 616 141 559 555 385 788 ,
  1 555 448 460 931 142 740 464 025 812 509 966 226 753 668 067 ,
  2 181 897 995 524 176 530 622 591 658 224 757 998 839 632 491 ,
  1 922 853 025 887 991 704 598 930 446 931 704 051 497 624 646 ,
  2 272 856 689 805 810 352 674 075 033 225 118 441 591 285 223 ,
  2 362 930 657 136 771 575 226 664 337 831 303 048 141 997 267 ,
  2 284 188 855 407 316 248 949 247 887 514 956 603 668 459 951 ,
  1 955 774 192 657 260 628 352 808 399 207 168 800 531 153 117
};
```

Utilizando 1 kernel

```
In[2]:= AbsoluteTiming[
  Map[FactorInteger, largeIntegers]
]
```

Utilizando los kernels disponibles combinado con Parallelize

```
In[3]:= LaunchKernels[]
Out[3]:= {KernelObject(1, local), KernelObject(2, local)}
In[4]:= AbsoluteTiming[
  Parallelize[Map[FactorInteger, largeIntegers]]
]
```

Cloud application

La función `CloudDeploy` permite generar aplicaciones para que se ejecuten en la nube. Inserte el enlace en un sitio web desde el que podrá ejecutarse.

```
CloudDeploy[Manipulate[ContourPlot3D[x^2 + y^2 + a z^3 == 1,  
  {x, -2, 2}, {y, -2, 2}, {z, -2, 2}, Mesh → None], {a, -2, 2}]]  
CloudObject[https://www.wolframcloud.com/objects/d13d97f6-33da-47b7-9b1f-52304ea9c62b]
```

webMathematica

<http://www.wolfram.com/products/webmathematica/index.html>

<http://oed.usal.es/webMathematica/>

WOLFRAM ENTERPRISE PRIVATE CLOUD

<https://www.wolfram.com/enterprise-private-cloud/>

Materiales adicionales:

Lo que hay que saber: <http://www.wolfram.com/language/elementary-introduction/>

<https://www.crcpress.com/Mathematica-Beyond-Mathematics-The-Wolfram-Language-in-the-Real-World/Len/9781498796293>

<http://www.wolfram.com/mathematica/quick-revision-history.html>

<http://www.wolfram.com/training/special-event/?source=nav>

Inicio