

Obtención de raíces con Excel y Matlab

Cortés Rosas Jesús Javier, González Cárdenas Miguel Eduardo
Pinilla Morán Víctor Damián, Salazar Moreno Alfonso
Tovar Pérez Víctor Hugo *

2019

Resumen

Esta publicación pertenece al proyecto *Plataforma educativa para Análisis Numérico*, realizado con el apoyo del Programa UNAM-DGAPA-PAPIME PE105717.

Este apartado presenta dos ejemplos de recursos de cómputo que permiten obtener raíces de ecuaciones algebraicas y trascendentes de manera muy eficiente. Ambos recursos, y todos los de otros lenguajes y herramientas de computación, están basados en los métodos numéricos ya revisados, tanto abiertos como cerrados, y son el ejemplo de cómo deben aplicarse en la solución de este tipo de problemas, ahora que se conoce su funcionamiento, sus condiciones de convergencia y detalles particulares¹.

1. Obtención de raíces con Excel

Microsoft Excel es una hoja electrónica de cálculo que, debido a la enorme penetración comercial del sistema operativo Windows, ha desplazado a todos sus competidores convirtiéndose en uno de los paquetes más populares.

Si bien su origen se encuentra en las hojas tabulares utilizadas en la contabilidad, la flexibilidad que proporciona la programación con direcciones entre celdas permite utilizar la hoja electrónica de cálculo para la gestión de bases de datos, de gráficas y de algoritmos matemáticos.

No es objetivo del presente trabajo proporcionar al lector los conocimientos necesarios para utilizar una hoja electrónica de cálculo. De hecho, se considera que quien pretenda aplicarla en el uso de métodos numéricos debe conocer con precisión los conceptos básicos, por lo que se recomienda profundizar particularmente en tres conceptos:

1. Direccionamiento relativo
2. Direccionamiento absoluto
3. Funciones

*Profesores de la División de Ciencias Básicas de la Facultad de Ingeniería de la UNAM.

La programación propuesta se hace con el uso de estos tres recursos ¹ y, como podrá constatar, el éxito en el uso de la hoja de cálculo para resolver métodos numéricos radica en la habilidad de acomodar los datos en la matriz de celdas para obtener la mayor ventaja de los direccionamientos relativo y absoluto.

Para dar cuenta de lo anterior, se recomienda explorar las soluciones propuestas con ayuda de las herramientas de *Auditoría de fórmulas* con que cuenta Excel.

1.1. Solución de ecuaciones algebraicas y trascendentes a través de librerías incluidas en Microsoft Excel

Son dos las librerías por medio de las cuales se pueden resolver ecuaciones algebraicas y trascendentes.

Los recursos que se muestran a continuación están contenidas en todas las versiones de Excel y los ejemplos mostrados se obtienen de la versión Office 365.

1.2. Buscar objetivo (Goal Seek)

La librería *Buscar objetivo* o *goal seek* se encuentra ubicada en el menú *Datos*, submenú *Análisis de hipótesis* opción *Buscar objetivo*. Al seleccionarlo se despliega la siguiente ventana:

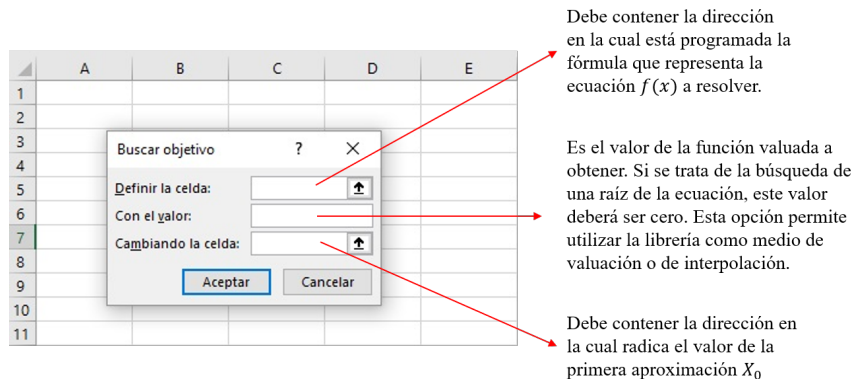


Figura 1: Ventana *Buscar objetivo*

Para poder utilizar esta librería es necesario que en celdas determinadas se programe la fórmula que represente la ecuación a resolver direccionada a una primera aproximación, de acuerdo a la siguiente descripción:

- Definir celda: Debe contener la dirección en la cual está programada la fórmula que representa la ecuación $f(x)$ a resolver.
- Con el valor: Implica el valor de la función valuada a obtener. Si se trata de la búsqueda de una raíz de la ecuación, este valor deberá ser cero. Esta opción permite utilizar la librería como medio de valuación o de interpolación.

¹No se encuentra en el alcance el uso de macros o recursos similares

- Para cambiar la celda: Debe contener la dirección en la cual radica el valor de la primera aproximación x_0 .

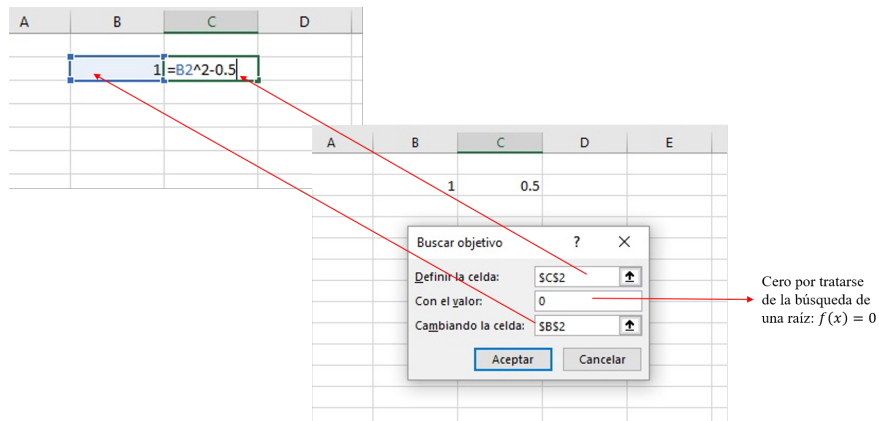


Figura 2: Captura de la información

Si la ecuación tiene solución, de acuerdo al valor inicial buscado (es decir, converge para dicha aproximación), en las celdas en las cuales se direccionaron la función y el valor inicial, quedarán el valor de la aproximación $f(x)$ y la raíz aproximada, respectivamente.

En la figura 2 se muestra la información previa, en la celda C1 se programa la fórmula $f(x) = x^2 + 0,5$, mientras que en la celda B1 se incluye la primera aproximación a la raíz, en este caso $x_0 = 1$. Se observa que C1 está direccionada a B1.

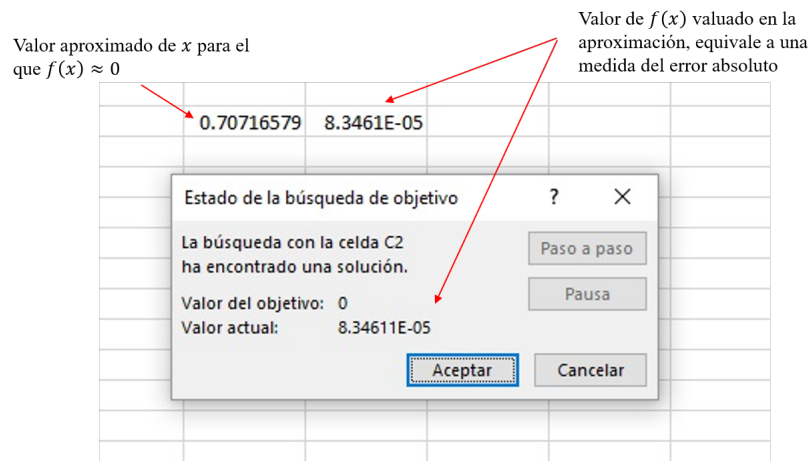


Figura 3: Resultados

Por último, si existe convergencia de la solución en la aproximación propuesta, el resultado se mostrará de manera similar a la figura 3.

1.3. Solver

El complemento *Solver*² proporciona una mejor aproximación que *Buscar objetivo*, además de que es más flexible ya que permite obtener máximos y mínimos, así como la valuación o interpolación de funciones. Opera en forma idéntica que *Buscar objetivo*.

Solver no se instala de inicio en Excel. Deberá instalarse en el menú *Archivo*, después en el menú *Complementos* y en el menú desplegable *Administrar*. Después de activarlo, aparecerá un submenú adicional *Solver* en el menú principal *Datos*.

La información previamente incluida en la hoja es (figura 4):

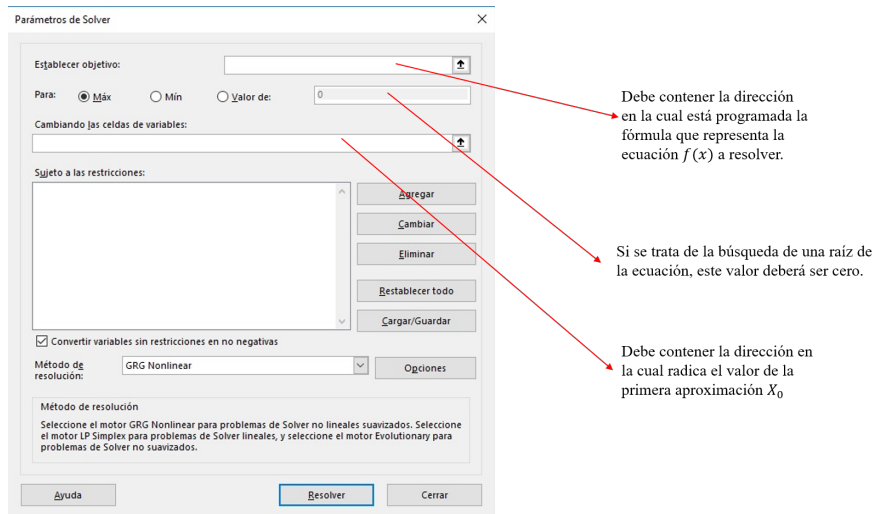


Figura 4: Información en la ventana *Solver*

- Celda objetivo: Debe contener la dirección en la cual está programada la fórmula que representa la ecuación $f(x)$ a resolver.
- Valores de: Implica el valor de la función valuada a obtener. Si se trata de la búsqueda de una raíz de la ecuación, este valor deberá ser cero. Esta opción permite utilizar la librería como medio de valuación o de interpolación.
- Cambiando las celdas: Debe contener la dirección en la cual radica el valor de la primera aproximación x_0 .

El procedimiento es idéntico al utilizado para *Buscar objetivo*. En la figura 5 se muestra la ubicación de la información en la ventana *Solver*.

El complemento Solver informa con una ventana auxiliar cuando obtiene o no una solución (6)

El complemento concluye haciendo click en el botón Aceptar, con lo que se cierra la ventana quedando los resultados en las celdas inicialmente seleccionadas (7).

Solver ofrece una alternativa más completa que *Buscar objetivo*. Pueden compararse los resultados correspondientes en las figuras 3 y 7: *Solver* arroja un menor error absoluto. Adicionalmente, pue-

²El nombre de este no cambia entre versiones en Español o en Inglés

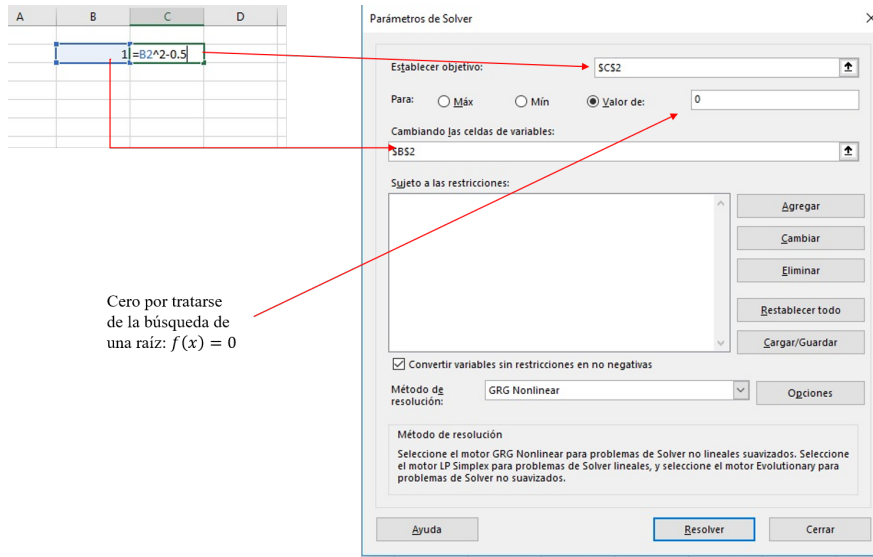


Figura 5: Llenado de la ventana *Solver*

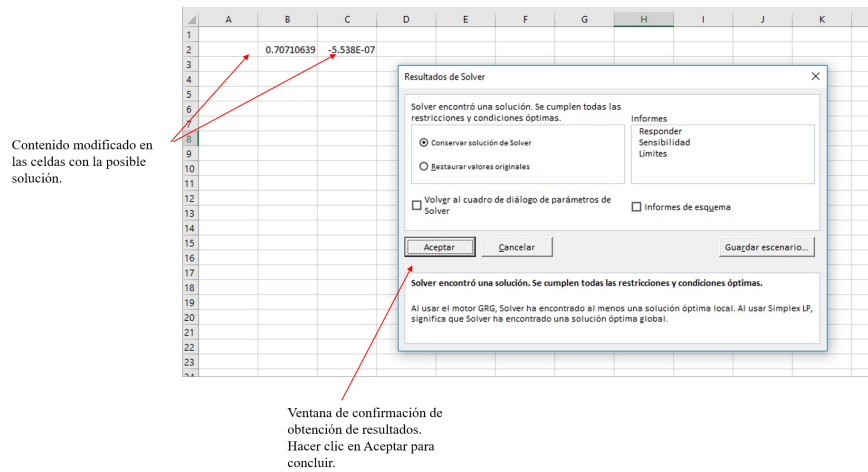


Figura 6: Ventana de confirmación de obtención de resultados

A	B	C	D
	0.70710639	-5.538E-07	

Figura 7: Resultado final

den utilizarse las opciones adicionales mostradas en su ventana para obtener máximos, mínimos e interpolaciones de la función.

No obstante, es común encontrar funciones que no converjan en el complemento *Solver*; ante esta situación, *Buscar objetivo* es la alternativa.

2. Solución de ecuaciones con MATLAB

MATrixLABoratory es un superlenguaje de programación cuyos fines son realizar análisis numérico. Su diseño original se concibió para funcionar en las grandes computadoras (mainframes) con sistema operativo UNIX. Si bien, actualmente hay versiones disponibles para Windows, éstas aún conservan el espíritu y muchas de las características de la versión UNIX, particularmente su estructura de archivos y su ortodoxo estilo de programación.

Estas versiones de MATLAB sobre Windows ofrecen prestaciones muy aceptables en cuanto a aspectos de exactitud y precisión, siendo su principal desventaja el uso excesivo de recursos tales como la memoria RAM y su despliegue gráfico.

No obstante, las versiones sobre UNIX son aún muy socorridas y resultan indispensables en las supercomputadoras de uso actual, ya que las arquitecturas de estos equipos permiten la vectorización de operaciones, lo cual es plenamente coincidente con la filosofía matricial de MATLAB; esta cohesión permite la explotación óptima de los recursos de cómputo realizando una cantidad asombrosa de operaciones matemáticas en intervalos increíblemente cortos de tiempo.

El presente trabajo parte del conocimiento básico en el manejo de MATLAB por parte del lector. Este conocimiento básico comprende el uso de la ventana de comandos, de la sintaxis mínima de los comandos, funciones y de la ayuda de MATLAB.

2.1. Operaciones básicas de polinomios con MATLAB

1. Introducción de polinomios. MATLAB reconoce a los polinomios como arreglos matriciales de una dimensión, es decir, como vectores. Sea el polinomio $P(x) = x^4 - x^3 + 6x^2 - 3x + 4$. La captura de información será:

```
>> p=[1 -1 6 -3 4]
```

El vector se forma con los coeficientes del polinomio, incluyendo a los de valor cero.

2. Evaluación de polinomios. La evaluación se hace a través del comando `polyval(p,x)` donde p es el vector de coeficientes que representa al polinomio y x es el valor en que se desea evaluar al polinomio.

```
>>c=polyval(p,5)
>>c=639
```

3. Derivación de polinomios. El comando `polyder(p)` devuelve el vector con los coeficientes de la primera derivada del polinomio p ³.

³En la entrega de resultados MATLAB omite los corchetes

```
>>d=polyder(p)
>>d= 4    -3    12    -3
```

4. Construcción de un polinomio a partir de sus raíces. El comando `poly(a)` devuelve el vector de coeficientes del polinomio formado por las raíces incluidas en la matriz a . Su captura se hace separando los renglones con un punto y coma.

```
>>a=[0.2349+2.2105i; 0.2349-2.2105i; 0.2651+0.8598i; 0.2651-0.8598i]
>>e=poly(a)
>>e=1.0000    -1.0000    6.0001    -3.0003    4.0003
```

El caracter i es de uso reservado para el número imaginario $i = \sqrt{-1}$.

5. Obtención de las raíces de un polinomio. El comando `roots(p)` devuelve las raíces del polinomio p .

```
>>f=roots(p)
>>f=0.2349+2.2105i; 0.2349-2.2105i; 0.2651+0.8598i; 0.2651-0.8598i
```

2.2. Graficación

MATLAB grafica una o varias funciones tanto en dos como en tres dimensiones. Operativamente, la gráfica se obtiene en una ventana específica que cuenta con herramientas que permiten la inclusión de textos, leyendas y otros objetos. Las gráficas pueden exportarse en varios formatos gráficos.

En esta sección se atenderá la grafica de polinomios (y en general de gráficas) en dos dimensiones. El comando encargado de ello es `plot(x,y)`, en el cual x es la variable independiente y y es la variable dependiente a graficar. El comando posee varios argumentos más que sirven para modificar el color y el caracter con que se forma la gráfica.

Se propone graficar el polinomio $P(x) = x^4 - x^3 + 6x^2 - 3x + 4$ en el rango $[-3, 5]$. El primer paso consiste en definir el vector que contendrá los valores de x en el rango determinado, lo cual se logra con la estructura de contador que tiene la siguiente sintaxis:

```
>>x=Vinicial:incremento:Vfinal
```

Para el ejemplo que nos ocupa la instrucción será:

```
>>x=-3:0.01:5;
```

Ahora bien, resta definir el vector que contenga las imágenes de la variable dependiente, es decir, del polinomio valuado en el rango definido:

```
>>Y=x.^4-x.^3+6.*x.^2-3.*x+4;
```

Sobre estas dos líneas de captura vale la pena hacer dos comentarios: la inclusión de un punto y coma (;) al final de cada instrucción inhibe la presentación de resultados en la ventana de comando, o `echo` como lo denomina MATLAB. Por otra parte, en la captura del polinomio se incluye un punto (.) antes de operaciones de multiplicación y potencia. La razón consiste en el carácter matricial de las operaciones que por defecto realiza MATLAB. La operación punto (.) le indica que se trata de operaciones escalares y no vectoriales.

La instrucción para graficar es:

```
>>plot(x,y)
```

cuyo resultado se muestra en la figura 8:

Si se desean ilustrar dos funciones en la misma gráfica, por ejemplo $Z = 10 \cdot e^x$ debe, inicialmente, calcular el vector z :

```
>>z=10*exp(x);
```

Después invocar el comando con esta forma:

```
>>plot(x,y,'r',x,z,'b')
```

Las funciones deben graficarse en pares ordenados $[x, y]$ aunque utilicen la misma variable independiente; por otra parte, los modificadores entre apóstrofes indican el color de cada gráfica. Para conocer los distintos modificadores disponibles puede consultarse la ayuda específica del comando `>>plot(x,y)`. El resultado se muestra en la figura 9.

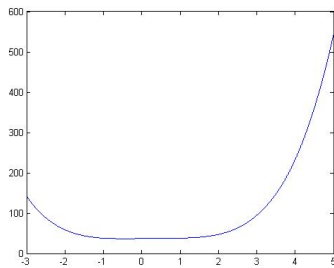


Figura 8: Graficación de una función

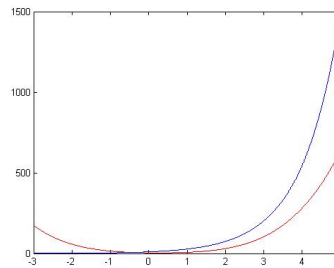


Figura 9: Graficación múltiple

2.3. Conclusiones

Excel es la hoja de cálculo más conocida y comercializada lo que la hace disponible en varias plataformas, desde equipos de cómputo de escritorio y portátiles así como tabletas y teléfonos inteligentes. Posé una capacidad aceptable para graficar y es ampliamente utilizada. El análisis numérico no es su especialidad, pero sí es una opción para ser usado con este fin.

MATLAB requiere experiencia previa en programación y dado que, para su uso es necesario un licenciamiento, su uso no es muy generalizado. Afortunadamente, un gran número de instituciones de educación posee licencias para uso y esto le permite a sus estudiantes y profesores utilizarlo. MATLAB fue creado para hacer análisis numérico, junto con FORTRAN, por lo que su dominio garantiza la solución de cualquier problema matemático (incluso con matemática analítica).

Otra de sus grandes ventajas es la enorme comunidad que en todo el mundo utiliza MATLAB y comparte su trabajo publicándolo en internet, lo que hace prácticamente innecesario adquirir textos o manuales.

Los comandos antes señalados son una pequeñísima parte de la biblioteca de funciones disponibles en MATLAB. Se recomienda, para tener un panorama general de su cobertura, invocar el comando `>>help`, tanto en forma general como en forma particular, seguido de la librería o el comando deseado.

Notas

¹Las figuras y gráficas incluidas en este trabajo fueron elaboradas por los autores

Referencias

García, J., Rodríguez, I., y Vidal, J. (2005). *Aprenda matlab 7.0 como si estuviera en primero* (U. P. de Madrid, Ed.).

García B., S. (2017). *Métodos numéricos*.

Mathews, J., y Fink, K. (1999). *Métodos numéricos con matlab* (P. Hall, Ed.).

Moore, H. (2007). *Matlab para ingenieros* (P. P. Hall, Ed.).

Sandoval, H. (2017). *Métodos numéricos*.