

REPRESENTACIÓN AUTOMÁTICA DE FUNCIONES EN EXCEL Y SU APLICACIÓN DOCENTE

BERNAL GARCÍA, JUAN JESUS

juanjesus.bernal@upct.es

*Universidad Politécnica de Cartagena – Dpto. de Métodos Cuantitativos e Informáticos
C/Real, 3. 30203-Cartagena*

MARTÍNEZ MARÍA-DOLORES, SOLEDAD MARÍA

soledad.martinez@upct.es

*Universidad Politécnica de Cartagena – Dpto. de Métodos Cuantitativos e Informáticos
C/Real, 3. 30203-Cartagena*

BERNAL SOTO, PABLO

pablo.bernal@gmail.com

*Universidad Politécnica de Cartagena – Grupo de investigación de Métodos
Cuantitativos e Informáticos para la economía y la empresa
C/Real, 3. 30203-Cartagena*

Recibido 26/10/2011

Revisado 27/11/2011

Aceptado 13/12/2011

RESUMEN: Excel cuenta de forma estándar con numerosas funciones matemáticas predefinidas, de manera que a través de la creación de tablas de valores XY es posible representarlas. Pero lo que no se usa con tanta frecuencia es el hecho de que también podemos evaluar funciones con solo escribirlas como texto, y representarlas de forma directa, sin necesidad de elaborar tablas utilizando para ello los nombres dinámicos. Si además lo combinamos con la utilización de otras herramientas de dicha hoja de cálculo y la correspondiente programación en VBA, podemos automatizar la presentación de gráficas, representando incluso dos a la vez o sus derivadas numéricas, e incluso desde otras aplicaciones de Office, tales como Word o PowerPoint. En conclusión, por todo ello creemos que la técnica presentada resulta muy interesante para ser utilizada en docencia, tanto por parte del profesorado, como por los estudiantes.

Palabras clave: Excel, funciones matemáticas, representación de funciones, aplicaciones docentes.

ABSTRACT: Excel incorporates a huge number of predefined math functions that are typically displayed by means of x-y tables, but very seldom the functions in text format are evaluated and directly graphically represented, without x-y tables, by using dynamic Excel named ranges. If this functionality is combined with other Excel tools and with the VBA programming, this task can be automatized and improved with the representation of more than one function in the same graph or the presentation of the numerical derivative of the function or even the importation from other Office apps, as Word or PowerPoint. We think this technique is a very useful teaching resource that can be used by both teachers and students.

Keywords: Excel, math functions, teaching applications, functions representing.

1. Introducción

La representación de funciones matemáticas en Excel puede resultar muy útil para enseñar a los alumnos la forma de determinadas funciones conocidas y muy utilizadas en las asignaturas de Métodos Cuantitativos y en las de Economía. Para hacerlo con una hoja de cálculo como Excel, lo usual es crear una tabla XY e insertar un gráfico de tipo *dispersión*¹. La elaboración y manejo de dichas gráficas puede ser mejorada recurriendo a otras herramientas incluidas en la propia hoja, como son los *controles de desplazamiento* dotándolas así de interactividad, como muestra la figura 1, [1] [2] [3].

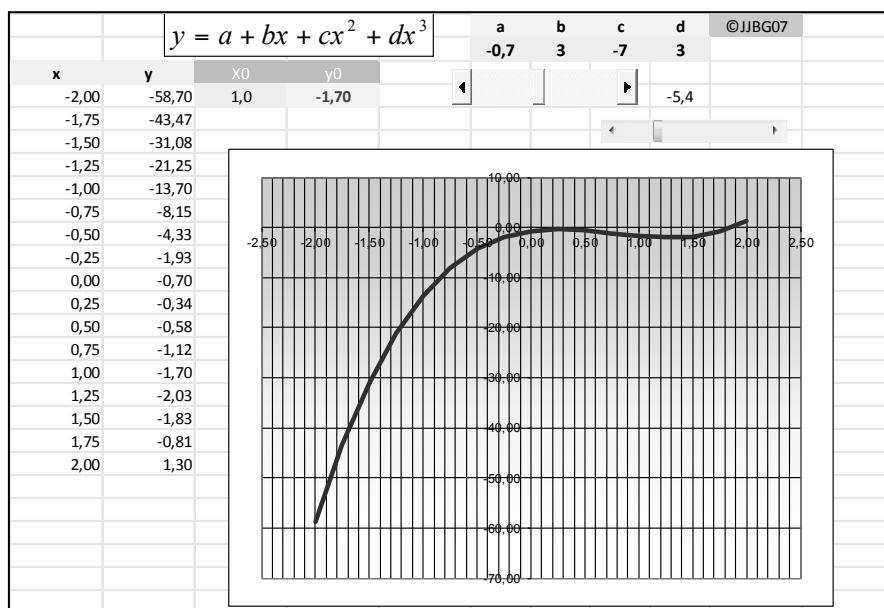


Figura 1. Gráfica XY con tabla de datos y controles de formulario

Evidentemente existen diversos programas para la representación de funciones, tanto de tipo matemático general como específico. Los hay de software libre, que pueden ser encontrados con cierta facilidad en Internet, y otros muy completos para geometría como por ejemplo GeoGebra², o FooPlot³ que nos permite dibujar varias curvas a la vez y almacenarlas en formato PDF. No obstante, las ventajas de hacerlo con Excel derivan de la flexibilidad de esta utilidad, y de que son de utilización habitual por parte de los alumnos. Además, como veremos más adelante, nos permite la conexión con otros programas de su propio paquete ofimático cuyo uso también es muy frecuente, como son Word y PowerPoint. Esto posibilita la integración dichas herramientas, de una forma local, sin necesidad de trabajar con conexión a la Red. Es cierto que comercialmente existen complementos para dotar a Excel de mayor potencia a la hora de representar funciones, como la aplicación XLFit⁴, pero dichas soluciones, lógicamente, no son gratuitas.

Conseguir una representación de calidad en Excel, puede ser tediosa y requerir un conocimiento avanzado. Por ello, presentamos aquí una metodología elaborada que indica de forma sencilla y rápida, cuál es el *aspecto* final de una función, lo que permite comprender mejor la amplitud de su significado, al tiempo que sirve a los alumnos de guía a la hora de resolver cualquier ejercicio de matemáticas que contenga una representación de funciones. Al tratarse de una propuesta innovadora, se han ido haciendo pruebas sucesivas que han mejorado progresivamente las representaciones y las han ido dotando de nuevas prestaciones.

¹ http://tushar-mehta.com/excel/software/plot_manager/index.html (fecha consulta: 09/12/11)

² <http://www.geogebra.org/> (fecha consulta: 09/12/11)

³ <http://fooplots.com/> (fecha consulta: 09/12/11)

⁴ <http://www.excelcurvefitting.com/?gclid=CJazgvqD-KwCFcMMtAodshcGSQ> (fecha consulta: 09/12/11)

Veamos en qué consiste el procedimiento, aclarando que el propósito inicial es que en una hoja Excel con la sola escritura de la expresión de la función, dibuje de forma automática la gráfica de la misma, pero según se expondrá más adelante, la técnica se ha ido depurando pudiéndose cambiar la vista de la misma mediante los límites de los ejes, e introducir parámetros de variación de la función inicial, e incluso realizar todo el proceso desde otras aplicaciones ofimáticas sin necesidad de manejar todas las posibilidades de Excel; todo ello, gracias a la programación en VBA (*Visual Basic for Applications*) [4], común a las aplicaciones del paquete Office de *Microsoft*. A modo de ejemplo, la figura 2, muestra la representación de la “Campana de Gauss”, cuya fórmula se ha escribió en una celda de la propia hoja.

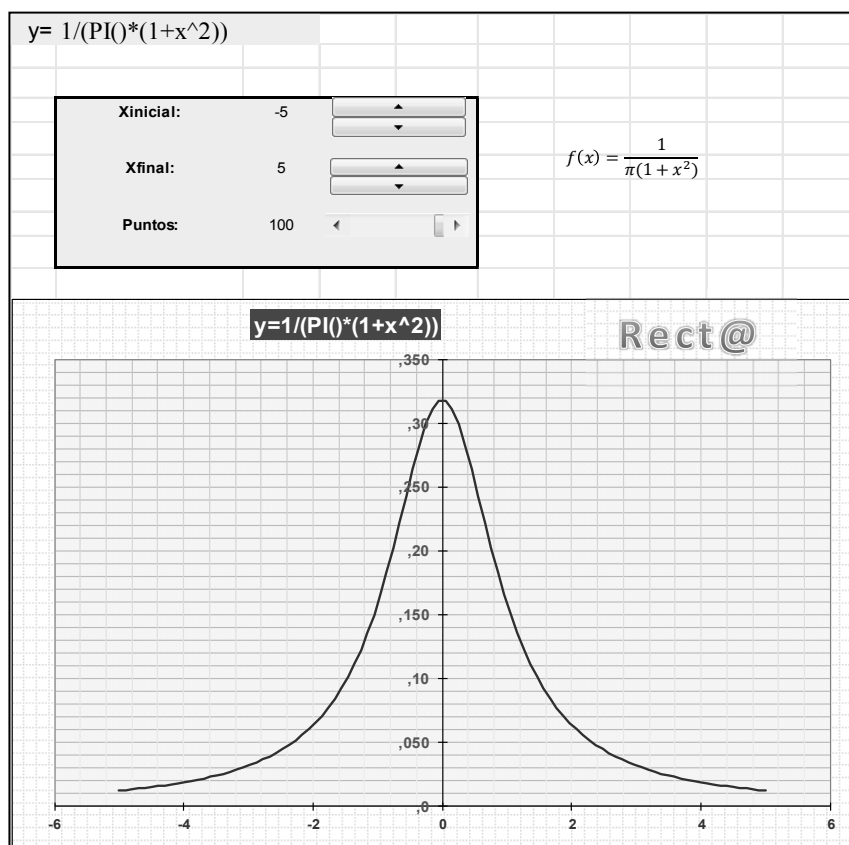


Figura 2. Representación de la campana de Gauss

2. Evaluación automática de funciones de una variable

Entre las funciones menos conocidas de Excel se encuentra la de EVALUAR, debido a que no aparece al invocar al asistente de fórmulas y funciones del programa, $f(x)$, ni en ninguna de las categorías existentes de funciones que éste nos relaciona. Si definimos con un nombre la celda que contiene la expresión de una función en modo texto mediante el administrador de nombres del menú fórmulas, como por ejemplo a la celda que contiene la función la llamamos con el nombre “resul” (figura 3), podremos colocar el cursor en cualquier celda situada en la misma línea de dicha fórmula, escribir =resul, y la función EVALUAR nos proporciona el valor de dicha expresión para x . En la figura 4 se observa que las siete expresiones del ejemplo incluyen operadores y/o funciones matemáticas válidas en Excel, así como que x puede ser un valor o una función; así en el ejemplo propuesto es el valor de $\text{PI}()$ [5].

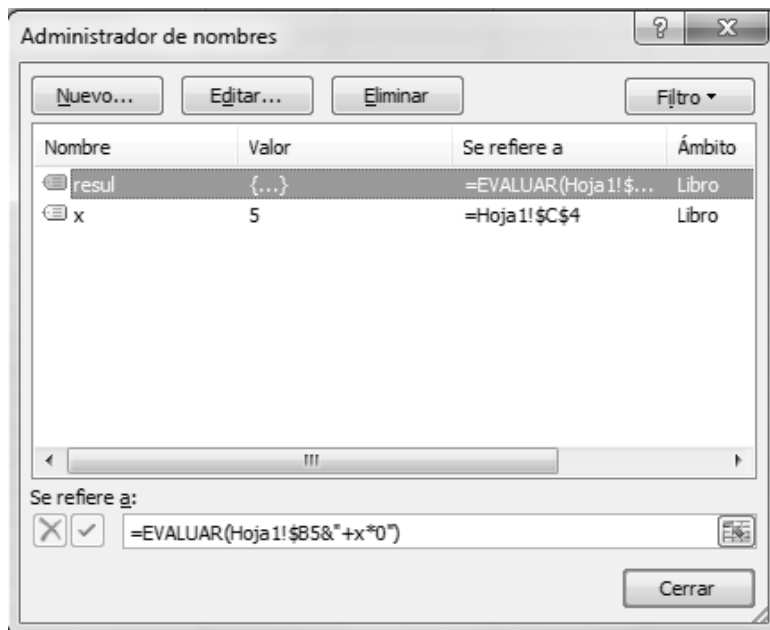


Figura 3. Nombre x y “resul”

Evaluar las funciones:		X	
		3,141593	valor
1	$\cosh(x^2+1/3)+\tan(x)$		15366,824
2	x^3+4*x^2-5		220,000
3	seno(x)		-1,0
4	$\cos(x^2)*\text{raiz}(x*2500)$		110,820
5	cosh(x)		74,210
6	$360-20*x$		260,000
7	$x^3-8*x^2+17*x-10$		0,000

Figura 4. Ejemplo de evaluación funciones (x=0,5)

3. Representación automática de funciones de una variable

Una aplicación más interesante para EVALUAR, consiste en utilizarla para representar funciones de una variable. A continuación presentamos distintas opciones con diferentes ejemplos.

3.1 Evaluar y representar una función

El primer paso, como ya hemos apuntado, es el de representar la gráfica de la expresión escrita como texto en una celda, tal y como aparece en el ejemplo mostrado en la figura 5 para $f(x)=e^x(x-1)$, o en su forma textual: $\text{exp}(x)*(x-1)$.

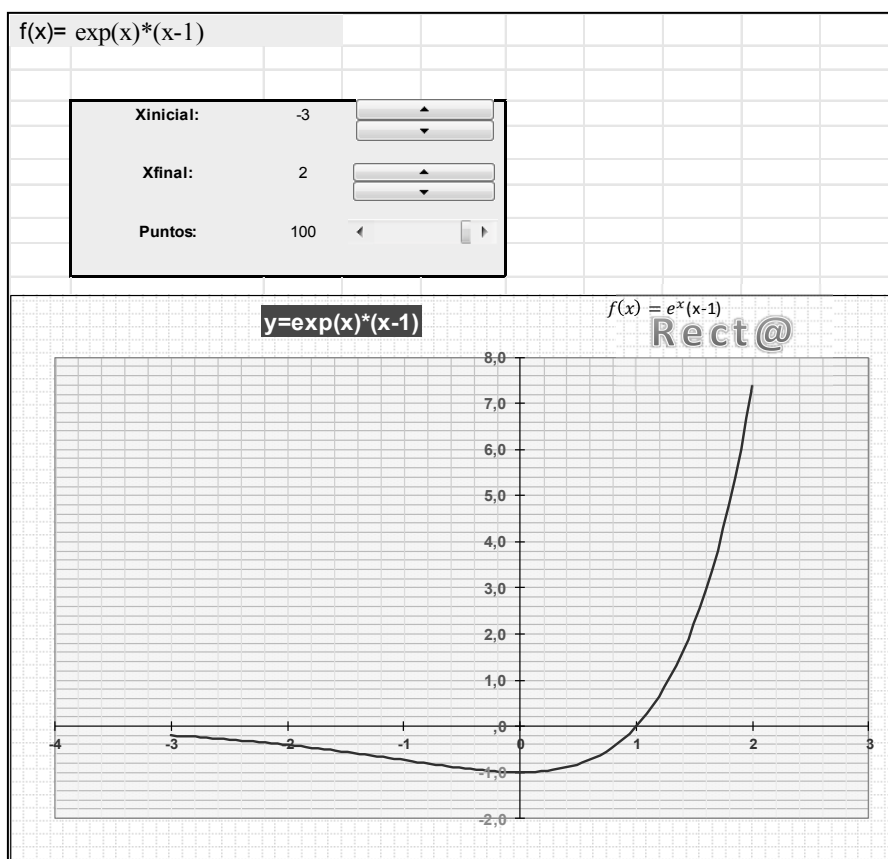


Figura 5. Gráfica de la función $e^x(x - 1)$

Para ello, debemos nombrar unas celdas, comenzando por la que contiene la función a representar (*expresión*), x e y , al mismo tiempo que si lo que queremos hacer es incluir una variable al inicio y al final del eje x , y que ésta se visualice, habrá que incluir los nombre de *xinicial* y *xfinal*, lo cual es muy útil a la hora de poder modificar manualmente la vista, a modo de zoom (figura 6).

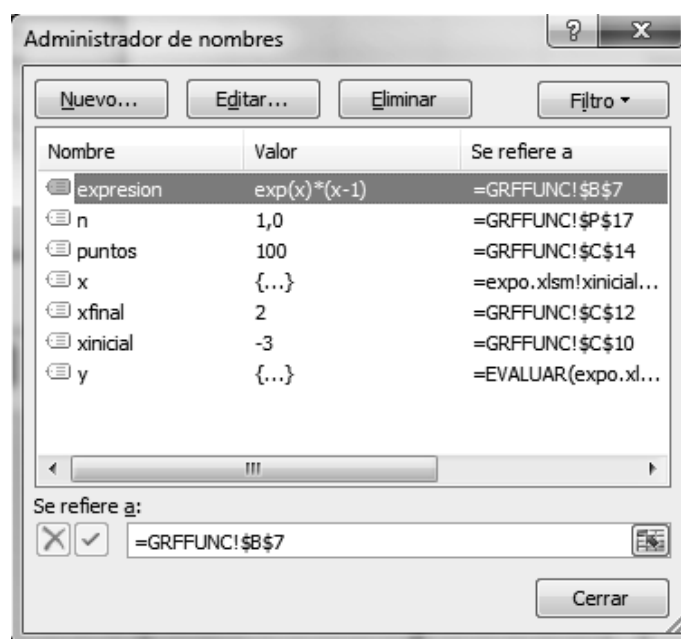


Figura 6. Celdas nombradas para representar funciones

Para que realmente actúe, en el nombre de los ejes debe figurar la ya citada función *EVALUAR*, y otra función que nos permite trabajar con rangos dinámicos. Nos referimos a la función *DESREF* [3] [6]:

$$X:=\text{expo.xlsx!xinicial}+(\text{FILA}(\text{DESREF}(\text{GRFFUNC!}\$N\$1;0;0;\text{expo.xlsx!puntos};1))-1)*(\text{expo.xlsx!xfinal}-\text{expo.xlsx!xinicial})/(\text{expo.xlsx!puntos}-1)$$

$$Y:=\text{EVALUAR}(\text{expo.xlsx!expresion}\&"+x*0")$$

Con la finalidad de ampliar la capacidad de representación, incluimos la posibilidad de añadir un parámetro (*n*), que factorize a *x*. Utilizando dicho parámetro por ejemplo, podemos mostrar a los alumnos de una forma sencilla y clara, cómo al aumentar la frecuencia en la función $\text{seno}(x)$, en esta ocasión $n=4$ en lugar de 1, su periodo 2π , se ve dividido por la misma cantidad $2\pi/4$ (figura 7).

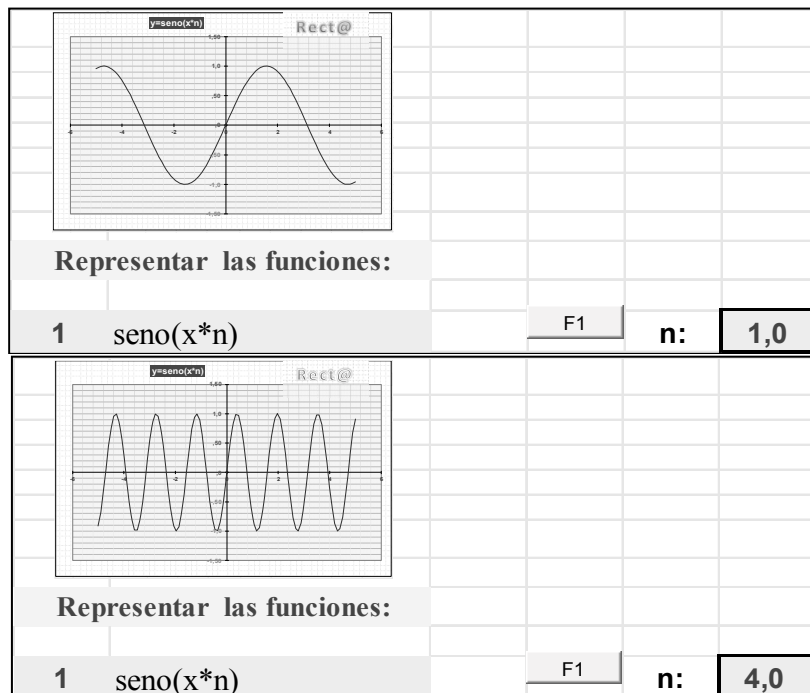


Figura 7. Gráfica de $\text{seno}(x)$ y $\text{seno}(4x)$

3.2. Otras herramientas de Excel

Recurriendo a diversas opciones que incluye la propia hoja de cálculo, es posible mejorar la técnica anterior facilitando su uso, incrementando sus prestaciones, o aumentando la información presentada. Vamos a comentar aquí algunas de ellas.

3.2.1 Formato del gráfico

En Excel siempre es posible mejorar la vista de un gráfico, y en este caso lo hemos hecho utilizando un relleno con *trama del área de trazado*, y añadiendo todas las *líneas de división* en las marcas de los ejes, horizontal y vertical, entre otras [1].

3.2.2 Eje x negativo en rojo

Es posible conseguir que los valores negativos del eje Y figuren en rojo, dándole formato número personalizado con la expresión [2]:

([Azul][>=0]#,##0;[Rojo][<0]-#,##0;Estandar)

3.2.3 Controles de formulario

Para variar de una forma sencilla los valores de visualización, hemos empleado el control de formulario denominado *control de número*, para *xinicial* y *xfinal*, o la *barra de desplazamiento* para el número de *puntos* a representar. Así en la anteriormente mostrada *figura 4*, se observa que se han empleado los valores -3, 2 y 100 respectivamente. Siendo este último valor de mayor interés informático que matemático, ya que empleando un menor número de puntos, se verían tramos lineales en lugar de una línea continua.

3.2.4 Insertar símbolos y ecuaciones

Mediante el menú Insertar de Excel podemos emplear el editor de ecuaciones, o introducir algunos símbolos, lo que nos permite incluir la expresión matemática junto a la representación gráfica de $f(x)$, cómo la citada fórmula de la figura:

$$f(x) = e^{x(x-1)} \quad (1)$$

3.2.5 Nombre variable de gráfico

Para que el título del gráfico resulte variable, se toma de la celda que le indiquemos, mediante la técnica de situarnos junto a $f(x)$, en la *barra de fórmulas* y escribir *=nombre celda*, de la que toma el texto. En el ejemplo propuesto, la celda utilizada tiene de nombre *expresión*. De forma análoga se pueden incluir *formas* con textos cambiantes (*figura 8*), lo que mejora la visibilidad de la función representada y de aquellos comentarios adicionales que se deseen hacer [3].

3.2.6 WordArt variable

También podemos crear un WordArt variable, que nos permita presentar el contenido de una celda en forma de cuadro de texto con este formato y añadirlo donde deseemos, todo ello de forma automatizada. En la *figura 8* se puede observar que aparece el texto *Recta@* sobre el gráfico de forma resaltada. Para poder realizar incrustaciones de este tipo se requiere programación VBA. (Véase en el anexo A) [7].

3.2.7 Hoja entera de gráfico

Una vez elaborada la representación, es posible mostrarla a una página completa en una hoja gráfica, para así conseguir un mayor detalle de la misma. Utilizaremos la opción de F11, copiar sobre el gráfico, y pegaremos sobre dicha hoja gráfica [2].

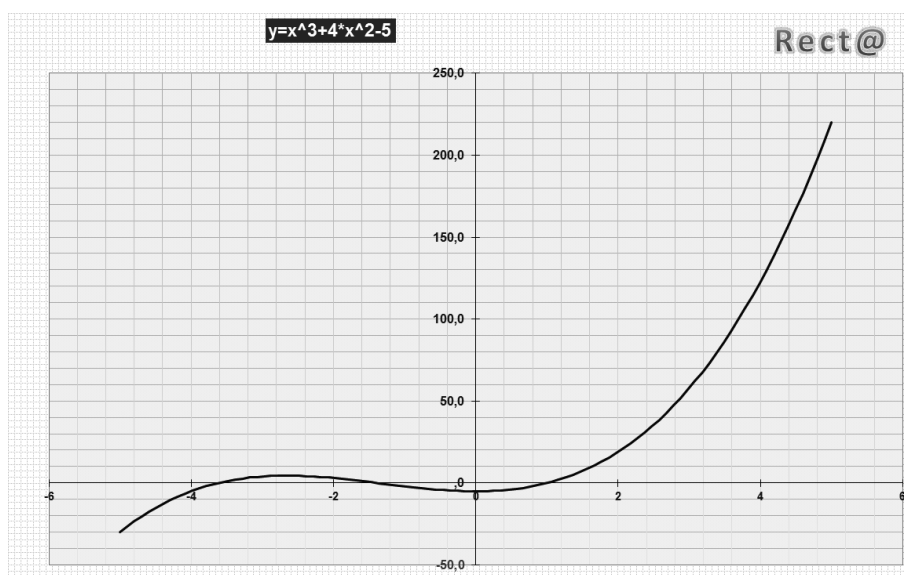


Figura 8. Gráfico en hoja completa

3.2.8 Cámara de fotos

También nos encontramos que a la hora de exportar el gráfico sobre la misma hoja o sobre otra del libro, Excel permite copiar y pegar *hipervínculos gráficos* dinámicos, lo cual se realiza de forma muy sencilla utilizando la herramienta denominada *cámara de fotos*. Existe la posibilidad de añadir su icono a la barra de herramientas de acceso rápido, para mayor comodidad a la hora de utilizarlo [3].

Su uso es tan sencillo como marcar la zona de la hoja, con tablas, gráficos, etc., y a continuación pulsar el icono de la cámara. Sobre una zona de la hoja aparecerá la imagen capturada, que además también es dinámica, lo que implica que si variamos los datos o el objeto que parezca en ella, se modificará automáticamente en la *imagen fotografiada*. Las figuras del presente artículo han sido capturadas mediante este sistema. Es aconsejable incluir en la hoja de cálculo imágenes redimensionadas a tamaño menor para mostrarlas como iconos junto a la función a representar.

3.2.9 Botones de elección de la función

Con el fin de automatizar la representación de un listado de funciones, se han añadido junto a la expresión de cada una de las funciones, un botón de macro (ver programación VBA en el Anexo B). Así, por ejemplo en la hoja de ejemplo presentada, al pulsar el botón de F2 se representa la función polinómica $x^3 + 4x^2 - 5$ (figura 9).

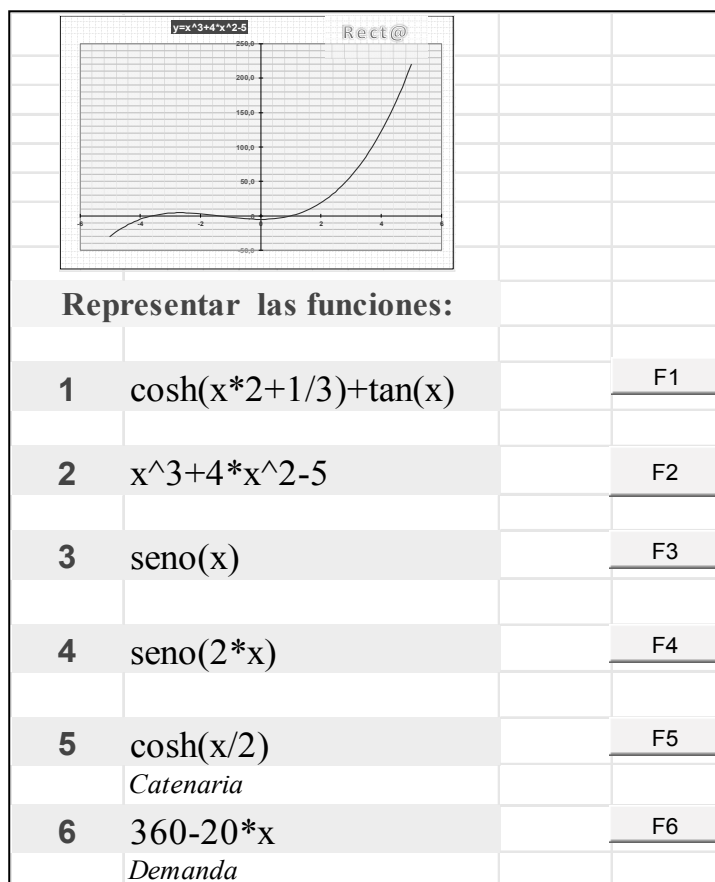


Figura 9. Representación obtenida al pulsar F2

3.3. Representación automática desde otros programas de Office

Trabajando en la programación para poder representar de forma automática las funciones, nos planteamos como siguiente objetivo el poder realizar las gráficas directamente desde su enunciado en el procesador de textos Word. La primera solución que encontramos fue la de insertar desde la opción *Objeto/Crear desde un archivo*, la hoja de Excel ya que aparece totalmente operativa, incluidos los controles, en la hoja de Word, para que podamos escribir la expresión a dibujar en la gráfica. En la *figura 10*, se muestra un ejemplo realizado con dicha técnica.

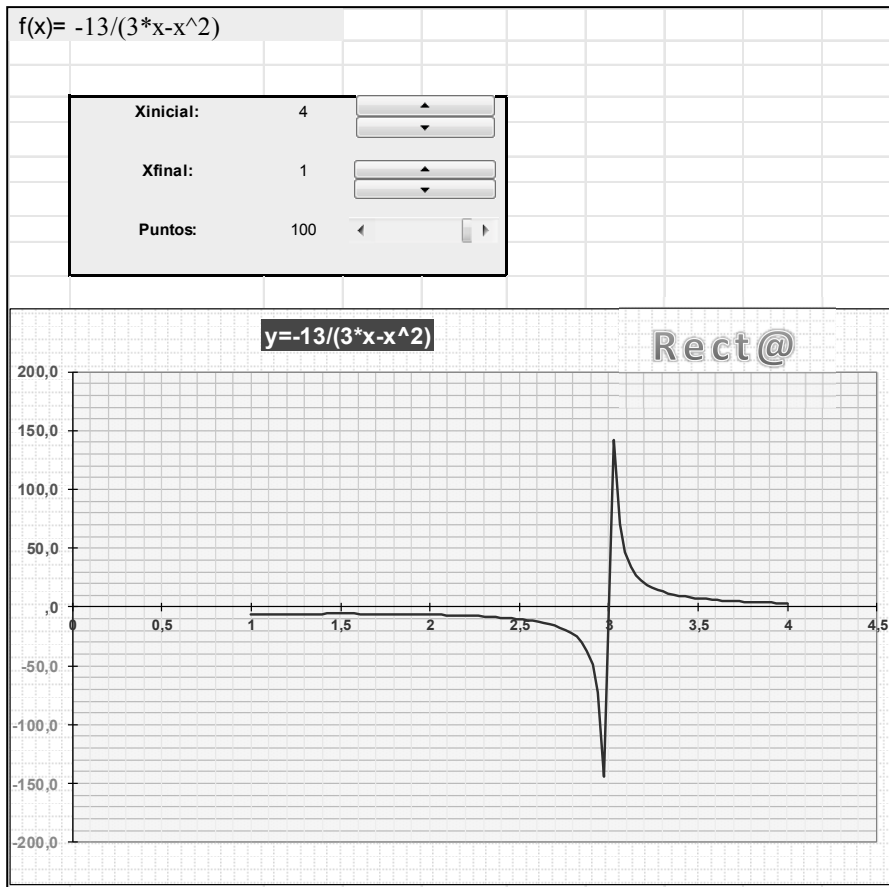


Figura 10. Insertar objeto de hoja de cálculo en Word

La siguiente versión realizada permite realizar la gráfica sin necesidad de incrustar la hoja de cálculo, ya que simplemente escribiendo la expresión a representar y pulsando el “botón Gráfica”, mediante la programación VBA que se presenta en el Anexo C, y que también funciona para el procesador de textos Word, la traslada a Excel y devuelve su gráfica. Todo ello de forma transparente al usuario, que puede así ver muy rápidamente la forma gráfica final de una función (figura 11).

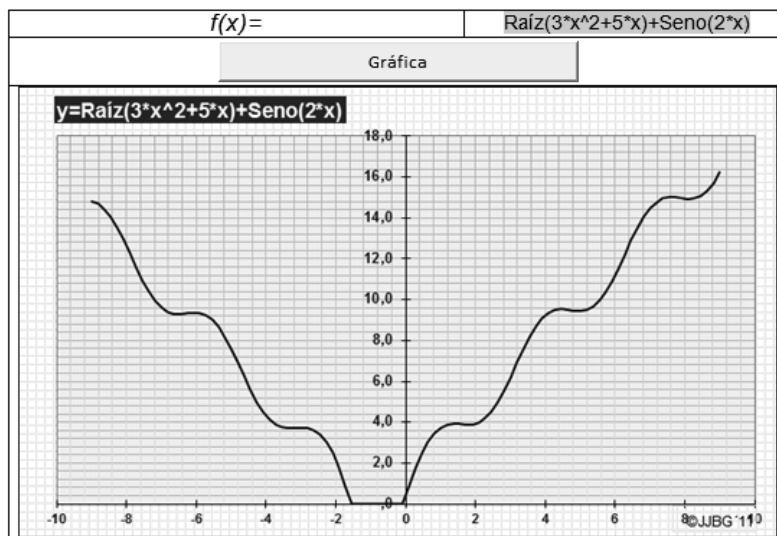


Figura 11. Gráfica de funciones desde Word

El siguiente objetivo fue trasladar esta técnica al programa de presentaciones PowerPoint, ya que es posible representar una función utilizando el “botón flecha Gráfica” desde el Modo Presentación de la aplicación. Esto nos permite realizar representaciones de gráficos de funciones en el mismo momento que se esté realizando exposición real. En la figura 12 se muestra como queda la captura de la expresión de la función en texto para una exposición con letras grandes para mejor visualización en la proyección, y el botón de flecha gráfica utilizado, así como el resultado que devuelve para dicha función. En el Anexo D se incluye el macro elaborado al efecto.

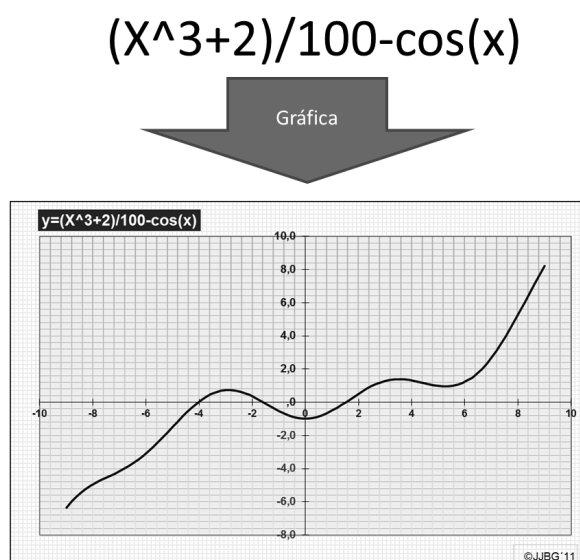


Figura 12. Gráfica de funciones desde PowerPoint

4. Aplicaciones prácticas

Creemos que resulta evidente la aplicación del uso de estos gráficos en docencia sobre todo utilizándolo en las asignaturas que sean básicamente de matemáticas. Hemos comprobado el funcionamiento de la técnica presentada en funciones con varias ramas, obtenida de enunciados reales de clase, para probar dicha aplicabilidad, obteniéndose un resultado muy satisfactorio (véase anterior figura 10).

Al estar nuestras materias incluidas en los Grados de Economía y Administración y Dirección de Empresas, pensamos que la representación de funciones de tipo económico puede resultar muy interesante para los alumnos, fundamentalmente por la posibilidad de variar parámetros que les permitan alcanzar una mayor comprensión de los mismos. A modo de ejemplo la figura 13, muestra una función de demanda realizada con esta metodología.

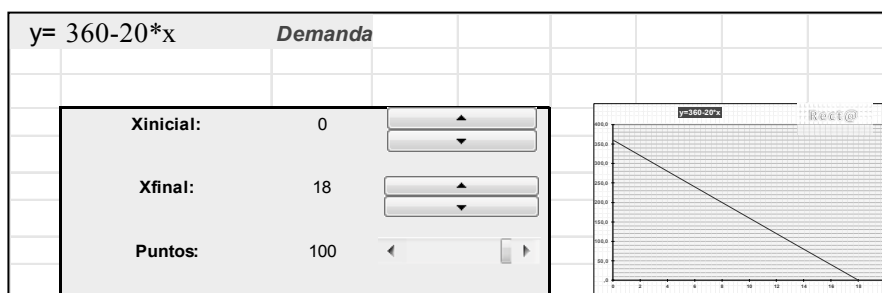


Figura 13. Representación de la función demanda

5. Mejoras al modelo

Cualquier método siempre puede y debe ser mejorado, bien optimizándolo técnicamente desde el punto de vista informático, o incrementando sus prestaciones. En este último sentido, otra de las propuestas que realizamos consiste en poder representar dos gráficas de forma simultánea (con colores y estilos diferentes), lo que permite su posterior comparación, o visualizar las intersecciones entre ambas.

En la figura 14 se muestran por separado las dos funciones, polinómica y trigonométrica que componían la gráfica representada en el ejemplo utilizado al exponer su utilización desde PowerPoint.

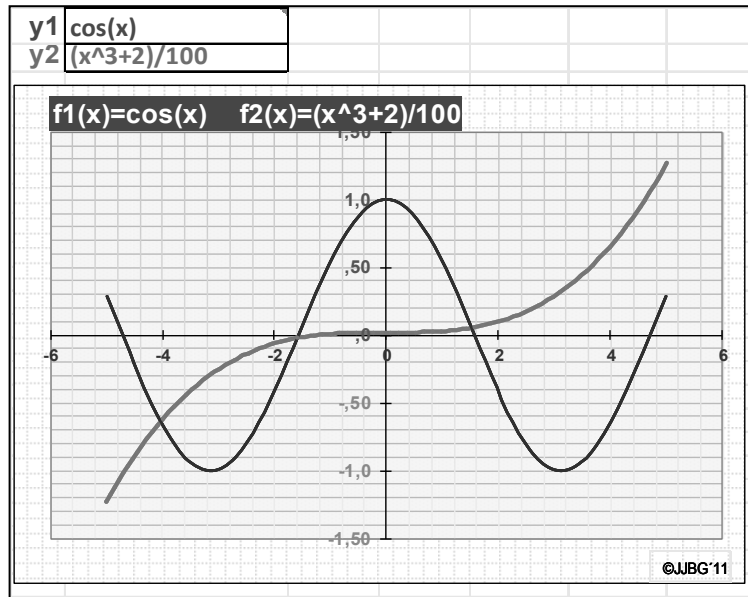


Figura 14. Dos gráficas simultáneamente, trigonométrica y polinómica

Otro caso realizado se muestra en la figura 15, dónde podemos visualizar una parábola y una tangente a ella (programación en el Anexo E). Una versión posterior permitió además representar, de forma automática las derivadas numéricas, de una o dos curvas representadas, con colores y estilos diferenciados, figura 16, y que se desarrolla en el Anexo F.

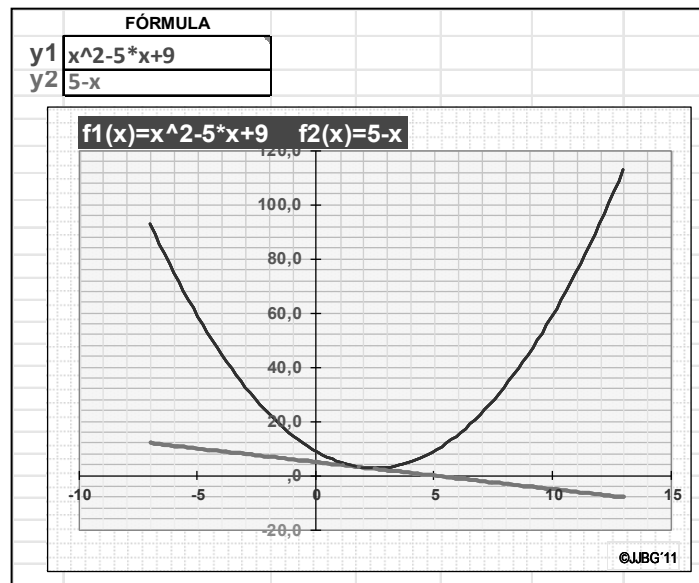


Figura 15. Gráfica de una curva y tangente a la misma

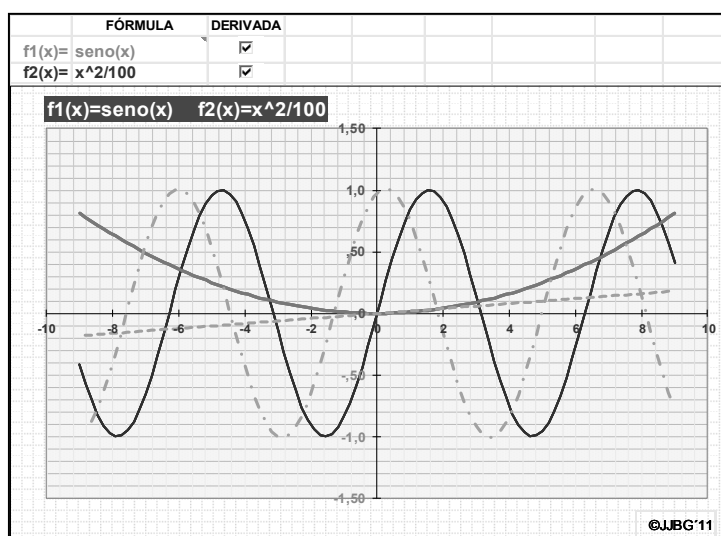


Figura 16. Dos gráficas con sus derivadas numéricas

6. Conclusiones

Se ha demostrado cómo la utilización avanzada de la hoja de cálculo, permite elaborar herramientas que sean de utilidad tanto a los profesionales, como para el autoaprendizaje de los alumnos. La elaboración de gráficos más complejos, pero a la vez más explicativos mediante la programación con VBA o mediante el uso de funciones poco usuales en el manejo de Excel, permite obtener una mayor información, más visual, al tiempo que una herramienta docente adicional, dinámica y totalmente modificable, tanto por el profesor como por los alumnos que la utilicen. Seguimos investigando la forma de incorporar cualquier otro tipo de información que pueda ser útil de mostrar en las gráficas que se pueden realizar con Excel.

Conscientes de que el método presentado debe ser validado para su utilización docente, se ha mostrado a alumnos de las tutorías de matemáticas del Centro Regional de la UNED en Cartagena, tanto del Grado de ADE como en el de Economía, habiéndose testado su facilidad de funcionamiento y su interés por parte de dichos discentes. Además, se están realizando pruebas en el aula de informática con los alumnos de un grupo de matemáticas empresariales de la Facultad de CC. de la Empresa de la Universidad Politécnica de Cartagena, con el fin de evaluar también su utilidad, al tiempo que solicitarles sugerencias de mejora para incorporar en futuras versiones.

Referencias bibliográficas

1. J.J Bernal García, Aportaciones para la mejora de la presentación gráfica de datos cuantitativos en Excel, *Revista Rect@. Vol. 16: 41242-412*, XVI Congreso ASEPUMA y IV Encuentro Internacional. (Cartagena, 2008).
2. J.J. Bernal García, Nuevas contribuciones a la mejora de la representación gráfica en Excel, *Anales ASEPUMA. N° 18.2010*, XVIII Congreso ASEPUMA y VI Encuentro Internacional (Santiago de Compostela, 2010)
3. J.J. Bernal García, Posibilidades especiales de selección de gráficos en Excel, *Anales ASEPUMA. N° 19*, VII Encuentro Internacional de Profesores Universitarios de Matemáticas para la Economía y la Empresa (Valencia, 2011)
4. B. Jelen y T. Syrstad, Excel. Macros y VBA. Trucos esenciales, *Ed Anaya*, (Madrid, 2008)
5. D. y R. Hawley, Excel 2007. Los mejores trucos, *Anaya-O'Reilly*, (Madrid, 2008)
6. J. Walkenbach, Excel 2010. Los mejores trucos, *Anaya multimedia*, (Madrid, 2011)
7. J. Walkenbach, Excel 2010. Programación con VBA, *Anaya Multimedia.-Wiley* (Madrid, 2011)

ANEXOS

A. Programación VBA en el módulo *ThisWorkbook* para *WordArt* variable (texto en F1).

```
Private Sub Workbook_SheetChange(ByVal Sh As Object, ByVal Target As Range)
ActiveSheet.Shapes("1 Rectángulo").Select
Selection.ShapeRange.TextEffect.Text = Range("F1")
Range("F1").Select
End Sub
```

B. Programación VBA de la función F2 (análoga al resto de Fi).

```
Sub funcion2()
' poner_funcion Macro
' Acceso directo: Ctrl+Mayús+F
Range("I13").Select
    Selection.Copy
Range("B7").Select
ActiveSheet.Paste
End Sub
```

C. Programación VBA en Word, para invocar una hoja Excel que realiza la gráfica y la devuelve a él.

```
Sub InsertaGráfica()
Application.ScreenUpdating = False
Selection.HomeKey Unit:=wdStory, Extend:=wdMove
Selection.MoveRight Unit:=wdCell
Selection.Copy
Call LeerExcel
Selection.MoveDown Unit:=wdLine, Count:=2
Selection.Delete Unit:=wdCharacter, Count:=1
Selection.PasteSpecial , , , wdPasteBitmap
Application.ScreenUpdating = False
Application.ScreenRefresh
End Sub
Private Sub LeerExcel()
On Error Resume Next
'dimensiones
Dim xlApp As Excel.Application
Dim xlLibro As Excel.Workbook
Dim xlHoja As Excel.Worksheet
'abrirprograma Excel
Set xlApp = New Excel.Application
xlApp.Visible = True
'abrir el archivo Excel
'(archivo en la misma carpeta)
Set xlLibro = xlApp.Workbooks.Open(ThisDocument.Path & _
"\GRAFdesdeWORD.xlsm", True, True, , "")
Set xlHoja = xlApp.Worksheets(1)
xlHoja.ChartObjects("Gráfico 1").Copy
'cerramos el archivo Excel
xlLibro.CloseSaveChanges:=False
xlApp.Quit
'reset variables de los objetos
Set xlHoja = Nothing
Set xlLibro = Nothing
```

```
Set xlApp = Nothing
End Sub
```

D. Programación VBA en PowerPoint, para invocar una hoja Excel que realiza la gráfica y la devuelve a dicha aplicación.

```
Sub InsertaGrafica()
On Error Resume Next
ActivePresentation.Slides(1).Shapes(3).Delete
ActivePresentation.Slides(1).Shapes(1).TextFrame.TextRange.Copy
'xls2ppt
Call LeerExcel
'ActivePresentation.Slides.Paste
Set oSlide = ActivePresentation.Slides(1)
Set oShape = oSlide.Shapes.PasteSpecial(ppPastePNG)
oShape.Top = 200
oShape.Left = 110
oShape.Width = 500
oShape.Height = 300
End Sub
Private Sub LeerExcel()
On Error Resume Next
'dimensiones
Dim xlApp As Excel.Application
Dim xlLibro As Excel.Workbook
Dim xlHoja As Excel.Worksheet
'abrirprograma Excel
Set xlApp = New Excel.Application
'xlApp.Visible = True
'abrir el archivo Excel
'(archivo en la misma carpeta)
Set xlLibro = xlApp.Workbooks.Open(ActivePresentation.Path & _
"\GRAFdesdeWORD.xlsm", True, True, , "")
Set xlHoja = xlApp.Worksheets(1)
xlHoja.ChartObjects("Gráfico 1").Copy
'cerramos el archivo Excel
xlLibro.CloseSaveChanges:=False
xlApp.Quit
'reset variables de los objetos
Set xlHoja = Nothing
Set xlLibro = Nothing
Set xlApp = Nothing
End Sub
```

E. Programación VBA para permitir realizar gráficas simultáneas.

```
Private Sub copia_imagen()
Dim objHolder As ChartObject
Dim objTemp As Object
Dim sngWidth As Single
Dim sngHeight As Single
Dim MyShape As String
Dim Filename1 As String
Dim shtTemp As Worksheet
Set shtTemp = ActiveSheet
Filename1 = "test_pic.gif"
MyShape = "7 Imagen"
Application.ScreenUpdating = False
```

```

Set objTemp = shtTemp.Shapes(MyShape)
' Crea un gráfico temporal como contenedor de la imagen de colores para exportar
Charts.Add
ActiveChart.ChartType = xlColumnClustered
ActiveChart.Location Where:=xlLocationAsObject, Name:=shtTemp.Name
Set objHolder = shtTemp.ChartObjects(shtTemp.ChartObjects.Count)
sngWidth = objTemp.Width
sngHeight = objTemp.Height
With objHolder
    .Chart.ChartArea.Border.LineStyle = xlNone
    .Width = sngWidth + 20
    .Height = sngHeight + 20
objTemp.Copy
    .Chart.Paste
    With .Chart.Shapes(1)
        .Placement = xlMove
        .Left = -4
        .Top = -4
    End With
    .Width = sngWidth + 1
    .Height = sngHeight + 1
    .Chart.Export Filename1, "GIF"
    .Chart.Shapes(1).Delete
    .Delete
End With
End Sub

```

F. Programación VBA para permitir realizar dos gráficas y sus derivadas numéricas a la vez.

```

Private Sub copia_imagen()
Dim objHolder As ChartObject
Dim objTemp As Object
Dim sngWidth As Single
Dim sngHeight As Single
Dim MyShape As String
Dim Filename1 As String
Dim shtTemp As Worksheet
Set shtTemp = ActiveSheet
Filename1 = "test_pic.gif"
MyShape = "7 Imagen"
Application.ScreenUpdating = False
Set objTemp = shtTemp.Shapes(MyShape)
' Crea un gráfico temporal como contenedor de la imagen de colores para exportar
Charts.Add
ActiveChart.ChartType = xlColumnClustered
ActiveChart.Location Where:=xlLocationAsObject, Name:=shtTemp.Name
Set objHolder = shtTemp.ChartObjects(shtTemp.ChartObjects.Count)
sngWidth = objTemp.Width
sngHeight = objTemp.Height
With objHolder
    .Chart.ChartArea.Border.LineStyle = xlNone
    .Width = sngWidth + 20
    .Height = sngHeight + 20
objTemp.Copy
    .Chart.Paste
    With .Chart.Shapes(1)
        .Placement = xlMove

```



```

.Left = -4
.Top = -4
End With
.Width = sngWidth + 1
.Height = sngHeight + 1
.Chart.Export Filename1, "GIF"
.Chart.Shapes(1).Delete
.Delete
End With
End Sub
Sub Barradedesplazamiento12_AlCambiar()
temp = Range("C14")
Range("C14") = ""
Range("C14") = temp
End Sub
Sub Macro1()
" Macro1 Macro
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveChart.Axes(xlCategory).Select
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveChart.SeriesCollection(11).Select
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveChart.SeriesCollection(10).Select
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveSheet.ChartObjects("Gráfico 1").Activate
ActiveChart.SeriesCollection(11).Select
ActiveSheet.ChartObjects("Gráfico 1").Activate
End Sub

```