

POVIX: Una interfaz gráfica de usuario (GUI) en MatLab para calcular el polinomio de Villarreal

POVIX: A graphical user interface (GUI) in MatLab to calculate the Villarreal polynomial

Lenin Quiñones Huatangari¹, Manuel Emilio Milla Pino², Leonardo Damián Sandoval³

RESUMEN

El matemático peruano Federico Villarreal en 1879 propuso una fórmula para calcular la potencia n-ésima de un polinomio $P(x)$, denominado Polinomio de Villarreal. El objetivo del presente trabajo fue crear una interfaz gráfica para usuario que nos permita calcular la potencia n-ésima entera de un polinomio, a partir de la fórmula propuesta por el científico. Para ello se utilizó la herramienta matemática MatLab v9.2 (R2017a) sobre el sistema operativo Windows 10. Con ayuda del Toolbox GUIDE (Graphical User Interface Development Environment) se diseñó la GUI basada en la fórmula propuesta por Villarreal. Además, con MatLab Compiler™ v9.01, se obtuvo una aplicación autónoma, capaz de ejecutarse en cualquier computadora que tenga una distribución de Windows 10. El resultado obtenido del presente trabajo fue la interfaz gráfica para usuario denominada POVIX, la cual los resultados que estima son los mismos que se ha obtenido por otros métodos analíticos.

Palabras clave: polinomio de Villarreal, potencia de polinomios, GUI, MatLab, álgebra.

ABSTRACT

Peruvian mathematician Federico Villarreal in 1879 proposed a formula to calculate the nth power of a polynomial $P(x)$, called Villarreal's Polynomial. The objective of this work was to create a graphical user interface that allows us to calculate the entire nth power of a polynomial, based on the formula proposed by the scientist. For this, the MatLab v9.2 (R2017a) mathematical tool on the Windows 10 operating system was used. With the help of the GUIDE Toolbox (Graphical User Interface Development Environment), the GUI was designed based on the formula proposed by Villarreal. In addition, with MatLab Compiler™ v9.01, an autonomous application was obtained, capable of running on any computer that has a Windows 10 distribution. The result obtained from this work was the graphical user interface called POVIX, which results in Estimates are the same as those obtained by other analytical methods.

Keywords: polynomial of Villarreal, power of polynomials, GUI, Matlab, algebra.

¹Licenciado en Matemáticas. Docente Ordinario de la Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas. Email: lenin.quinones@untrm.edu.pe

²Doctor en Ciencias. Docente de la Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas. Email: manuel.milla@untrm.edu.pe

³Licenciado en Matemáticas. Docente Ordinario de la Universidad Nacional de Jaén. Email: leomatepe@hotmail.com

I. INTRODUCCIÓN

Las interfaces gráficas de usuario o interfaces de usuario (GUI) se utilizan para proporcionar a los usuarios un enfoque amigable y visual a las aplicaciones de software, además sirven para especificar todos los parámetros de entrada y una mayor flexibilidad de configuración (Depcik & Assanis, 2005). Permiten un control sencillo (con uso de ratón), lo cual elimina la necesidad de aprender un lenguaje y escribir comandos a fin de ejecutar una aplicación («GUI de MATLAB», 2017).

Para diseñar GUI con el programa MatLab se utiliza la herramienta GUIDE (Graphical User Interface Development Environment), (Hunt, Lipsman, & Rosenberg, 2014), (Lent, 2013). Hasta el momento se han implementado diversas y variadas GUI que abordan diferentes áreas del conocimiento, como por ejemplo GUI para medicina, mecánica, educación, etc., (Gözel, Eminoglu, & Hocaoglu, 2008), (Fedák, Balogh, & Zásalický, 2012), (Giurma et al., 2009), (Egert et al., 2002), (Tibor, Fedak, & Durovský, 2011), (Tintaya, 2014).

El problema de elevar un polinomio a una potencia dada presenta un interés teórico más bien que práctico, en cualquier caso, y sea cual fuere el método utilizado, resulta una operación en extremo laboriosa cuando tanto el grado del polinomio como el exponente de la potencia son relativamente grandes (Alberto Rodríguez, 1957). Analizaremos los diversos procedimientos para resolver este problema. Sea un polinomio de grado m elevado a la potencia n , es decir:

$$(a_0 + a_1 + a_2 + \dots + a_i + \dots + a_m)^n \quad (1)$$

Definición de potencia: Consiste en multiplicar por sí mismo el polinomio tantas veces como unidades tiene el exponente.

Usando el Binomio de Newton: Se procede a descomponer el polinomio dado en un binomio cuyo primer término es el primero del polinomio base, y el segundo, la suma de los restantes. Se aplica entonces la fórmula para elevar un binomio a una potencia, y en el desarrollo aparecerán ahora potencias de polinomios cuyo grado es en una unidad inferior al grado del polinomio base, y que a su vez se descomponen en otros binomios, procediendo se así sucesivamente hasta llegar a potencias de auténticos binomios (Carbajal Ordinola, 2016).

Teorema Multinomial: Para cualquier entero

positivo m y cualquier entero no negativo n , se verifica que:

$$(a_0 + a_1 + a_2 + \dots + a_i + \dots + a_m)^n = \sum_{k_1+k_2+\dots+k_m=n} \binom{n}{k_1, k_2, \dots, k_m} \prod_{1 \leq t \leq m} a_t^{k_t} \quad (2)$$

Donde

$$\binom{n}{k_1, k_2, \dots, k_m} = \frac{n!}{k_1! k_2! \dots k_m!}$$

Es un coeficiente multinomial. Este teorema es la misma Fórmula de Leibnitz para la potencia de un polinomio (Elorduy, 2017). El sustento teórico son las permutaciones con repetición que se les denomina coeficiente multinomiales.

Utilizando polinomios de McLaurin: Sea $f(x)$ una función esta puede ser expresada por la ecuación (3), (Purcell, Rigdon, & Varberg, 2007).

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \dots + \frac{f^{(n)}(0)}{n!}x^n + \dots \quad (3)$$

donde $f(x)$ representa la expresión algebraica (1), y requiere cálculo de las diversas derivadas sucesivas de dicha función. En el caso de un polinomio de grado cuarto elevado a la cuarta potencia, exigiría el cálculo de dieciséis derivadas, en las que luego habría que particularizar para $x=1$.

Pedro Martir Armet el 17 de febrero de 1819 expone en la Real Academia de Ciencias y Artes de Barcelona la Memoria 12 de dicha institución (M. Montanuy Fillat et al, 1990), que era un procedimiento para resolver la expresión (1), haciendo uso de las combinaciones y permutaciones (Armet, 1822).

Sin embargo el polinomio $a_0 + a_1 + a_2 + \dots + a_i + \dots + a_m$, se puede considerar como una función de la variable x , y ordenar conforme a las potencias crecientes de esta variable. Si el polinomio careciera de tal variable, se podrá considerar cada uno de sus términos como coeficientes de dichas potencias crecientes de una variable auxiliar x ; al resolver la potencia de este polinomio bastaría hacer $x = 1$, para obtener la solución algebraica (1).

Federico Villarreal fue un reconocido estudioso de las ciencias exactas, nació en Túcume, departamento

de Lambayeque - Perú en 1850 y desde muy temprana edad mostró un interés especial por las matemáticas (noticias.universia.edu.pe, 2017). Este sabio políglota, matemático y docente universitario es considerado uno de los seis personajes destacados de la ciencia y tecnología del Perú (Segovia, 2013) y es el primer doctor en matemática en el Perú. Descubre en 1879 un método para elevar un polinomio a una potencia cualquiera, a diferencia de otros métodos que solo se aplica para números naturales, que luego recibirá el nombre de "Polinomio de Villarreal" (López, 2014).

En este contexto la investigación se trazó como objetivo implementar una GUI en MatLab que nos permita calcular y mostrar la figura del polinomio potencia obtenida al usar la fórmula propuesta por Villarreal en 1879. Esta herramienta metodológica nos permite conocer y revalorar el trabajo realizado por este científico peruano.

II. MATERIALES Y MÉTODOS

Toda la programación del GUI se realizó en una computadora con una distribución Windows a 64 bits.

La interfaz gráfica de usuario se realizó con el comando "guide" de MatLab, el cual crea dos archivos: El primer archivo *.fig ("fig file"), en donde pueden insertarse ejes, botones, deslizadores, texto de edición, menús, etc. Cada uno de los cuales tienen ciertas propiedades y realiza una función determinada. El segundo archivo, *.m donde se encuentra todas las acciones de cada elemento de la GUI, además se puede acceder del archivo *.fig al *.m, mediante ("Callback Function") de cada elemento.

Usando el compilador MatLab Compiler™ v9.01, se logró obtener una aplicación autónoma .exe capaz de ejecutarse en cualquier computadora con una distribución Windows que no tenga necesariamente el MatLab instalado.

Para el presente estudio daré a conocer el caso particular, cuando n es un número natural. Por lo tanto, el polinomio de Villarreal puede ser expresado:

Sea $n \in \mathbb{N}$ y $P(x) = a_0 + a_1x^1 + a_2x^2 + \dots + a_mx^m$ un polinomio de grado m en la variable x . Entonces

$$[P(x)]^n = a_0^n + \sum_{k=1}^{mn} b_k x^k \quad (4)$$

Donde

$$b_k = \sum_{j=1}^n b_{k-j} C_j \left(\frac{D_j}{k} - 1 \right). \text{ Además de } C_j = \frac{a_j}{a_0}, D_j = j(m+1), \forall j = 1 \dots n$$

III. RESULTADOS

POVIX consta de ocho archivos:

Tres script: El primer script está compuesta por todas las acciones del GUI, que da la bienvenida al programa (Povix.m), el segundo script las acciones que se puedan realizar al abrir el manual de ayuda al usuario (manual.m) y el tercer script contiene la programación en lenguaje .m para calcular el polinomio de Villarreal.

Tres GUIs: La primera GUI contiene la interfaz que da la bienvenida al programa y es lo primero que el usuario va a ver (Povix.fig) ver Figura 1, la segunda GUI nos permitirá mostrar el manual de ayuda sobre el Polinomio de Villarreal, ver Figura 2 y finalmente la tercera GUI nos permitirá ingresar los dos parámetros necesarios para calcular el problema, el $P(x)$, la potencia y finalmente en esta GUI se muestra la gráfica del polinomio y la expresión algebraica del mismo en la parte superior de la figura ver Figura 3.

El instalador del compilador empleado, necesario para computadoras que no tengan instalado MatLab.

Un archivo Povix.exe el cual nos permite utilizar POVIX sin necesidad de haber instalado.

Figura 1. Pantalla de bienvenida a POVIX

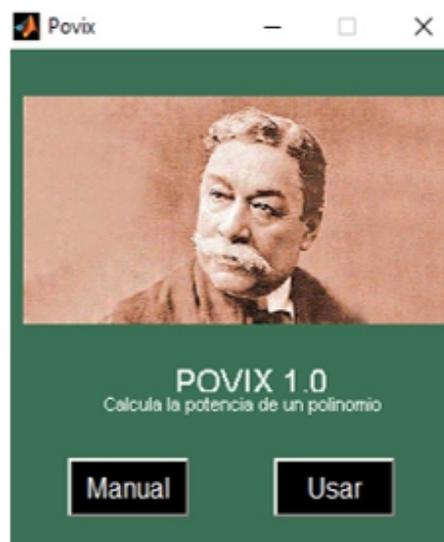


Figura 2. Pantalla del manual de POVIX

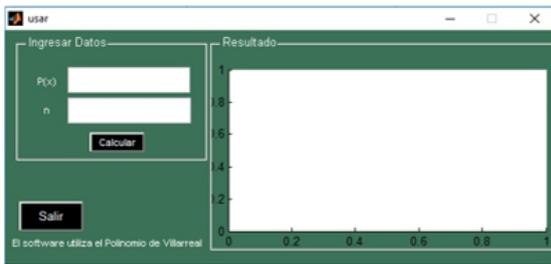


Figura 3. Pantalla de uso de POVIX

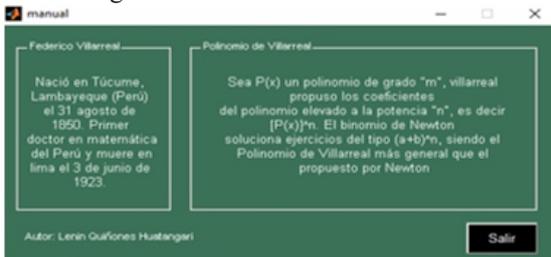
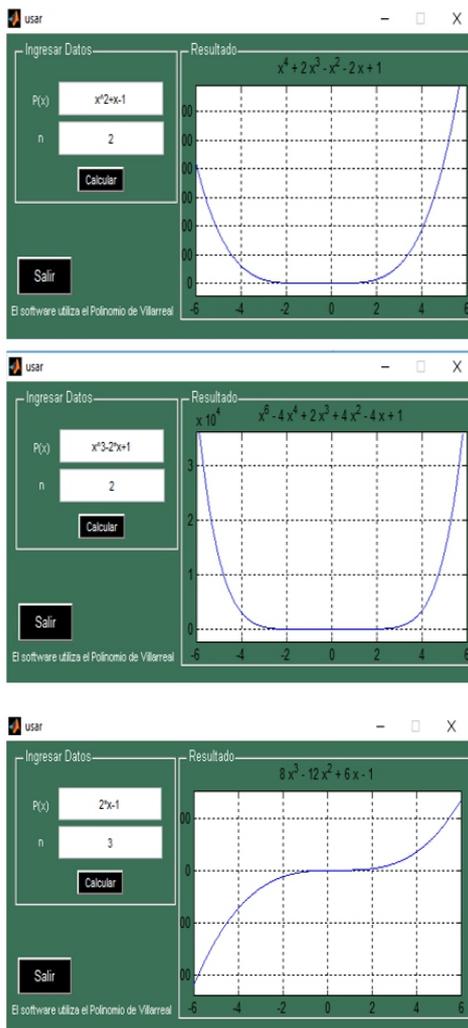


Figura 4. Resultados de calcular la potencia de un polinomio, usando POVIX



IV. DISCUSIÓN

Al ejecutar POVIX, este nos va a mostrar una GUI de entrada, ver Figura 1, donde se puede elegir cualquiera de los tres botones y cada uno de ellos se obtiene diferentes acciones. Si presionamos el botón Manual, obtendremos un resumido manual, como se puede observar en la Figura 2.

La Figura 3 es el resultado elegir el botón usar, el cual permite al usuario seleccionar el polinomio y la potencia que desea procesar. Finalmente, el usuario podrá hacer click sobre el botón Calcular con el ícono negro y comenzar el cálculo.

La Figura 4 muestra una secuencia de los cálculos realizados para diversos polinomios a diferentes potencias. En dicha secuencia se puede identificar de forma sencilla la generación de diferentes polinomios a lo largo del medio de simulación. En la captura se puede ver las gráficas de dicho cálculo.

V. CONCLUSIONES

Este trabajo nos permitió calcular y graficar el polinomio que se obtiene del resultado de dicho problema utilizando la fórmula propuesta por Federico Villarreal en 1879, para el caso particular cuando n pertenece \mathbb{N} . La interfaz gráfica de usuario POVIX nos permite un control sencillo, solo con uso de ratón, lo cual elimina la necesidad de aprender el lenguaje .m y escribir comandos a fin de ejecutar dicha aplicación. Los resultados obtenidos en el software son 100% iguales con los calculados por el autor. Se obtuvo una aplicación portable que incluye su propio compilador, permitiendo que funcione en cualquier computadora con una distribución de Windows a 64 bit. Finalmente se puede concluir que POVIX es una herramienta de aprendizaje para comprender teórica y geoméricamente la potenciación de un polinomio y específicamente el Polinomio de Villarreal.

VI. REFERENCIAS BIBLIOGRÁFICAS

Alberto Rodríguez. (1957). Potencias de un Polinomio. rev.colomb.mat, Volumen 5, Número 1- 2, p. 37- 43, 1960. ISSN electrónico 2357-4100. ISSN impreso

- 0034-7426.
- Armet, P. (1822). *Memoria que demuestra el modo de hallar el resultado de las potencias de los polinomios: leída en la Junta que celebra la Academia Nacional de Ciencias Naturales el 17 de Febrero de 1819*. Imp. Vda e Hijos de Antonio Brusi.
- Carbajal Ordinola, D. (2016). *Polinomio de Villarreal*. Recuperado 28 de octubre de 2018, a partir de https://www.researchgate.net/publication/305115632_Polinomio_de_Villarreal.
- Depcik, C., & Assanis, D. N. (2005). Graphical user interfaces in an engineering educational environment. *Computer Applications in Engineering Education*, 13(1), 48–59.
- Egert, U., Knott, T., Schwarz, C., Nawrot, M., Brandt, A., Rotter, S., & Diesmann, M. (2002). MEA-Tools: an open source toolbox for the analysis of multi-electrode data with MATLAB. *Journal of neuroscience methods*, 117(1), 33–42.
- Elorduy, G. (2017, septiembre 28). Potencia de un polinomio (Fórmula de Leibnitz). Recuperado 22 de octubre de 2017, a partir de <https://eluniversomatematicoblog.wordpress.com/2017/09/28/potencia-de-un-polinomio-formula-de-leibnitz/>
- Fedák, V., Balogh, T., & Záskalický, P. (2012). Dynamic simulation of electrical machines and drive systems using matlab gui. En *MATLAB-A Fundamental Tool for Scientific Computing and Engineering Applications-Volume 1*. InTech.
- Giama, B., Vijayajothi, P., Kahar, O., Rafiq, M., Kadir, A., & Kanesan, M. (2009). Graphical user interface (GUI) in MatLab for solving the pulsatile flow in blood vessel. *CFD Letters*, 1(1), 50–58.
- Gözel, T., Eminoglu, U., & Hocaoglu, M. H. (2008). A tool for voltage stability and optimization (VS&OP) in radial distribution systems using matlab graphical user interface (GUI). *Simulation Modelling Practice and Theory*, 16(5), 505–518.
- GUI de MATLAB. (2017). Recuperado 23 de octubre de 2018, a partir de <https://es.mathworks.com/discovery/matlab-gui.html>
- Hunt, B. R., Lipsman, R. L., & Rosenberg, J. M. (2014). *A guide to MATLAB: for beginners and experienced users*. Cambridge University Press.
- Lent, C. S. (2013). *Learning to program with MATLAB: Building GUI tools*. John Wiley & Sons.
- López, R. V. (2014). Federico Villarreal y Villarreal. *Pro Mathematica*, 14(27-28), 135-154. Recuperado a partir de <http://revistas.pucp.edu.pe/index.php/pro-mathematica/article/view/8163>
- M. Montanuy Fillat et al (último). (1990). La Matemática en la Real Academia de Ciencias y Artes de Barcelona: Las memorias durante el periodo 1770-1890. *Universidad de Barcelona*, 13, 107-130.
- noticias.universia.edu.pe. (2017). 6 personajes destacados de la ciencia y tecnología del Perú. Recuperado 21 de octubre de 2017, a partir de <http://noticias.universia.edu.pe/cultura/noticia/2015/04/01/1122648/6-personajes-destacados-ciencia-tecnologia-peru.htm>
- Purcell, E. J., Rigdon, S. E., & Varberg, D. E. (2007). *Cálculo*. Pearson Educación
- Segovia, J. L. (2013). Federico Villarreal Villarreal Un hombre de ciencia. *Cátedra Villarreal*, 1(1). <https://doi.org/10.24039/cv20131112>
- Tibor, B., Fedak, V., & Durovský, F. (2011). Modeling and simulation of the BLDC motor in MATLAB GUI. En *Industrial Electronics (ISIE), 2011 IEEE International Symposium on* (pp. 1403–1407). IEEE.
- Tintaya, C. O. J. (2014). Procesamiento Digital de Señales Sísmicas con Matlab. *Revista de Investigación de Física*, 10(02). Recuperado a partir de <http://revistasinvestigacion.unmsm.edu.pe/index.php/fisica/article/view/8581>