

SOFTWARE EDUCATIVO PARA CÁLCULO NUMÉRICO

María E. Ascheri^a, Rubén A. Pizarro^a, Gustavo J. Astudillo^a, Pablo García^a, María E. Culla^a

^a *Facultad de Ciencias Exactas y Naturales, Universidad Nacional de La Pampa, Avenida Uruguay 151, Santa Rosa, La Pampa, Argentina, mavacheri@exactas.unlpam.edu.ar, <http://www.exactas.unlpam.edu.ar>*

Palabras clave: educativo, herramientas gratuitas, FLOSS, cálculo numérico.

Resumen. En un Proyecto de Investigación ya finalizado, elaboramos un software educativo utilizando herramientas computacionales de tipo comercial, en el cual implementamos los métodos numéricos clásicos para la resolución de ecuaciones no lineales. Usamos este software en el desarrollo de Cálculo Numérico que se dicta para estudiantes de Ingeniería Civil, Licenciatura en Física y Profesorado en Matemática. Los resultados alcanzados fueron satisfactorios, obteniéndose buenas respuestas académicas, promoviendo el protagonismo del educando y facilitando el trabajo que, para alumno y profesor, supone la tarea de formación.

En un nuevo Proyecto de Investigación, comenzamos a diseñar otro software educativo utilizando herramientas computacionales gratuitas, que contenga el tema anterior e interpolación y aproximación polinomial y ajuste de curvas.

Para el desarrollo de la aplicación descrita en el párrafo anterior, nos propusimos alcanzar los siguientes objetivos:

1. Generar el contexto educativo acorde a los contenidos a desarrollar y a los objetivos propuestos en la asignatura.
2. Proporcionar una herramienta tecnológica para afianzar los conceptos teóricos y la aplicación de los métodos numéricos a través de la visualización gráfica del funcionamiento de los mismos.

En este trabajo, presentamos las características, los resultados y conclusiones del nuevo software obtenidas a partir de instancias de evaluación y de una encuesta realizada a los alumnos de Cálculo Numérico.

1 INTRODUCCION

En un Proyecto de Investigación ya finalizado, desarrollamos un software educativo utilizando herramientas computacionales de tipo comercial. En la aplicación, implementamos los métodos numéricos clásicos para la resolución de ecuaciones no lineales. Usamos este software en el desarrollo de la asignatura Cálculo Numérico que se dicta para estudiantes de las carreras Ingeniería Civil (2° año), Licenciatura en Física (3° año) y Profesorado en Matemática (3° año). Los resultados alcanzados fueron satisfactorios, obteniéndose buenas respuestas académicas, promoviendo el protagonismo del educando y facilitando el trabajo que, para estudiante y profesor, supone la tarea de formación.

Teniendo como base la experiencia alcanzada (Ascheri y Pizarro, 2005 y 2006), nos propusimos, en el ámbito de un nuevo Proyecto de Investigación, diseñar otro software educativo utilizando herramientas de tipo FLOSS¹, que contenga los temas *interpolación y aproximación polinomial y ajuste de curvas*, además de *resolución de ecuaciones no lineales*.

La nueva aplicación a elaborar debía permitirnos alcanzar los siguientes objetivos:

1. Generar el contexto educativo acorde a los contenidos a desarrollar y a los objetivos propuestos en la asignatura.
2. Proporcionar una herramienta tecnológica para afianzar los conceptos teóricos y la aplicación de los métodos numéricos a través de la visualización gráfica del funcionamiento de los mismos.
3. Poner a disposición de los estudiantes una herramienta de acceso libre y disponible a través de la Web.

En una etapa previa al diseño de esta nueva aplicación, buscamos en la Web herramientas de acceso libre destinadas a la enseñanza y el aprendizaje de los temas *resolución de ecuaciones no lineales, interpolación y aproximación polinomial y ajuste de curvas*. Comprobamos que no existían aplicaciones con las características que buscábamos. Los hallazgos se centraban, básicamente, en el cálculo numérico de los problemas a resolver. También localizamos un gran número de sitios que ofrecen los algoritmos implementados en diferentes lenguajes de programación ó páginas Web conteniendo desarrollos a partir de *applets*. En nuestra opinión, estos recursos, creemos que limitan las posibilidades de aprendizaje de los métodos ya que la mayoría de ellos son aplicados para funciones fijas. Otros sitios presentan animaciones que muestran el desarrollo de los métodos y/o arrojan los resultados de la aplicación de un método en particular en forma genérica o sobre una función específica. Este tipo de recursos, si bien pueden ser de utilidad, no actúan como herramientas que promuevan aprendizajes significativos para el proceso de enseñanza y aprendizaje de los métodos numéricos con los cuales se esté trabajando.

En virtud de los resultados que arrojaron las búsquedas realizadas (Ascheri et al., 2007), nos concentramos en evaluar un conjunto de herramientas, de tipo FLOSS, que nos permitieran desarrollar un software educativo que pudiera ser distribuido libre y fácilmente entre los estudiantes. Después de analizar varias opciones, nos decidimos por realizar la aplicación en PHP. El motivo de la elección se centró sobre dos aspectos fundamentales: la posibilidad de diseñar una aplicación dinámica y disponible desde la Web, y la existencia de la librería para el diseño de gráficos matemáticos denominada JPGRAPH².

Un aspecto central en nuestro Proyecto y que definió cómo se iba a diseñar y desarrollar la aplicación es que estábamos pensando en un software educativo. Consideraremos software educativo a toda aplicación informática que se haya diseñado intencionalmente para impactar

¹ Free/Libre and Open Source Software: <http://flossproject.org/>

² Para más información consultar: <http://jpgraph.net/>

en el proceso de enseñanza y de aprendizaje, es decir, tomando la definición dada por [Marquès \(1996\)](#): “programas para ordenador creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje”.

Además de que sea un software educativo, como en cualquier actividad de aprendizaje, tuvimos que tener en cuenta los destinatarios. En nuestro caso, estudiantes universitarios de ciencias duras con una formación básica en Matemática. Todo ello, nos llevó a definir desde el aspecto de la aplicación (iconografía, tipografía y color), pasando por la terminología que debíamos utilizar para interactuar con el usuario, hasta qué datos les pediríamos y cómo deberían presentarse los resultados. Como afirma [Bruner \(1988\)](#), un estudiante construye el conocimiento en base a su maduración, experiencia física y social. Por lo tanto, para que este software educativo pudiera propiciar los aprendizajes en nuestros estudiantes, era fundamental definir un contexto acorde a los destinatarios.

Otro aspecto que debimos considerar es en qué teoría pedagógica se enmarcaría la aplicación. Teníamos claro que no queríamos una aplicación que sólo presentara una secuencia de contenidos y actividades donde el estudiante fuera recompensado o castigado en función de si las respuestas son correctas o no. Queríamos un ambiente donde el usuario debiera analizar qué datos ingresar (sobre una base teórica), pudiera realizar tantas pruebas como deseara y esto lo estimulara a sacar conclusiones que le permitieran comprender y aprender los diferentes métodos. De acuerdo a la opinión de [Bruner \(1988\)](#), algunas de las habilidades a adquirir son: la capacidad de identificar la información relevante para un problema dado, de interpretarla, de clasificarla en forma útil, de buscar relaciones entre la información nueva y la adquirida previamente.

Nuestra aplicación, según [Marquès \(1998\)](#), debería cumplir las siguientes características:

Características	Descripción
Facilidad de uso	En lo posible autoexplicativos y con sistemas de ayuda
Motivación	Mantener el interés de los estudiantes
Relevancia curricular	Relacionados con las necesidades del docente
Versatilidad	Adaptables al recurso informático disponible
Enfoque pedagógico	Que sea actual: constructivista o cognitivista
Orientación	Con control del contenido del aprendizaje
Evaluación	Inclusión de módulos de evaluación y seguimiento

Tabla 1: Características principales de los programas educativos

Cabe aclarar que, de las características expuestas en la [Tabla 1](#), la evaluación fue un aspecto educativo sobre el cual no hemos trabajado en nuestra aplicación. Por ello, no se desarrolló ningún módulo para cubrir este aspecto.

Finalmente, desde el punto de vista técnico, hubo dos aspectos importantes que se tuvieron en cuenta en el diseño del software. Por una parte, que buscábamos diseñar un software libre. Esto es, “el software libre es aquél que puede ser distribuido, modificado, copiado y usado; por lo tanto, debe venir acompañado del código fuente para hacer efectivas las libertades que lo caracterizan” ([Culebro Juárez et al., 2006](#)). Las libertades, según la Free Software Foundation³, permiten:

- ejecutar el programa sea cual sea nuestro propósito,

³ Sitio Oficial: <http://www.fsf.org/>

- estudiar el funcionamiento del programa y adaptarlo,
- redistribuir copias,
- mejorar el programa y luego publicarlo para el bien de toda la comunidad.

El otro aspecto a tener en cuenta desde lo técnico fue la usabilidad. Es decir, la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario en condiciones específicas de uso (Wikipedia, 2011). En este sentido, se tuvieron en cuenta las siguientes heurísticas adaptadas por Instone (1997 a,b) para aplicaciones Web y citadas por Baeza Yates y Rivera Loaiza (2002):

- Visibilidad del estado del sistema.
- Similitud entre el sistema y el mundo real.
- Control por parte del usuario y libertad.
- Consistencia y cumplimiento de estándares.
- Prevención de errores.
- Preferencia al reconocimiento frente a la memorización.
- Flexibilidad y eficiencia de uso.
- Estética y diseño minimalista.
- Ayuda para que el usuario reconozca, diagnostique y se recupere de los errores.
- Ayuda y documentación.

En la siguiente sección presentamos las diferentes etapas de desarrollo de la aplicación.

2 DESARROLLO

Los contenidos que se abordan en el software corresponden a las unidades II y IV, respectivamente, del programa analítico de Cálculo Numérico, a saber, *Resolución numérica de ecuaciones no lineales, Interpolación y aproximación polinomial y Ajuste de curvas*. Estos contenidos están precedidos por los de la unidad I, los cuales se refieren a *Errores y sistemas numéricos de la computadora*. Estos temas se desarrollan con cierto grado de profundidad, conceptualmente hablando, debido a que la falta de precisión es una de las mayores dificultades del Análisis Numérico, lo cual es expresado a los estudiantes a través de numerosos ejemplos dados en las clases teóricas y complementado con ejercicios en las clases prácticas. En el cronograma de actividades que debemos presentar al inicio del cuatrimestre ante la Dirección del Departamento de Matemática, se prevé, entre las clases teóricas y prácticas, que los temas de la unidad I lleven un mes para su desarrollo. Además, en las dos primeras semanas de cursado, aproximadamente, en las clases prácticas, se dan las ideas básicas (teórico-práctico) sobre el uso del software libre y gratuito Octave, totalmente compatible con MatLab. Usamos Octave simplemente por una cuestión de que no disponemos de suficientes licencias de MatLab. Adicionalmente, cabe señalar que, para cursar Cálculo Numérico, los estudiantes deben tener cursada o rendida (según cursen como regulares o por promoción sin examen final, respectivamente) la asignatura Programación, entre otras, lo cual nos da una idea sobre los conocimientos previos de los alumnos acerca del uso de la computadora y de algún lenguaje de programación.

Cuando pasamos ya a los contenidos de la unidad II, a los estudiantes se les ha dado en las clases teóricas las herramientas necesarias para poder analizar, según el problema a resolver, cuál de los métodos es conveniente utilizar (métodos abiertos, métodos que usan intervalos, separación de raíces, teoremas del punto fijo y sobre condiciones suficientes de convergencia, interpretación gráfica, orden de convergencia, raíces reales simples, múltiples, complejas, entre otros). Además de los conceptos teóricos, se dan una gran variedad de ejemplos para

estudiar las diferentes situaciones a las que pueden enfrentarse de acuerdo al problema a resolver. Estos contenidos son afianzados aun más en las clases prácticas a través de la resolución de ejercicios en el aula con calculadora, lápiz y papel. Luego sí, comenzamos con el trabajo en la sala de computación y con el uso del software.

Pero, ¿qué nos motivó a desarrollar el software educativo? Teniendo como base varios años de experiencia como docentes de esta asignatura, observábamos sistemáticamente que los estudiantes elaboraban sus propios programas para implementar los diferentes métodos que resolvían numéricamente ecuaciones no lineales. Estos programas ofrecían a los estudiantes información valiosa, evitando la realización de cálculos engorrosos, tediosos y a gran escala. Pero el inconveniente detectado era que sólo obtenían valores numéricos o alguna leyenda alusiva a la convergencia o no del método, relacionando cota de error con cantidad de iteraciones y sin más análisis. Es decir que no realizaban ningún otro estudio o exploración relativo a las características de las funciones, cómo eran sus gráficas, qué parámetros iniciales eran convenientes tomar, cuáles eran las razones por las que un método convergía o no. Esto hacía que, frecuentemente, los estudiantes realizaran los cálculos matemáticos aplicando las diferentes fórmulas sin efectuar un análisis detallado del comportamiento de cada método según la situación problemática abordada y los resultados obtenidos. Este análisis les afianzaría su comprensión, permitiendo un aprendizaje significativo en beneficio del uso de estas temáticas en aplicaciones futuras en diversas disciplinas. Con la elaboración de este software educativo pretendimos sumarle a los resultados numéricos obtenidos, la interpretación gráfica de los métodos aplicados a distintas situaciones problemáticas. Esto es, por medio de sucesivos gráficos se muestran, entonces, cómo las distintas iteraciones se van acercando o no a la aproximación de las soluciones buscadas y por qué un método es más conveniente que otro en determinadas situaciones.

Por otro lado, también quisimos que este software permita la interacción con el alumno, a efectos de que influya positivamente en su aprendizaje.

En líneas generales, podemos afirmar que la ventaja que presenta la elaboración e implementación del software educativo radica, principalmente, en la posibilidad de contar con una herramienta que se adapte exactamente a las necesidades del curso. Entre estas necesidades están las referidas a:

- La comprensión profunda de los contenidos conceptuales desarrollados.
- El logro de los objetivos generales del curso y de los objetivos específicos relativos a esta unidad temática.
- La posibilidad de mantener una estrecha relación entre teoría y práctica.
- La interacción de cada estudiante con el software a efectos de que pueda desarrollar las actividades propuestas siguiendo su propio ritmo de aprendizaje.
- La velocidad para el cálculo científico y obtención de resultados.
- La visualización de los procesos y resultados.
- La adquisición de habilidad y destreza en la lectura y procesamiento de información.

Creemos importante señalar que, para cada uno de los métodos que se incluyen en el software, existen pantallas de ayuda en las cuales se muestran las principales características de cada uno ellos (intervalo de análisis, valor o valores iniciales a ingresar, cota de error admisible, número de iteraciones máximas a realizar, fórmula a emplear, condiciones suficientes de convergencia, criterio de stop, entre otros).

De manera análoga, hemos procedido para trabajar con los temas de la Unidad IV referidos a *Interpolación y aproximación numérica* y *Ajuste de curvas*, que también están desarrollados en el software de elaboración propia, aunque se encuentran en una etapa de prueba a efectos

de realizar los reajustes necesarios, si correspondiere.

En una primera etapa de desarrollo del software educativo (primer prototipo) (Ascheri et al., 2008a,b), implementamos los *métodos iterativos para calcular raíces reales de ecuaciones no lineales*: bisección, iterativo de punto fijo, secante, regla falsi, Newton-Raphson y von Mises (ver Figura 1). Al acceder a cada opción, la aplicación le solicita al usuario que ingrese la función a graficar conjuntamente con otros parámetros iniciales. El software genera la gráfica correspondiente, la cual puede ser corregida, en lo que se refiere a su rango de visualización, cambiando algunos de los parámetros ingresados previamente. Luego, el usuario puede aplicar cualquiera de los métodos numéricos antes mencionados. Los resultados son mostrados en forma numérica y gráfica (ver Figura 2).

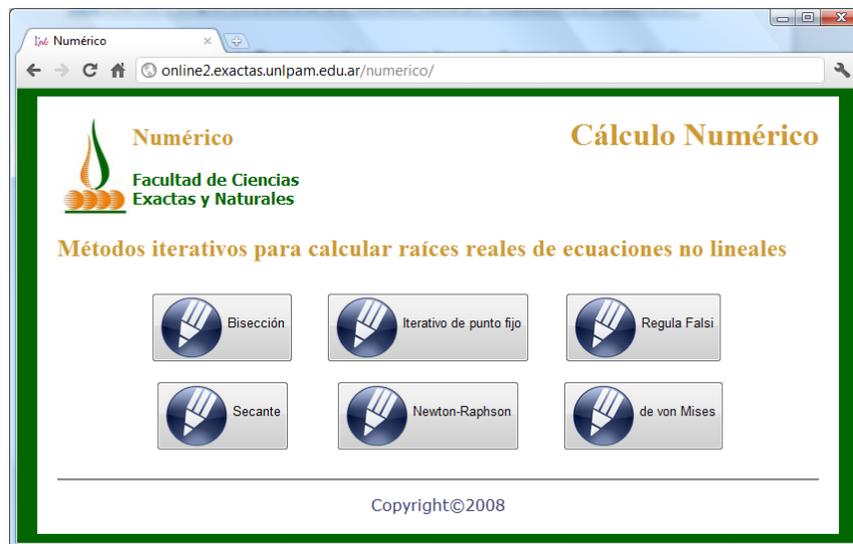


Figura 1: Captura de la pantalla inicial del primer prototipo

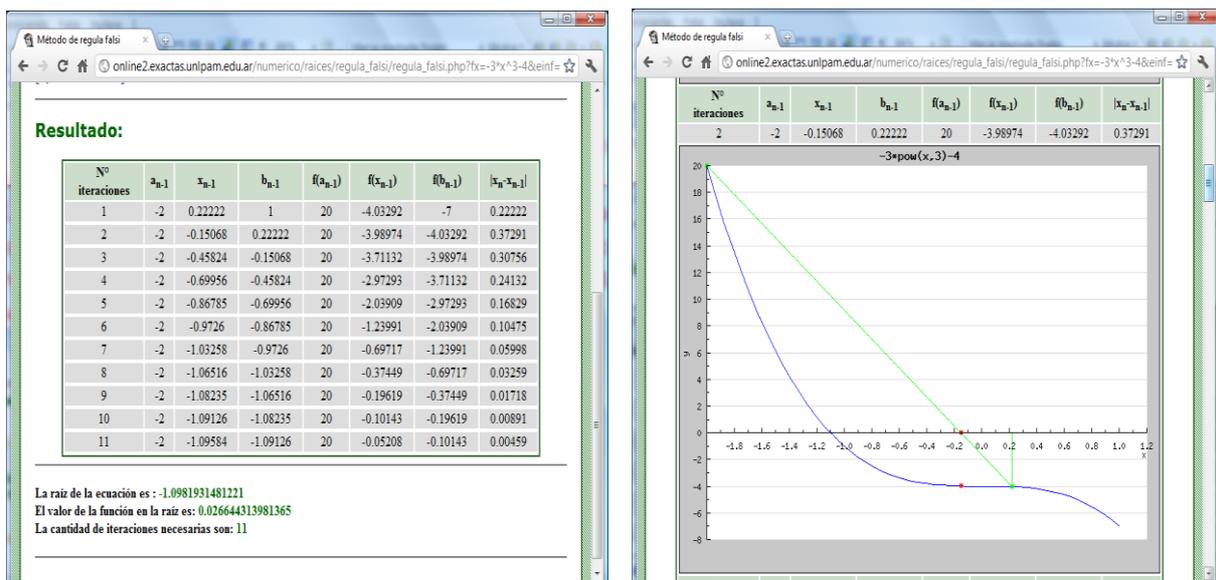


Figura 2: Presentación de los resultados al aplicar el método de la regla falsi

A finales del año 2008 y durante el 2009, trabajamos en la evaluación de la aplicación en lo que se refiere a su usabilidad. Para esta primera etapa de la evaluación, utilizamos una técnica

denominada caminata cognitiva. En ella, “un grupo de expertos simula la manera en cómo un usuario caminaría por la interfaz al enfrentarse a tareas particulares” (Baeza Yates y Rivera Loaiza, 2002). Esto nos permitió identificar un conjunto de fortalezas y debilidades en la aplicación (Ascheri et al., 2009 y 2010a). A los resultados de la evaluación de usabilidad (ver Tabla 2), le sumamos una encuesta anónima que realizamos a los estudiantes que cursaron Cálculo Numérico y usaron el software durante el segundo cuatrimestre del ciclo lectivo 2009 (ver Figuras 3 y 4).

Aspecto evaluado	Se cumple aceptablemente	Se cumple parcialmente	No se cumple
1. Visibilidad del estado del sistema		✓	
2. Similitud entre el sistema y el mundo real	✓		
3. Control por parte del usuario y libertad	✓		
4. Consistencia y cumplimiento de estándares	✓		
5. Prevención de errores	✓		
6. Preferencia al reconocimiento	✓		
7. Flexibilidad y eficiencia de uso		✓	
8. Estética y diseño minimalista		✓	
9. Ayuda para los errores	✓		
10. Ayuda y documentación			✓

Tabla 2: Resumen de la evaluación de usabilidad de la aplicación Web

Utilización del software

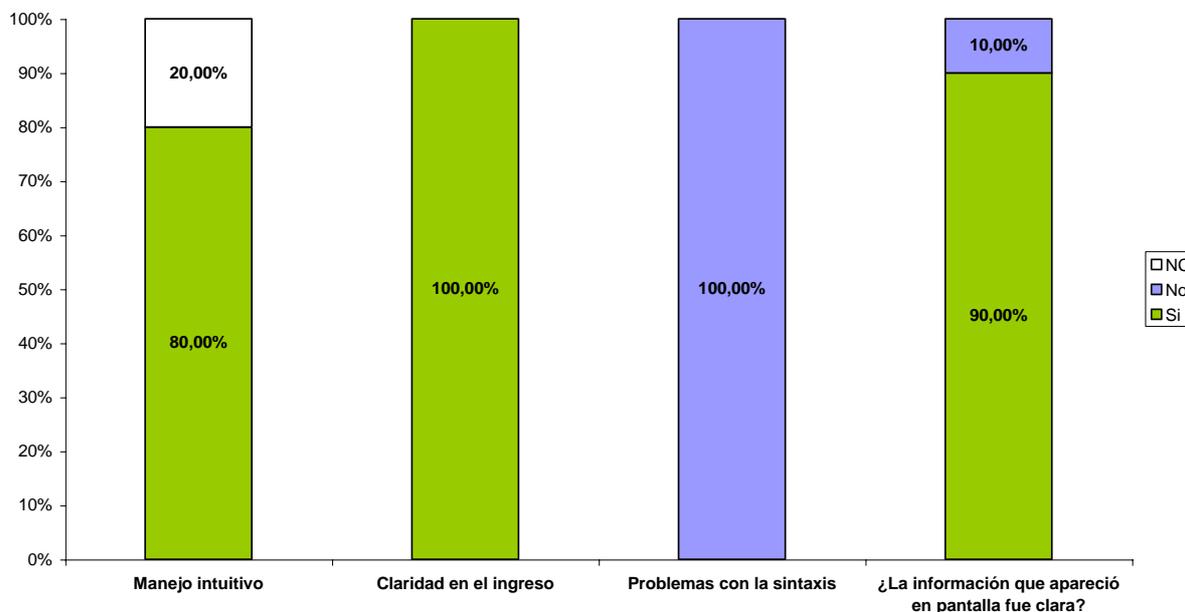


Figura 3: Presentación de los resultados al aplicar el método de la regla falsi

La aplicación favoreció los aprendizajes

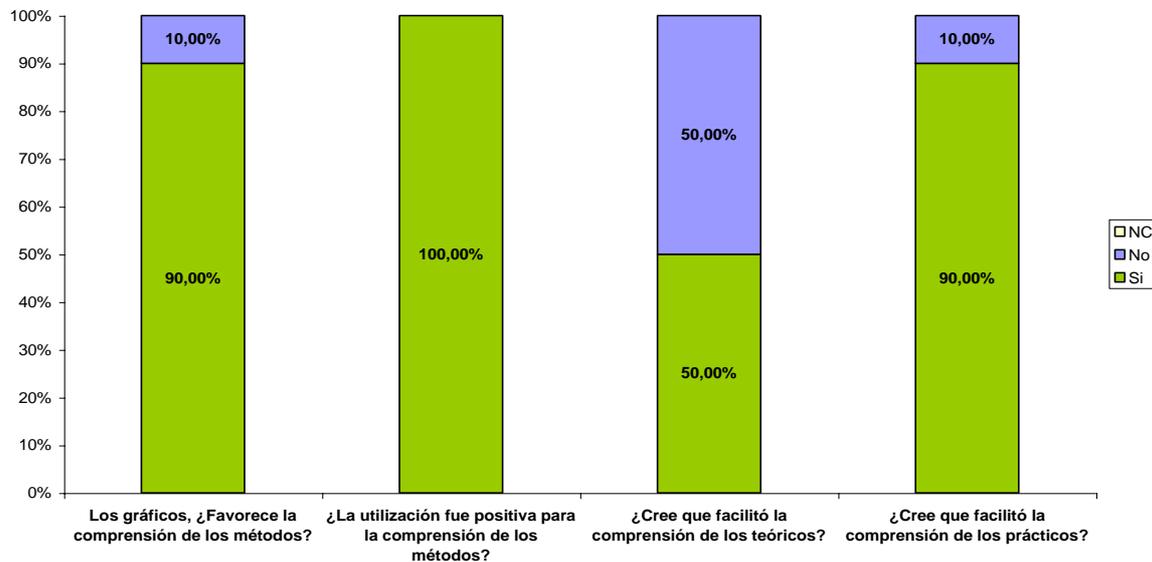


Figura 4: Presentación de los resultados al aplicar el método de la regla falsi

Si bien las respuestas dadas por los estudiantes nos mostraron varios aspectos positivos en lo que se refiere al uso del software educativo, también nos señalaron algunas cuestiones que debíamos ajustar para alcanzar los objetivos que nos habíamos propuesto a partir de la elaboración de este software.

Esta evaluación nos llevó a que, durante los años 2010 y 2011, trabajáramos en la implementación de un segundo prototipo. En éste, tratamos de mejorar los aspectos marcados como debilidades en la evaluación del software. Intentamos, entonces, mejorar las ayudas aportadas por el software y brindar una mayor cantidad de información relacionada con los datos ingresados por el usuario, por ejemplo, la sintaxis a utilizar cuando se ingresa la ecuación que se desea analizar. Además, desarrollamos la parte correspondiente a la implementación de los métodos numéricos referidos a los temas: *interpolación polinomial de Newton y de Lagrange y ajuste de curvas por el método de mínimos cuadrados* (Ascheri et al., 2010b, c; 2011).

En la Figura 5, mostramos algunas de las pantallas del software educativo en su estado actual⁴.



(a) Página inicial de la aplicación

⁴ Disponible en: <http://online2.exactas.unlpam.edu.ar/numerico/index.html>

polinomio de interpolación y el nuevo valor a interpolar. Obtenemos, para este caso particular, los resultados que se muestran en la [Figura 5 \(c\)](#).

Por último, si en la [Figura 5 \(a\)](#) elegimos la opción **Ajuste de curvas**, vemos los resultados numéricos y gráficos que arroja el software para esta situación particular, en la cual hemos elegido el ajuste exponencial para los pares de datos ingresados (ver [Figura 5 \(d\)](#)).

Durante el ciclo lectivo 2011, segundo cuatrimestre, pondremos a prueba la parte del software educativo referida a **Interpolación** y **Ajuste de curvas** para desarrollar los teóricos, prácticos y evaluaciones parciales correspondientes, además de trabajar, claro está, con el **Cálculo de raíces**. Creemos que esto nos llevará a tener que realizar nuevamente algunos ajustes y nos permitirá obtener nuevos resultados y conclusiones en lo que respecta al uso de este software educativo.

3 RESULTADOS Y DISCUSIÓN

Como integrantes de la cátedra de Cálculo Numérico y del grupo de investigación, podemos afirmar que el diseño de este software educativo nos ha permitido incorporar herramientas de programación y contar en la actualidad con una nueva herramienta didáctica para el desarrollo de algunos contenidos teóricos, prácticos y evaluaciones parciales. Además, esta aplicación puede ser usada por los estudiantes de Cálculo Numérico y por cualquier otro usuario interesado en estos temas, puesto que es una herramienta de acceso libre y disponible a través de la Web. Por ello es que podemos afirmar que estamos muy próximos al logro de los objetivos que nos habíamos propuesto alcanzar a partir del diseño de este software educativo.

Por otro lado, continuar con la elaboración del software ampliando los contenidos a desarrollar, incorporar su uso a otras asignaturas y analizar la metodología de trabajo, son seguramente líneas de investigación que nos permitirán aproximarnos al logro de los cambios en Educación que se pretenden conseguir. Además, creemos que es relevante continuar investigando cuál es el impacto del uso del software en el rendimiento académico de los estudiantes, por lo que incluiremos también este tema entre las futuras líneas de investigación a seguir.

4 CONCLUSIONES

Con respecto al uso del software educativo, podemos concluir que si bien ha sido positiva su implementación en el desarrollo de Cálculo Numérico como herramienta facilitadora para el proceso de enseñanza y de aprendizaje de los contenidos temáticos involucrados, aún la utilización de las computadoras no es considerada por los estudiantes como un proceso natural; siguen priorizando el uso de lápiz, papel y calculadora. Esto se debe, fundamentalmente, a que los estudiantes han transcurrido al menos 15 años en un sistema educativo en el cual se han mantenido tareas netamente tradicionales para realizar las correspondientes actividades.

Las políticas públicas diseñadas para la inclusión de TIC (por ejemplo, Programa Conectar Igualdad⁵) junto con la ubicuidad de Internet, en nuestra opinión, propiciarán la utilización de aplicaciones como la que estamos desarrollando. Adicionalmente, propuestas como las de “la nube”⁶, donde tanto la información como las aplicaciones están disponibles a través de

⁵ Más información en: <http://conectarigualdad.gob.ar/>

⁶ Más información en: http://es.wikipedia.org/wiki/Computación_en_nube

Internet, también pueden ayudar a aumentar la cantidad de usuarios que busquen aplicaciones disponibles en la Web para realizar sus trabajos y/o actividades académicas.

REFERENCIAS

- Ascheri, M. E. y Pizarro, R. A. Software para la enseñanza - aprendizaje de algunos métodos numéricos. VII Edumat. EMAT Editora. Buenos Aires. Argentina, 1-10, 2005.
- Ascheri, M. E. y Pizarro, R. A. Uso de tecnología en la enseñanza-aprendizaje de temas de Cálculo Numérico. ALME, México, 19: 879-885, 2006. Disponible en: <http://www.clame.org.mx>.
- Ascheri, M.E., Pizarro, R. A., Astudillo, G. J., García, P. y Culla, M. E. Relevamiento de software en línea para la enseñanza-aprendizaje de métodos numéricos. Herramientas para su desarrollo. V Congreso sobre Enseñanza de la Matemática Asistida por Computadora. Cartago, Costa Rica, 20-24, 2007. Disponible en: <http://www.cidse.itcr.ac.cr/ciemac/5toCIEMAC/Ponencias/Relevamientodesoftwareenlinea.pdf>
- Ascheri, M. E., Pizarro, R. A., García, P., Astudillo, G. J. y Culla, M. E. Desarrollo de un Software Educativo para la Visualización Gráfica del Comportamiento de Algunos Métodos Numéricos. X Workshop de Investigaciones en Ciencias de la Computación. La Pampa. Argentina, 790-794, 2008a.
- Ascheri, M. E., Pizarro, R. A., García, P., Astudillo, G. J. y Culla, M. E. Un software educativo con herramientas libres y acceso Web para temas de Cálculo Numérico: Primer prototipo. II REPEM. La Pampa. Argentina, 223-230, 2008b.
- Ascheri, M. E., Astudillo, G. J., García, P., Pizarro, R. A. y Culla, M. E. Software educativo para el tratamiento de algunos temas de Cálculo Numérico: Primera etapa de evaluación. XI Workshop de Investigaciones en Ciencias de la Computación. San Juan, Argentina, 670-673, 2009. Disponible en: http://www.wicc2009.com.ar/index.php?option=com_content&view=article&id=61&Itemid=74.
- Ascheri, M. E., Astudillo, G. J., García, P., Pizarro, R. A. y Culla, M. E. Elaboración de un software educativo usando herramientas gratuitas. Primeras evaluaciones. V TE&ET, RedUNCI. Santa Cruz. Argentina, 1-8, 2010a.
- Ascheri, M. E., Pizarro, R. A., Astudillo, G. J., García, P. y Culla, M. E. Resolución de ecuaciones no lineales e interpolación numérica usando software educativo elaborado con herramientas gratuitas. III REPEM. La Pampa. Argentina, 476-483, 2010b.
- Ascheri, M. E., Pizarro, R. A., Astudillo, G. J., García, P. y Culla, M. E. Diseño y elaboración de un software educativo para temas de Cálculo Numérico. Edutec. Bilbao. España, 1-10, 2010c.
- Ascheri, M. E., Pizarro, R. A., Astudillo, G. J., García, P. y Culla, M. E. Software educativo desarrollado para temas de Cálculo Numérico: Últimos avances. XVI EMCI Nacional y VIII EMCI Internacional. Olavarría. Argentina, 1-8, 2011.
- Baeza Yates, R. y Rivera Loaiza, C. Ubicuidad y Usabilidad en la Web, 2002. Recuperado en enero de 2009, de <http://www.dcc.uchile.cl/~rbaeza/inf/usabilidad.html>.
- Bruner, J. Desarrollo cognitivo y educación. Morata. Madrid, 1988.
- Culebro Juárez, M., Gómez Herrera, W. y Torres Sánchez, S. Software libre vs software propietario. Ventajas y desventajas. Creative Commons, México, 2006. Recuperado en marzo de 2011, de <http://www.softwarelibre.cl/drupal//files/32693.pdf>.
- Instone, K. Site Usability Evaluation – Part 1, 1997a. Recuperado en febrero de 2011, de <http://instone.org/siteeval>.
- Instone, K. Site Usability Heuristics for the Web – Part 2, 1997b. Recuperado en febrero de

2011, de <http://instone.org/siteeval>.

Marquès, P. El software educativo. Universidad Autónoma de Barcelona, 1996. Recuperado en enero de 2011, de http://www.lmi.ub.es/te/any96/marques_software.

Marquès, P. La evaluación de programas didácticos. Comunicación y Pedagogía. Barcelona, 149: 53-58, 1998.

Wikipedia. La enciclopedia libre. Usabilidad, 2011. Disponible en: <http://es.wikipedia.org/w/index.php?title=Usabilidad&oldid=47212718>.