

SOFTWARE EDUCATIVO PARA TEMAS DE CÁLCULO NUMÉRICO

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

Facultad de Ciencias Exactas y Naturales - Universidad Nacional de La Pampa

RESUMEN

Con este Proyecto, se investiga el impacto que producirá en el proceso de enseñanza aprendizaje de temas de Cálculo Numérico, la utilización de un software desarrollado para tal fin. Los temas abordados por este software son “Resolución Numérica de Ecuaciones no lineales” e “Interpolación y Aproximación Polinomial y Ajuste de Curvas por Mínimos Cuadrados”.

Se realizó un relevamiento sobre software educativos relacionados con los contenidos abordados, como así también de herramientas libres para el desarrollo Web que permitieran el diseño de este nuevo software. Luego de seleccionar las herramientas para el diseño de la aplicación, se elaboró un prototipo utilizado para la enseñanza y el aprendizaje del tema “Resolución Numérica de Ecuaciones no lineales”.

Actualmente, y luego de la utilización del software, se encuentra en desarrollo el capítulo correspondiente a los métodos de aproximación e interpolación. Se pretende, con el desarrollo de este software, disponer de un material didáctico que permita facilitar el proceso de enseñanza – aprendizaje de las temáticas involucradas en Cálculo Numérico. Este sistema en línea amigable con el usuario, ofrecerá un buen soporte al docente de esta asignatura y de otras vinculadas con los contenidos que aquí se abordan.

Palabras clave: *software educativo, cálculo numérico, software libre*

CONTEXTO

Nuestro Proyecto de Investigación, acreditado y financiado por la Facultad de Ciencias Exactas y Naturales de la Universidad Nacional de La Pampa, aborda líneas de investigación estrechamente vinculadas con varios de los Proyectos que se desarrollan en el Departamento de Matemática, de los cuales formamos parte. Se estudian, en el presente Proyecto, las herramientas libres para el desarrollo de software educativo, la elaboración de software con estas herramientas y la incidencia de este software en el proceso de enseñanza-aprendizaje. De esta forma, se vincula con otros Proyectos del Departamento relacionados a temas de ingeniería didáctica, inclusión de tecnologías en el proceso de enseñanza-aprendizaje y la importancia de la visualización gráfica en la enseñanza de la matemática. Por tal motivo, el desarrollo del presente Proyecto demanda una continua interacción con los investigadores de los demás Proyectos de la Institución, actuando en beneficio de sus avances.

1. INTRODUCCIÓN

Actualmente estamos desarrollando un software educativo que incluye el tema “Resolución numérica de ecuaciones no lineales” e incluirá “Interpolación y Aproximación Polinomial y Ajuste de Curvas por Mínimos Cuadrados” en el segundo cuatrimestre del actual ciclo lectivo. El presente Proyecto se desarrolla íntegramente con software libre.

Se utiliza PHP, HTML, CSS, la librería JGraph y GIMP para el diseño y edición de imágenes (en formato PNG). Varios de los algoritmos que implementan los diferentes métodos están desarrollados en Matlab o C++. Estos algoritmos están diseñados para obtener una respuesta numérica a través de la aplicación de diferentes métodos. Por medio de las actividades de este Proyecto, se implementan los algoritmos a PHP y además de la obtención de datos numéricos, se programa la implementación gráfica y la visualización del funcionamiento de los métodos.

En la primera etapa del Proyecto, navegamos la Web en busca de aplicaciones destinadas a la enseñanza y el aprendizaje de temas de Cálculo Numérico, particularmente, de aquellos que se abordarán en este software. Existen, en general, un gran número de sitios que ofrecen los algoritmos implementados en diversos lenguajes o applets que aplican los métodos para funciones fijas, o simplemente arrojan los resultados de la aplicación del método. Esto nos dio la pauta de que deberíamos desarrollar nuestro software educativo desde cero. El desafío no estaba únicamente en desarrollar una aplicación Web, sino que además debería ser un software educativo. Debíamos aclarar, entonces, qué entendemos por software educativo. Consideraremos software educativo a toda aplicación informática que se haya diseñado intencionalmente para impactar en el proceso de enseñanza-aprendizaje. Tomaremos la definición dada por Pere Marquès (1996, p.2): *“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”*. Teniendo esto como referencia, avanzamos en la selección de herramientas de libre acceso que nos permitieran hacer el desarrollo de 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). Estas libertades, según la Free Software Foundation, son:

1. la libertad para ejecutar el programa sea cual sea nuestro propósito,
2. la libertad para estudiar el funcionamiento del programa y adaptarlo,
3. la libertad para redistribuir copias,
4. la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad.

A finales del año 2008 y principios del 2009, se comenzó con la evaluación de la aplicación. Para esta primera etapa de la evaluación, se utilizó la técnica denominada *caminata cognitiva*. En esta técnica *“un grupo de expertos simula la manera en cómo un usuario caminaría por la interfaz al enfrentarse a tareas particulares.”* (Baeza Yates & Rivera Loaiza, 2002, p.8).

Esto nos permitió identificar un conjunto de fortalezas y debilidades en la aplicación desde los puntos de vista didáctico-matemático, del aprendizaje, del tratamiento del error y de la usabilidad. Desde el punto de vista didáctico-matemático, se analizó cuál es el encuadre en el que se encuentra la aplicación acorde con la enseñanza de la matemática. También, se utilizó la clasificación propuesta por Kemmis et al. (1977) para identificar la fundamentación educativa de la aplicación, y clasificarla dentro del paradigma instruccional, revelatorio o conjetural. El primero, paradigma instruccional, deriva la instrucción programada (Skinner), y se caracteriza por el refuerzo inmediato y el avance en pequeños pasos debido a la atomización de tareas complejas. Los programas son del estilo drill-and-practice. Se le presenta al alumno actividades con el objetivo de avanzar en una dirección determinada. Por cada paso (en esa dirección), el alumno recibe un refuerzo en términos de correcto/incorrecto (retroalimentación). *“El rol de la computadora es la presentación de los contenidos, la prescripción de la tarea y la motivación del estudiante a través de una rápida*

retroalimentación” (Kemmis et al., 1977, p.25). El segundo paradigma, el revelatorio, denominado así porque el conocimiento se le va revelando gradualmente al alumno. Se basa en las teorías de Bruner y Ausubel. Los programas del tipo simulación tienen la raíz en este paradigma. Las actividades que se proponen tienen la intención de reducir la brecha entre lo que sabe el alumno y el contenido (brecha cognitiva). Estas actividades deben estar diseñadas de manera que los alumnos accedan a los contenidos a partir del descubrimiento. “*La función de la computadora es la simulación o el manejo de información*” (Kemmis et al., 1977, p.26). El tercer paradigma, el conjetural, se basa en las teorías de Popper, Piaget y Paper. Propone la construcción del conocimiento a través de la experiencia. Las actividades deben propiciar el acceso al conocimiento a través de la generación y contratación de hipótesis. “*El rol de la computadora es manipular espacios, campos, lenguajes para la creación de modelos, programas, planes o estructuras conceptuales*” (Kemmis et al., 1977, p.27).

Entran en esta última clasificación los paquetes de modelización e inteligencia artificial. Para identificar el tipo de aprendizajes que estimula el software, se utilizó el análisis hecho por Squires y Mc Dougall (1994) de las relaciones entre los paradigmas antes mencionados y las habilidades cognitivas, también propuestas por Kemmis et al.

Para evaluar 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, 2009), utilizamos las heurísticas adaptadas por Instone (1997) y citadas por Baeza Yates et al. (2002) para evaluar la usabilidad:

- 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.

El análisis dio como resultado un conjunto de recomendaciones que nos permitieron realizar varias mejoras al software, que implican desde una mejor navegabilidad y opciones más claras para facilitar la resolución de los ejercicios y la visualización de los resultados, hasta una visión global de los métodos en la asignatura e información de sus autores.

Posteriormente, se implementó el software durante el año 2009 con los alumnos de Cálculo Numérico. Estos nos permitió desarrollar una segunda etapa de evaluación.

Como mencionamos anteriormente, los alumnos de Cálculo Numérico utilizaron el software para el desarrollo de los trabajos prácticos relativos a la unidad temática “*Resolución de Ecuaciones no Lineales*”. También lo usaron para resolver algunos de los ejercicios incluidos en el primer examen parcial, correspondientes a este tema.

Con el objetivo de rescatar la opinión de los alumnos sobre la utilización del software, se implementó una encuesta, disponible en <http://online2.exactas.unlpam.edu.ar/moodle/course/view.php?id=72>.

Esta encuesta fue respondida por los alumnos en forma anónima y se obtuvieron los siguientes resultados:

Pregunta	Respuestas		
	Si	No	NC
1. ¿Le pareció intuitivo el funcionamiento del software utilizado?	8	0	2
2. Las opciones en las que ingresa los datos para aplicar el método, ¿le parecieron claras?	10	0	0
3. Cada uno de los gráficos que representan las diferentes iteraciones, ¿favorecen la comprensión de los métodos?	9	1	0
4. ¿Tuvo inconvenientes con la sintaxis de las ecuaciones a solucionar?	0	10	0
5. ¿Utilizó las ayudas del software?	8	2	0
6. ¿Utilizó el software sólo en las PC de la Facultad?	6	4	0

7. ¿La utilización del software fue positiva para la comprensión de los métodos de resolución de ecuaciones no lineales?	10	0	0
12. ¿Ingresó a alguno de los links relacionados con los autores de los métodos que aparecen en el software	1	9	0
8. La información que apareció en pantalla fue:	Clara 9		Confusa 1
9. Cree que facilitó la comprensión de:	Prác. 9	Teor. 5	Ninguno 0

Además de las respuestas a las preguntas formuladas en la encuesta, los alumnos tuvieron la posibilidad de agregar los comentarios que creyeran necesarios. Así fue que 6 de los 10 alumnos participantes dejaron los siguientes aportes:

Complete con cualquier consideración que crea necesaria:

- *En lo personal me ayudó un montón. Por ejemplo, cuando tenía que indicar el intervalo donde se encontraba la raíz o raíces. Además, la representación gráfica me permitía tener una primera aproximación de donde estaba la raíz y así no darle un valor inicial que haga que el método diverja.*
- *Con respecto a la pregunta 5 sobre las ayudas, intenté usarlas pero me tiraba error la página, no se si se solucionó pues no he intentado nuevamente. Sobre la pregunta 8 que se refiera a la información en pantalla es confusa en un primer momento ya que tira muchos datos además de los gráficos que hacen difícil el análisis en una primera instancia, pero luego de ir familiarizándose con el software se hace mas comprensible y legible toda la información que se muestra.*
- *La ayuda a veces no anda.*
- *Me facilitó muchísimo el trabajo. Me ahorró tiempo en graficar funciones que me demandarían.*
- *mucho trabajo.*
- *El recurso presentado en la página me facilitó además de la comprensión de los*

métodos vistos, la corrección de las producciones propias hechas en octave.

- *No tengo otra consideración.*

Si bien las respuestas muestran la aceptación del software por parte de los alumnos, existen aspectos que se señalan y que deberán ser ajustados para los próximos pasos en los que el software se ampliará con la inclusión de nuevas unidades temáticas de Cálculo Numérico. Se está trabajando en tal sentido, tratando de mejorar las ayudas que brinda el software y ofrecer mayor cantidad de información relacionada con los datos ingresados por el usuario, por ejemplo, la sintaxis a utilizar al ingresar la ecuación que se desea analizar.

La información obtenida, además de permitir realizar los ajustes en el software ya elaborado, permitirá prevenir inconvenientes en el desarrollo de los temas relacionados con aproximación e interpolación, en los cuales nuevamente la visualización de cómo los diferentes métodos obtienen diferentes aproximaciones, es de gran importancia para la comprensión de su funcionamiento.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Las líneas de investigación y desarrollo que seguimos son:

- Búsqueda y análisis de herramientas libres que permitan desarrollar software educativo con acceso Web.
- Desarrollo de software educativo para la enseñanza-aprendizaje de algunos de los métodos de resolución de ecuaciones no lineales.
- Elaboración de una planificación adecuada que permita la eficiente inserción del software en el desarrollo de las clases, optimizando de esta forma los resultados esperados.

3. RESULTADOS OBTENIDOS / ESPERADOS

Se obtuvo una completa bibliografía sobre los temas que nos ocupan y sobre los software

existentes. Se han podido detectar las herramientas a utilizar para el desarrollo del software y conocer además, los software educativos existentes destinados a temas de Cálculo Numérico.

También, se ha desarrollado el software para la implementación de algunos de los métodos de resolución de ecuaciones no lineales, y se ha utilizado con los alumnos en el desarrollo de la unidad correspondiente. Se ha realizado un estudio de los resultados obtenidos en las evaluaciones parciales como así también, de las opiniones vertidas por los alumnos, usuarios del software. Con ello, se espera poder llegar hacer los reajustes necesarios a efectos de optimizar el diseño y posterior uso del software educativo que estamos desarrollando.

4. FORMACION DE RECURSOS HUMANOS

El presente Proyecto lo integran cinco docentes del Departamento de Matemática, Licenciados y Profesores de Matemática e Ingenieros en Sistemas. Algunos de los integrantes del equipo han finalizado recientemente su tesis de maestría relacionada con el Proyecto y otros se encuentran en la elaboración de la misma.

Se han presentado y publicado en Reuniones Científicas de carácter nacional e internacional, trabajos que han surgido de este Proyecto.

De esta forma, se han adquirido compromisos tendientes a realizar intercambios de las producciones que surjan de los Proyectos implicados, logrando así ampliar la población a la que estará dirigida la implementación del software educativo y que no se restrinja sólo a la cátedra de Cálculo Numérico, docentes y alumnos.

5. BIBLIOGRAFIA

Ausubel, D., Novak J., Hanesian H. (1997). *Psicología educativa. Un punto de vista cognitiva*. México. Trillas.

Baeza Yates, R. & Rivera Loaiza, C. (2002). *Ubicuidad y Usabilidad en la Web*. Consultado

en Febrero, 11, 2009 en <http://www.dcc.uchile.cl/~rbaeza/inf/usabilidad.html>.

Bruner, J. (1960). *The Process of Education*. Cambridge: Harvard University Press.

Culebro Juárez, M., Gómez Herrera, W. y Torres Sánchez, S. (2006). *Software libre vs software propietario. Ventajas y desventajas*. México: (CC) Creative Commons. Consultado en Marzo, 16, 2009 en: <http://www.softwarelibre.cl/drupal//files/32693.pdf>.

Eaton J W. (2002) GNU Octave Manual. A high – level Interactive Language for Numerical Computations, Publisher: Network Theory Ltd., Free License GNU General Public License.

Instone, K. (1997). *Site Usability Evaluation – Part. 1*. Consultado en Febrero, 12, 2009 en <http://instone.org/siteeval>.

Kemmis, S., Atkin R., y Wright E., *How do students learn?*1977, Centre for Applied Research in Education, University of East Anglia, UK:

Nakamura S. (1992) *Métodos Numéricos Aplicados con Software*, 1º Ed. Prentice may Hispanoamérica S. A. México

Marqués, P. (1996). *El software educativo*. Universidad Autónoma de Barcelona. Consultado en Enero, 26, 2009 en: http://www.lmi.ub.es/te/any96/marques_softwa re

Papert, S. (1987). *Desafío de la mente: Computadoras y educación*. Buenos Aires, Galápagos.

Piaget, J. (1985): *Psicología y Pedagogía*. Barcelona: Ariel.

Squires D. y Mc Dougall A. (1994). *Cómo elegir y utilizar software educativo*. Morata. Barcelona.

The PHP group. <http://www.php.net>