



PROPUESTA METODOLÓGICA PARA DESARROLLO DE MODELOS DE REDES NEURONALES ARTIFICIALES SUPERVISADAS

PROPOSED METHODOLOGY FOR DEVELOPING SUPERVISED ARTIFICIAL NEURAL NETWORKS MODELS

Sonia I. Mariño

Universidad Nacional del Nordeste- Argentina

simarinio@yahoo.com

Carlos R. Primorac

carlosprimorac@gmail.com

Argentina

RESUMEN.

En Ciencias de la Computación, la adopción disciplinada de métodos y técnicas permite la resolución de problemas. Se presenta una propuesta metodológica para la construcción de Redes Neuronales Artificiales supervisadas integrando a un ciclo de vida de la Ingeniería del Software las fases contempladas en el desarrollo de los mencionados modelos. Se validó en un dominio de la botánica a fin de ilustrar la resolución de problemas del mundo real. Particularmente, se trata desde una perspectiva de innovación educativa para estudiantes de Sistemas de Información, carrera con una fuerte orientación en temas de Ingeniería del Software, quienes como futuros profesionales se insertan en la academia y en las empresas.

PALABRAS CLAVES.

Metodologías, tecnologías inteligentes, Redes Neuronales Artificiales Supervisadas, Ingeniería del Software.

ABSTRACT.

In Computer Science, disciplined adoption of methods and techniques allows troubleshooting. The aim of this paper is present a methodology for modeling supervised Artificial Neural Networks. It is based on the integration of the development phases covered by mentioned models in a life cycle of software engineering. It was validated in a domain of botany in order to illustrate solving real-world problems. Specially, it is treated from the perspective of educational innovation oriented to Information Systems students, future professionals will be inserted in academia and businesses,

KEY WORDS.

Methodologies, intelligent technologies, Supervised Artificial Neural Networks, Software Engineering.





1. Introducción.

1.1 Fundamentos teórico-pedagógicos de la experiencia.

La carrera Licenciatura en Sistemas de Información (FaCENA – UNNE, Argentina) dispone de un plan de estudios enmarcado en los lineamientos curriculares de la Red UNCI (2006). Éste describe el perfil del graduado de la carrera Licenciatura en Sistemas de Información que tiene como competencia básica la adquisición de habilidades en el diseño y elaboración de proyectos científicos, tecnológicos y académicos, plasmados en productos o entregables (Mariño *et al.*, 2015).

En la Educación Superior se enfatiza el diseño de perfiles profesionales orientados al trabajo y producción de conocimiento. Se coincide con Gutiérrez Vargas (2002) en que las prácticas profesionales de este siglo “*han sido modificadas por el desarrollo de la ciencia, la tecnología y la informática*”.

El Aprendizaje Basado en Problemas (ABP) es una propuesta educativa innovadora, que centra el aprendizaje en el estudiante, promoviendo su construcción y su significación (Fernández & Duarte, 2013; Martí *et al.*, 2013). Según Minnard & Minnard (2013, p. 1077) “*es una estrategia de enseñanza aprendizaje en la que tanto la adquisición de conocimientos como el desarrollo de habilidades y actitudes resulta importante*”.

Villarreal & Esteley (2010) reconocen la importancia del trabajo con actividades de modelización matemática y cuya principal característica es aprender Matemática a partir de la generación de modelos que resuelvan problemas propuestos por alumnos o profesores.

La Inteligencia Artificial (en adelante IA) como una disciplina de la Ciencias de la Computación proporciona una diversidad de métodos, técnicas y herramientas para modelizar y resolver problemas simulando el proceder de los sujetos cognoscentes. Se sostiene la importancia de conocer y aprender técnicas de la Inteligencia Artificial a partir de su experimentación y aplicación en la resolución de problemas del mundo real.

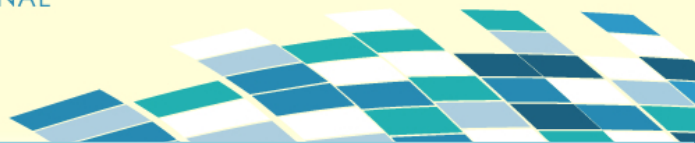
Siguiendo a la Red UNCI (2006), otra de las disciplinas de la Ciencias de la Computación es la Ingeniería del Software. Conciernen a la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, y el estudio integra Matemáticas, Ciencias de la Computación y prácticas cuyos orígenes se encuentran en la Ingeniería (ACM, 2006; Pressman, 2010; Swebok, 2004).

1.2 Descripción de la asignatura objeto de la experiencia.

Inteligencia Artificial (IA) es una asignatura optativa del plan de estudios, comprendida en el área de conocimiento identificada como Ciencias y Métodos Computacionales.

La asignatura objeto de la experiencia se caracteriza por el desarrollo de clases teórico-prácticas, clases prácticas en laboratorio, seminarios, enfatizando la lectura y análisis de publicaciones, estudio independiente, comunicación asincrónica, acceso a un entorno virtual.





Como se mencionó en Mariño & Vanderland (2013), en la asignatura se adopta un encuadre teórico-epistemológico fundado en el aprendizaje por construcción de significados, se emplean una diversidad de técnicas de motivación y activación a fin de asegurar el conocimiento de los principales paradigmas y técnicas de la IA y su aplicación en la resolución de problemas abstraídos del mundo real. La formación integral del alumno implica:

- i) Abordar los contenidos referentes a las principales técnicas de la inteligencia artificial.
- ii) Diseñar sistemas inteligentes aplicando las técnicas estudiadas.
- iii) Construir modelos representativos de situaciones reales y desarrollar las simulaciones empleando las técnicas correspondientes.
- iv) Desarrollar aplicaciones profesionales y científicas aplicando los conceptos estudiados, constituyendo éste uno de los aspectos de la formación académica.
- v) Elaborar informes técnicos, manuscritos y publicaciones referentes a los temas abordados.

En particular el último punto es un tema transversal y fundamental incorporado con la finalidad de orientar a los alumnos en la elaboración y mejora de sus producciones intelectuales, con repercusión en el momento de generar el trabajo final de graduación.

Los sistemas informáticos que incorporan técnicas de la IA pertenecen a una de las grandes categorías del software que presenta retos continuos para estos ingenieros (Pressmann, 2010; Sommerville & Sawyer, 2005). Aplican algoritmos para la resolución de problemas complejos imposibles de abordar por medio de un análisis directo. Las tecnologías inteligentes y sus aplicaciones incluyen la Robótica, los Sistemas Expertos y las Redes Neuronales Artificiales, entre otras.

Las Redes Neuronales Artificiales (en adelante RNA) pertenecen al paradigma conexionista de la IA (Castillo *et al.*, 1999; Freeman & Skapura, 1991; Klerfors & Huston, 1998; Martin del Brio & Sanz Molina, 2007; Russell & Norving, 2004; Rzepoluck, 1997). Están constituidas por nodos o unidades vinculadas mediante conexiones o enlaces sinápticos. Estos modelos no requieren conocer la expresión o la función a modelar, sólo se debe disponer de un conjunto de ejemplos satisfactorios (conjunto de aprendizaje) para que la RNA pueda aproximar esta función aplicando una regla de aprendizaje. A cada conexión se asigna un peso numérico. Los pesos constituyen el principal recurso de memoria de la red neuronal y el aprendizaje se realiza usualmente con la activación de estos pesos.

Para facilitar su comprensión, Wintermute (2002) compara las RNA con el estudio de los grafos en la matemática discreta. Como lo expresan Vazquez y otros (2008), las RNA son herramientas auxiliares cuando se desconoce la relación existente entre los factores y sus determinados, al suponer complejo y no lineal el dominio del problema.

En este trabajo se expone una metodología que integra en los fundamentos teóricos del ciclo de vida en cascada e iterativo de la Ingeniería del Software aquellos vinculados a la construcción de modelos de RNA supervisados. Su aplicación en aulas de Educación Superior permitirá producir innovaciones educativas.





Basado en Macchiarola y otros (2012), se enuncian como aspectos importantes los siguientes:

- i) “se trata de la introducción de un cambio o novedad”; en la forma de abordar la metodología para la generación de modelos de RNA;
- ii) “son procesos planificados o intencionales”; verificados con un caso experimental y que se implementarán y validarán en el desarrollo de las clases;
- iii) “se orientan a mejorar procesos y resultados de aprendizaje”, se adhiere a la construcción de conocimientos significativos basados en el aprender haciendo, es decir mediante la experimentación con abstracciones del mundo real; y
- iv) “suponen cambios en las prácticas y en los supuestos o concepciones que subyacen a las mismas”. Es decir, generalmente no se aborda la integración de metodologías y ciclo de vida de la Ingeniería de Software y aquellas metodologías para la simulación de problemas utilizando tecnologías de la Inteligencia Artificial.

Siguiendo lo expuesto en Dorfman & Thayer (1997), una metodología debe cumplir los siguientes requisitos

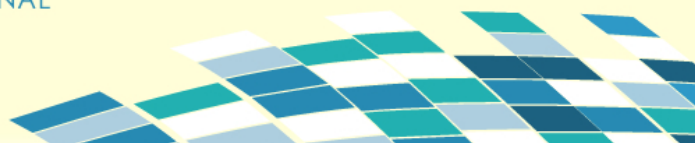
- i) está documentada: su procedimiento de uso está contenido en un documento o manual de usuario.
- ii) es repetible: su aplicación siempre es la misma.
- iii) es enseñable: los procedimientos descritos tienen un nivel suficientemente detallado y existen ejemplos para que personal cualificado pueda ser instruido en la misma.
- iv) está basada en técnicas probadas: implementa procedimientos fundamentales probados u otras metodologías más simples,
- v) ha sido validada: ha funcionado correctamente en un gran número de aplicaciones.
- vi) es apropiada al problema que se debe resolver.

El artículo se compone de cuatro secciones. La segunda sección describe la metodología propuesta para la generación de modelos de RNA. La tercera sección expone un caso de estudio, sintetizando los modelos de RNA entrenados y seleccionados aplicando la metodología elaborada y empleando una herramienta de software. Finalmente se enuncian algunas consideraciones finales y futuros trabajos.

2. Propuesta metodológica.

Se describe la metodología diseñada y aplicada en el diseño, desarrollo y entrenamiento de modelos de RNA supervisadas. Consiste en una propuesta integradora, dado que al modelo iterativo como ciclo de vida de la Ingeniería del Software se incorporan aquellas consideraciones particulares inherentes al diseño y desarrollo de modelos de RNA. El entrenamiento y los datos del estudio y de la muestra. Se trabajó en el año 2015.





A continuación se sintetizan las fases contempladas. La Figura 1 ilustra un ciclo de vida de cascada adoptado en cada caso de estudio para su modelización con RNA. En la fase denominada Desarrollo, se identifica un proceso iterativo, específicamente en 3.2d, 3.2e, 3.2f representadas en la Figura 2. Estas se repiten hasta localizar aquel modelo que constituya una mejor representación o solución al problema formulado.

A fin de ilustrar la metodología propuesta, se indican las fases y su adecuación para la aplicación de RNA en la identificación de especies botánicas:

Fase 1. Análisis. Consistió en:

1.1 Elección del modelo de RNA.

Se realizó un estudio referente al tratamiento de problemas con algoritmos supervisados y no supervisados. Se optó por el primero considerando el amplio rango de aplicaciones en que éstos fueron validados y en numerosos dominios del conocimiento.

1.2 Estudio del dominio de conocimiento.

La comprensión y formulación del problema depende de su entendimiento adecuado. La elección de variables evidenciales o variables de entrada se determina generalmente a partir de los datos suministrados.

A modo de ejemplificar, en este trabajo como fuente de información se utilizó un conjunto de datos, disponible en el repositorio UCI, identificado como "Iris Dataset". Contiene 3 clases de 50 instancias cada una. Cada instancia está compuesta por cuatro atributos numéricos predictivos y la clase correspondiente a una especie del género Iris.

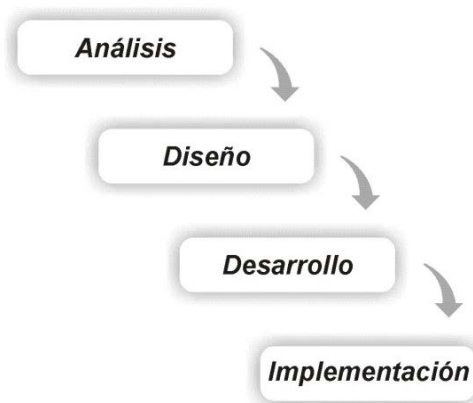


Figura 1. Fases del ciclo de vida adoptado para construir modelos de RNA



Figura 2. Proceso iterativo en la fase de desarrollo

1.3. Estudios estadísticos previos. Consistió en calcular estadísticos básicos como valores mínimos, máximo, media, desviación estándar y el coeficiente de correlación. A modo de ejemplo, la Tabla 1, presenta un resumen estadístico del conjunto de datos.

Tabla 1. Resumen estadístico.

<i>Medidas</i>	<i>Mín.</i>	<i>Máx.</i>	<i>Media</i>	<i>Desviación Estándar</i>	<i>Correlación</i>
<i>Longitud del Sépalo</i>	4.3	7.9	5.84	0.83	0.7826
<i>Ancho del Sépalo</i>	2.0	4.4	3.05	0.43	-0.4194
<i>Longitud del Pétalo</i>	1.0	6.9	3.76	1.76	0.9490
<i>Ancho del Pétalo</i>	0.1	2.5	1.20	0.76	0.9565

Fase 2. Diseño. Esta fase abordó el diseño de modelos de RNA. Se proponen los siguientes momentos:

2.1 Selección de variable objetivo.

Consistió en determinar la variable objetivo y sus posibles valores. En el dominio de conocimiento elegido se determinaron los valores de la variable objetivo, es decir, las especies consideradas Iris Setosa, Iris Virginica, Iris Versicolour.

2.2 Selección de las variables evidenciales.

Se eligieron como variables evidenciales los atributos que componen cada instancia o patrón. En el caso de estudio concierne a Longitud del sépalo (cm), Ancho del sépalo (cm), Longitud del pétalo (cm), Ancho del pétalo (cm).



2.3 Definición de nodos de entrada y nodos de salida.

Los nodos de entrada coinciden con las variables evidenciales. Los nodos de salida están representados por los valores que identifican las especies o valores esperados de éste nodo (Figura 3).

2.4 Diseño del patrón.

Consistió en determinar el patrón o registro que el modelo de RNA emplea en el proceso de aprendizaje en el dominio de conocimiento elegido y para su posterior testeo. Cada patrón o instancia (ejemplar o ejemplo) corresponde a un caso de entrenamiento o testeo. El algoritmo de aprendizaje y la herramienta de simulación determinan la conformación de cada patrón.

Para crear los conjuntos de datos de entrenamiento y comprobación se diseñó el patrón. Cada patrón consta de dos partes. La primera parte es, un conjunto de números que representan los valores de las variables de entrada o evidenciales, empleadas para estimar los resultados. Si hay n variables predictoras, los primeros n elementos de cada patrón, así como las primeras n columnas del archivo de datos de entrenamiento, contienen los valores "predictores". A la sección del "predictor" se agrega una segunda parte, la sección del "criterio" o "resultado", que consta de uno o más números, cada uno representa los valores de las variables de salida, es decir, los valores observados y que deberá predecir el modelo de RNA.

En el caso de estudio expuesto, el patrón consta de siete columnas, las cuatro primeras corresponden a las variables evidenciales y las tres últimas a los valores que asume la variable objetivo para cada una de las especies consideradas. En estas tres columnas, el valor uno significa que ese patrón corresponde a dicha especie, mientras que el valor cero en las restantes columnas implica que ese patrón no corresponde a las otras especies.

Fase 3. Desarrollo. Abordó los momentos referentes al desarrollo de modelos supervisados de RNA.

3.1. Selección de las herramientas informáticas.

Se concretó mediante la elección de un toolbook para modelizar RNA. Estas herramientas de software son de carácter generalista, motivo por el cual deben evaluarse a fin de verificar que las opciones disponibles son adecuadas para el tratamiento de un problema en particular. Se examinaron las aplicaciones para la creación y tratamiento de modelos de redes neuronales artificiales disponibles. Los toolbooks o herramientas existentes están destinadas a la predicción y clasificación. En la selección de las mismas se mencionan características como: la velocidad de ejecución, los requerimientos del sistema, las interfaces, el máximo número de variables y capas a especificar. La herramienta de modelización de RNA permite construir un modelo en base a los datos, posibilitando la especificación de diversos parámetros de aprendizaje. La conjunción de sus funcionalidades se emplea en la definición de distintas arquitecturas de modelos neuronales artificiales.





3.2 Construcción de modelos de RNA.

La construcción de modelos de RNA implicó una revisión de fundamentos conceptuales necesarios para la implementación experimental.

Se aplicaron conceptos relacionados con: división de los conjuntos de entrenamiento y testeo, la topología de una red neuronal, dimensionamiento de la red, arquitectura de una RNA, algoritmos de aprendizaje. Se seleccionó el método de aprendizaje supervisado de propagación hacia atrás, por considerarse el más adecuado al problema y el de mayor validez en amplios dominios del conocimiento. Se abordó:

a. Definición de la arquitectura. La arquitectura de una RNA se refiere a comunicación de las neuronas de una capa con las de la misma o de otra capa. Se optó por comunicaciones para conectar neuronas de diferentes capas o conexiones intercapas totalmente conectado. Es decir, cada neurona de una capa se conecta a cada neurona de la siguiente capa (Figura 3).

b. Definición de la topología. Consistió en la definición de la topología de la RNA. Se carece de un procedimiento específico para definir a priori el número de capas ocultas y el número de neuronas por capa necesarios para lograr el aprendizaje de la RNA. Se propusieron distintas configuraciones modificando el número de nodos y capas intermedias. Se establecieron iguales funciones de activación en todos los modelos propuestos: lineal en la capa de entrada y sigmoideal en capas intermedias y de salida (Figura 3).

c. Preparación de los datos. El formato del archivo para registrar los datos recolectados depende de la herramienta software de RNA seleccionada.

El conjunto de datos fue reordenado aleatoriamente, garantizando así que cada segmento sea representativo del total. De esta manera se reservaron cuatro segmentos para entrenamiento (80%) y el restante (20%) para validación del modelo.

La división de los conjuntos de entrenamiento y testeo permitió aplicar la técnica de validación cruzada, método estadístico de evaluación y comparación de algoritmos de aprendizaje. La forma básica se denomina validación cruzada de k-iteraciones o k-fold cross-validation (Payam et al., 2009).

d. Entrenamiento de los modelos. Consistió en la ejecución de los modelos. Se entrenaron y validaron distintos modelos de RNA construidos ad-hoc, cuyos resultados se exponen en la Tabla 1. Para todos los modelos se estableció el máximo número de épocas en 500. Como criterios de parada se empleó el máximo número de épocas y el límite en el RMS durante el entrenamiento.

e. Aplicación de la técnica de validación cruzada de k-iteraciones. Esta técnica consiste en dividir primero el conjunto de datos en k segmentos de igual tamaño (o cercanamente iguales). Luego se realizan k iteraciones de entrenamiento y validación. En cada una de estas iteraciones se reserva un segmento de los datos para validación mientras que el resto de los k-1 segmentos se utiliza para el entrenamiento del modelo. La validación cruzada puede aplicarse en tres contextos diferentes:

- i) evaluación de estimación,
- ii) selección del modelo y mejora de la precisión de los parámetros del modelo. La Figura 4 ilustra el procedimiento utilizado en este trabajo, siendo $k = 5$. Las secciones sombreadas de los datos son utilizadas para el entrenamiento mientras que la sección sin sombrar se utiliza para la validación. Se realizaron cinco iteraciones de entrenamiento y validación para cada uno de los modelos expuestos en la Tabla 2, reservando en cada iteración $k-1$ segmentos de datos para el entrenamiento (80%) del modelo y el segmento restante (20%) para su validación. De este modo se obtuvieron cinco resultados por cada muestra, la media entre los resultados de cada modelo es el valor del resultado final. Se empleó el valor del RMS en el conjunto de validación como medida de calidad para la selección del mejor modelo (Tabla 2).

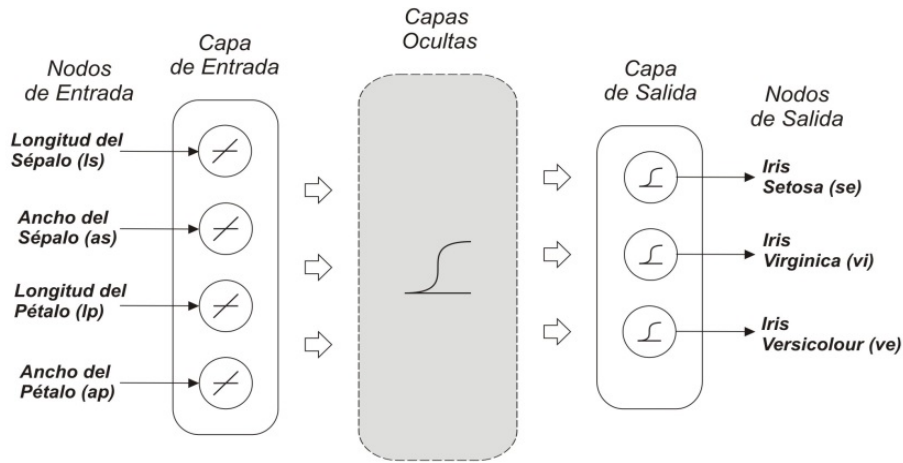


Figura 3. Topología del Modelo de RNA construido.

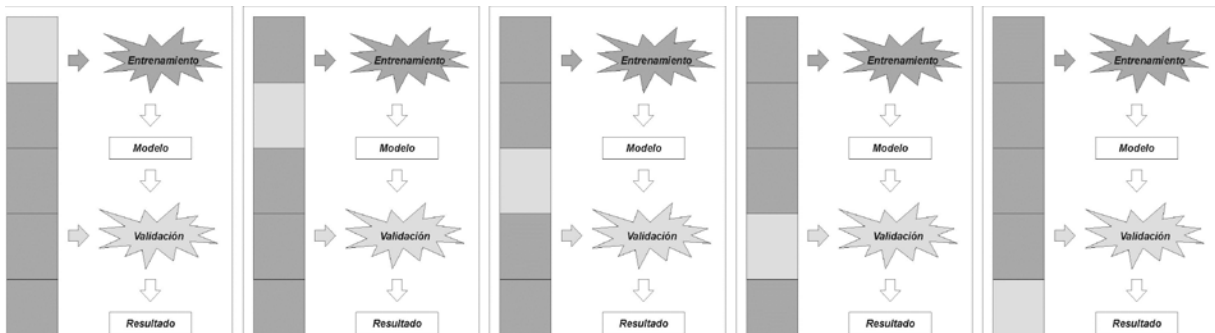


Figura 4. Procedimiento de validación cruzada con 5-iteraciones (Fuente: Payam et al., 2009).



f. Evaluación del rendimiento de los modelos. En el aprendizaje supervisado, una medida de calidad del modelo está dada en términos de los valores del error estándar como la suma de los cuadrados de los errores (Ecuación 1) y el error cuadrático medio (RMS) (Ecuación. 2) (Castillo *et al.*, 1999).

$$(1) SSQ = \sum_{p=1}^r \|b_p - \bar{b}_p\|^2$$

$$(2) RMS = \sqrt{\sum_{p=1}^r \|b_p - \bar{b}_p\|^2 / r}$$

Dónde: \bar{b}_p es la salida de la red para el vector de entrada ap , b_p es el valor de salida deseado para el vector de entrada ap y r es el número de residuales.

Fase 4. Implementación. Se abordaron los siguientes momentos:

4.1 Validación del modelo

Es el proceso por el cual se asegura que, en función de los objetivos planteados, el modelo responda a una descripción adecuada del fenómeno en estudio (Miser, 1993). Finalizado el desarrollo de los modelos, se deben implementar validaciones externas con los especialistas en la temática.

4.2 Transferencia del producto

El modelo de RNA validado por los especialistas puede ser objeto de implementación para la clasificación en ámbitos particulares del dominio de conocimiento.

3. Discusión.

En el siglo XXI, la Educación Superior se ha transformado reflejándose en un nuevo paradigma socio-cognitivo, aprendizaje permanente, integración adecuada de contenidos y métodos, desarrollo sistémico de estrategias cognitivas y metacognitivas, desarrollo de un razonamiento lógico, reorientación espacio-temporal y un nuevo estilo de comunicación y socialización (Perez Lindo *et al.*, 2005).

Por ello, se considera apropiada la adopción del Aprendizaje Basado en Problemas en espacios de Educación Superior, en este caso tratado en una disciplina como la Ciencia de la Computación.

El método que integra prácticas de la Ingeniería del Software para la construcción de perceptrones multicapas, ilustra el ABP en la asignatura Inteligencia Artificial, dado que: siguiendo a Solaz-Portolés *et al.* (2011) en “la finalidad de comprender y abordar el problema, los estudiantes han de identificar lo que necesitan aprender (aprendizaje autodirigido).” Asimismo, la resolución del problema utilizando el método propuesto favorecerá la integración de conocimientos provenientes de diversas áreas de la disciplina.

Además, a fines de su validación, se modelizó el reconocimiento de tres especies vegetales considerando cuatro caracteres como variables evidenciales. En el entrenamiento de los diversos modelos se estableció el número de neuronas por capas y las funciones de transferencia.





A continuación en la Tabla 2 se sintetizan los resultados obtenidos, donde: i) La Columna 1 expone la arquitectura, donde el primer y el último valor representan el número de unidades en las capas de entrada y salida respectivamente. Los valores intermedios representan el número de unidades en las capas ocultas. ii) En la columna identificada como Función de activación, indica aquella función empleada en cada una de las capas, donde. 3: lineal 1: sigmoideal [0,1]. iii) En la columna 3 se muestra el número de iteraciones de entrenamiento que cada modelo generó para el aprendizaje de la RNA. iv) en la columna 4 se indica la época en la cual converge el modelo. v) las columnas 5 a 6 muestran los valores de SSQ y RMS para el conjunto de entrenamiento en cada iteración. v) la columna 7 contiene los valores de RMS en el conjunto de validación en cada iteración. vi) la Columna 8 muestra el valor RMS promedio en los conjunto de validación, resultante de la aplicación de la técnica de validación cruzada de k-iteraciones.

La Figura 5 representa los RMS en los conjunto de entrenamiento y validación en cada una de las k-iteraciones. Un análisis de los resultados de los modelos de RNA expuestos en la Tabla 2, permitió seleccionar como más apto el identificado como modelo 4:4:3, considerando como criterio primordial debido a que arrojó el menor valor de RMS en el conjunto de validación.

Se observó además que el modelo seleccionado, presenta menor confusión entre las clases ve (Iris Versicolor) y vi (Iris Virginica). En el modelo 4:4:3 se detectó poca confusión, sin embargo en los modelos 4:5:3 y 4:5:4:3 tienden a confundir entre las dos clases mencionadas siendo el modelo con dos capas ocultas el de mayor error.

Cabe aclarar que las prácticas concernientes a la modelización y validación aplicadas a problemas del mundo real, se desarrollan en la asignatura a fin de asegurar el aprendizaje de métodos específicos y su correcta utilización.

Tabla 2. Síntesis de los experimentos

Arquitectura	Función de Transferencia	Iteración	Converge en época	SSQ _{train}	RMS _{train}	RMS _{test}	\overline{RMS}_{test}
4:5:1	3:1:1	1	92	2.240e1	2.495e-1	2.724e-1	1.511e-1
		2	96	3.189e-21	2.976e-12	1.491e-1	
		3	33	3.752e1	3.229e-1	3.344e-1	
		4	84	2.000e0	7.454e-2	5.441e-11	
		5	252	2.000e0	7.454e-2	3.056e-12	
4:5:4:3	3:1:1:1	1	26	4.064e1	3.352e-1	3.506e-1	3.401e-1
		2	28	3.997e1	3.332e-1	3.342e-1	
		3	19	4.092e1	3.372e-1	3.437e-1	
		4	15	4.902e1	3.609e-1	3.468e-1	
		5	55	4.016e1	3.340e-1	3.254e-1	
4:4:3	3:1:1	1	87	9.915e-21	5.248e-12	2.109e-1	1.500e-1
		2	184	1.999e1	2.356e-1	2.588e-1	
		3	138	5.591e0	1.246e-1	2.130e-1	
		4	36	3.625e0	1.003e-1	6.756e-2	
		5	351	7.652e-18	1.458e-10	5.097e-6	



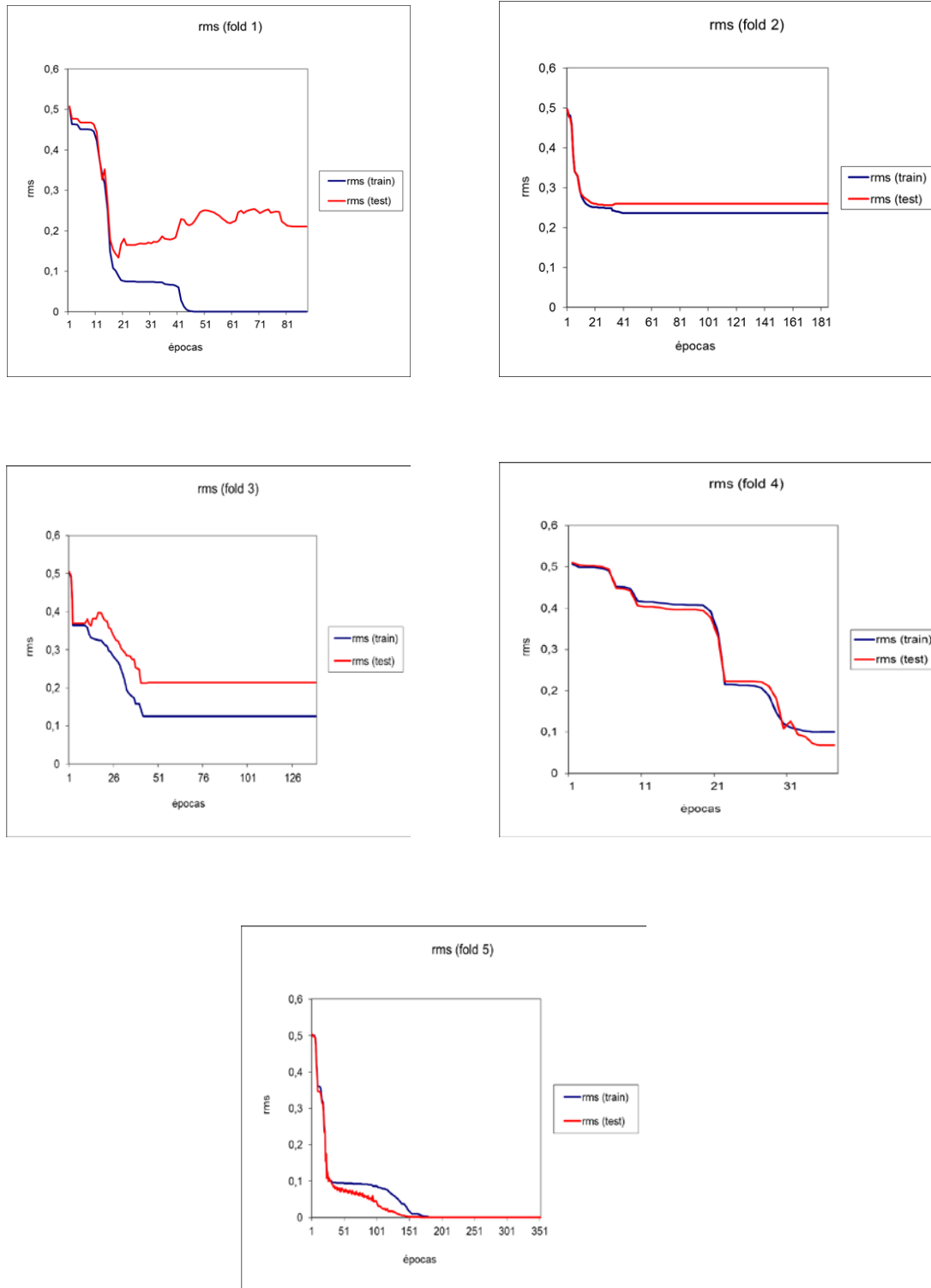


Figura 5. RMS en los conjuntos de entrenamiento y validación en cada una de las k-iteraciones del método de validación cruzada.





4. Consideraciones finales.

Principalmente en numerosos trabajos se expone un método para la construcción de modelos de RNA enfocado en aquellas fases concernientes a su modelización y verificación. En este documento, se mejoró una metodología –diseñada previamente- para la generación de modelos supervisados de RNA integrándola en un ciclo de vida de la Ingeniería del Software y abordando conceptos propios de la técnica inteligente.

También, centrado en el Aprendizaje Basado en Problemas, se ha expuesto una aproximación al tratamiento de modelos de Redes Neuronales Artificiales desde una perspectiva ingenieril, y su adopción en las clases de la asignatura optativa Inteligencia Artificial a fin de asegurar que el estudiante sea protagonista de su propio aprendizaje, y de ese modo consolidar el saber adquirido en la metodología elaborada, su integración en otras de la Ingeniería del Software y finalmente su uso en un intento de simular a partir del paradigma conexionista el proceder de expertos en un dominio de conocimiento.

Particularmente, dado el perfil de la carrera, la metodología expuesta se difundirá en las clases de la asignatura, para promover el uso de ciclos de vida de la Ingeniería del Software en la aplicación de tecnologías inteligentes.

Se sostiene que trabajos como el expuesto aporta a la formación integral de profesionales que pueden insertarse en ámbitos académicos y profesionales atendiendo a los requerimientos actuales de la sociedad del conocimiento.

Las futuras líneas de trabajo incluyen el estudio y empleo de herramientas más flexibles que además de la construcción y simulación de los modelos faciliten su integración en un sistema informático, para generar un sistema inteligente con interfaces centradas en los usuarios finales. Por lo expuesto, se continuará innovando desde aulas de Educación Superior vinculando distintas áreas de la Ciencia de la Computación.

Referencias bibliográficas.

- ACM (2006). *Computing Degrees & Careers*. ACM.
- Bourque P. & Dupuis, R, (ed.) (2004), SWEBOK. Guide to the Software Engineering Body of Knowledge - 2004 Version. IEEE Computer Society. Recuperado de <https://www.computer.org/web/swebok>
- Castillo, E., Cobo, A., Gutiérrez, J. M. & Pruneda, R. E. (1999). *Introducción a las redes funcionales con aplicaciones: Un nuevo paradigma funcional*. Ed. Paraninfo.
- Dorfman, M. & Thayer, R. H. (1997). *Software Engineering*, IEEE Computer Society Press.
- Fernández, F. H. & Duarte, J. E. (2013). El aprendizaje basado en problemas como estrategia para el desarrollo de competencias específicas en estudiantes de Ingeniería. *Formación Universitaria*. 6 (5), 29-38. Recuperado de <http://dx.doi.org/10.4067/S0718-50062013000500005>,
- Freeman, J. A. & Skapura, D. M. (1991). *Redes Neuronales: Algoritmos, aplicaciones y técnicas de programación*. Addison-Wesley Iberoamericana S.A. y Ed. Díaz de Santos S.A.





- Gutiérrez Vargas, M. E. (2002). El aprendizaje de la ciencia y de la información científica en la educación superior. *Anales de Documentación*, 5, 197-212.
- Klerfors, D. & Huston, T. L. (1998). *Artificial Neural Networks: An individual project*, within MISB-420-0. St. Louis. Saint Louis University. School of Business & Administration. Recuperado de <http://www.hj.se/~de96klda/NeuralNetworks.htm>,
- Mariño, S. I. & Vanderland, M. A. (2013), Estrategias de enseñanza– aprendizaje en la asignatura Inteligencia Artificial. Los trabajos prácticos entre los años 2003– 2010, *Documentación*, 6, 31, 44-52.
- Mariño, S. I.; Alfonso, P. L., Alderete, R. & Godoy, M. V. (2015), El proceso del diseño de proyectos de fin de carrera en la disciplina Informática. Una experiencia en la asignatura Trabajo Final de Aplicación, *International Journal of Educational Research and Innovation*, Número 4. Año II (Diciembre 2015).
- Martí, E., Poveda, F., Gurgu, T., Rocarías, J., Gil, D. & Hernández, A. (2013). Una propuesta de seguimiento, tutorías on line y evaluación en la metodología de Aprendizaje Basado en Proyectos. Recuperado de <http://refbase.cvc.uab.es/files/MPG2013.pdf>
- Martin Del Brio, B. & Sanz Molina, A. (2007). *Redes neuronales y sistemas borrosos*. Ed. Alfaomega Grupo Editor.
- Minnaard, C. & Minnaard, V. (2013), Aprendizaje Basado en Problemas (ABP) en los alumnos de carreras técnicas, *XV Workshop de Investigadores en Ciencias de la Computación*, Red de Universidades con Carreras en Informática (RedUNCI) 1077-1081
- Miser H. J. (1993) A foundational concept of science appropriate for validation in operational research. *European Journal of Operational Research* 66.
- Payam R., Lei T. & Huan L. *Cross Validation* [en línea]. Encyclopedia of Database Systems, Editors: M. Tamer Özsu and Ling Liu. Springer, 2009. Recuperado de <http://www.public.asu.edu/~ltang9/papers/ency-cross-validation.pdf>,
- Macchiarola, V. Martini, C., Montebelli. E. & Pizzolitto, A. L. (2012), El aprendizaje institucional en una universidad que innova. *Revista Iberoamericana de Educación* 58/4
- Pérez Lindo A., Ruiz Moreno, L., Varela, C., Grosso, F., Camós, C., Trottini, A. M., Burke, M. De L. & Darin, S. (2005). Gestión del conocimiento. Un nuevo enfoque aplicable a las organizaciones y la universidad. Grupo Editorial Norma. Bs. As.
- Pressman, R. S. (2010), *Ingeniería del Software: Un Enfoque Práctico*. Mc Graw Hill Interamericana.
- RED UNCI (2006), Red de Universidades con Carreras en Informática Lineamientos curriculares. Recuperado de redunci.info.unlp.edu.ar/
- Russell, S. J. & Norving, P. (2004), *Inteligencia Artificial: Un enfoque moderno*. Ed. Pearson Alambra,
- Rzepoluck, E. J. (1997) *Neural Networks Data Analysis Using Simulnet*. Ed. Springer.
- Solaz-Portolés, J. J. López, V. S. & Gómez López, A. (2011) Aprendizaje basado en problemas en la Educación Superior: una metodología necesaria en la formación del profesorado”. *Didáctica de las Ciencias Experimentales y Sociales*. 25, 177-186.
- Sommerville, I. & Sawyer, P. (2005) *Requirements Engineering: A good practice guide*. Ed. John Wiley.





- UCI. Machine Learning Repository. Recuperado de <http://archive.ics.uci.edu/ml/datasets/Iris>,
- Vazquez, J. C., Castillo, J. J., Rojas, M. Del C. & Marciszack, M. (2008), Redes Neuronales Artificiales aplicadas a Ciencias Sociales. En: *Anales del X Workshop de Investigadores en Ciencias de la Computación*. 129-135.
- Villarreal, M. & Esteley, C. (2010), *Modelización en la educación matemática: perspectivas, experiencias y desafíos*, III REPEM – Memorias Santa Rosa, Agosto 2010. Recuperado de <http://online2.exactas.unlpam.edu.ar/repem/cdrepem10/memorias/Talleres/t1.pdf>
- Wintermute. (2002). *Redes de neuronas artificiales y pensamiento*. Recuperado de <http://www.genaltruista.com>

