

Revista
Española de
Innovación,
Calidad e
Ingeniería del Software

Volumen 3, No. 1, abril, 2007

Web de la editorial: www.ati.es

E-mail: reicis@ati.es

ISSN: 1885-4486

Copyright © ATI, 2007

Ninguna parte de esta publicación puede ser reproducida, almacenada, o transmitida por ningún medio (incluyendo medios electrónicos, mecánicos, fotocopias, grabaciones o cualquier otra) para su uso o difusión públicos sin permiso previo escrito de la editorial. Uso privado autorizado sin restricciones.

Publicado por la Asociación de Técnicos en Informática

Revista Española de Innovación, Calidad e Ingeniería del Software (REICIS)

Editores

Dr. D. Luís Fernández Sanz

Departamento de Sistemas Informáticos, Universidad Europea de Madrid

Dr. D. Juan José Cuadrado-Gallego

Departamento de Ciencias de la Computación, Universidad de Alcalá

Miembros del Consejo Editorial

Dr. Dña. Idoia Alarcón

Depto. de Informática
Universidad Autónoma de Madrid

Dr. D. José Antonio Calvo-Manzano

Depto. de Leng y Sist. Inf. e Ing. Software
Universidad Politécnica de Madrid

Dra. Tanja Vos

Instituto Tecnológico de Informática
Universidad Politécnica de Valencia

D. Raynald Korchia

InQA.labs

D. Rafael Fernández Calvo

ATI

Dr. D. Oscar Pastor

Depto. de Sist. Informáticos y Computación
Universidad Politécnica de Valencia

Dra. Dña. María Moreno

Depto. de Informática
Universidad de Salamanca

Dra. D. Javier Aroba

Depto de Ing.El. de Sist. Inf. y Automática
Universidad de Huelva

D. Antonio Rodríguez

Telelogic

Dr. D. Pablo Javier Tuya

Depto. de Informática
Universidad de Oviedo

Dra. Dña. Antonia Mas

Depto. de Informática
Universitat de les Illes Balears

Dra. D. José Ramón Hilera

Depto. de Ciencias de la Computación
Universidad de Alcalá

Contenidos

REICIS

Editorial	4
<i>Luís Fernández Sanz, Juan J. Cuadrado-Gallego</i>	
Presentación	5
<i>Luis Fernández</i>	
Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software	6
<i>Miguel Garre, Juan José Cuadrado, Miguel A. Sicilia, Daniel Rodríguez y Ricardo Rejas</i>	
Aplicación de modelos de competencias a la gestión de sistemas de información	23
<i>Alfonso Urquiza</i>	

Editorial

The logo for REICIS, consisting of the word "REICIS" in a bold, white, serif font, centered within a solid black rectangular box.

En este número de abril de 2007 de REICIS volvemos a incorporar artículos remitidos directamente por los autores a la dirección de la revista. Las contribuciones han seguido el clásico proceso de aceptación que incluye la revisión anónima de miembros del comité editorial con la emisión de informes de comentarios y peticiones de adaptaciones y cambios si se estima de interés su publicación.

En breve queremos incorporar a partir del próximo número una breve sección de contribuciones invitadas aportadas por expertos de la industria y del mercado que nos transmitirán, en breves reflexiones, las tendencias o los temas de actualidad que perciban como más interesantes desde su posición privilegiada en el ámbito de la ingeniería y la calidad del software.

En cualquier caso, REICIS continua invitando desde estas líneas a todos los profesionales relacionados con el mundo de la Innovación, Calidad e Ingeniería del Software a que utilicen REICIS como el medio para dar a conocer sus trabajos e investigaciones teniendo las máximas garantías de la profesionalidad con que serán tratados sus trabajos. Podrán encontrar todas las instrucciones necesarias para el envío de sus contribuciones en la página web de la revista: www.ati.es/reicis.

Luis Fernández Sanz
Juan J. Cuadrado-Gallego
Editores

El primer número de REICIS del año 2007 publica, tras el proceso de revisión de nuestro comité editorial, dos contribuciones remitidas directamente a la revista.

En el primer artículo, el grupo de trabajo de la Universidad de Alcalá formado por Miguel Garre, Juan José Cuadrado, Miguel A. Sicilia, Daniel Rodríguez y Ricardo presentan un estudio comparativo de distintos algoritmos de clustering en la estimación de costes de desarrollo utilizando los datos del repositorio ISBSG. Se obtienen importantes conclusiones sobre su bondad de aceptación mediante el uso de los indicadores MMRE y PRED (0.3)

En la segunda contribución, Alfonso Urquiza reflexiona sobre “” analizando Félix García, Manuel Serrano y Mario Piattini nos presentan un conjunto de medidas para determinar el rendimiento y la capacidad de los procesos software con la base del estándar internacional ISO/IEC 15504-5:2006(E), tradicionalmente conocido como SPICE. La propuesta presentada se complementa con herramientas de ayuda para su implantación efectiva en las organizaciones implicadas en programas de mejora de sus procesos de software.

Luis Fernández Sanz

Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software

Miguel Garre, Juan José Cuadrado, Miguel A. Sicilia, Daniel Rodríguez, Ricardo Rejas

Dept. de Ciencias de la Computación

ETS Ingeniería Informática, Universidad de Alcalá

Ctra. Barcelona km 33.6 - 28871

Alcalá de Henares, Madrid

{miguel.garre,jjcg,msicilia,daniel.rodriguez}@uah.es

Resumen

Los modelos de estimación de coste software que obtienen una única relación matemática entre el esfuerzo y algún otro atributo característico de los proyectos software, proporcionan buenos resultados cuando la base de datos de proyectos, a partir de la que mediante métodos de regresión se obtiene la relación anteriormente mencionada, está formada por proyectos homogéneos. Sin embargo, para bases de datos de proyectos procedentes de muy diversas fuentes, tales como la base de proyectos de ISBSG formada por miles de proyectos heterogéneos, el utilizar una única relación matemática para representar a todos estos proyectos, no ofrece tan buenos resultados como si los proyectos fuesen homogéneos. En este trabajo se plantea, como mejora del proceso de estimación, segmentar la base de datos ISBSG en diferentes grupos de proyectos mediante la utilización de tres algoritmos de agrupamiento diferentes: COBWEB, EM, y k-means, de manera que para cada uno de estos grupos (formados por proyectos homogéneos entre sí) se obtenga una relación matemática diferente. La segmentación llevada a cabo por estos algoritmos mejora la estimación con respecto al modelo que utiliza la base de datos sin segmentar. Por otra parte si se comparan entre sí los resultados obtenidos al aplicar cada uno de ellos, se observa que el algoritmo que presenta un mejor comportamiento es EM debido a su naturaleza probabilista.

Abstract

The software cost estimation models that obtain an only mathematical relation between the effort and some other attribute characteristic of the software projects, provide good results when the projects data base, from which by means of regression methods the mentioned relation is obtained previously, is formed by homogenous projects. Nevertheless, for projects data bases coming from very diverse sources, such as the ISBSG projects data base formed by thousands of heterogeneous projects, using an only mathematical relation to represent all these projects, does not offer so good results as if the projects were homogenous. In this work one considers, like improvement of the estimation process, to segment the ISBSG data base in different groups from projects by means of the use of three different clustering algorithms: COBWEB, EM, and k-means, so that for each one of these clusters (formed by homogenous projects to each other) a different mathematical relation is obtained. The carried out segmentation by these algorithms improves the estimation with respect to the model that uses the data base without segmenting. On the other hand if the

results obtained when applying are compared to each other, it is observed that the algorithm whom a better behaviour presents is EM due to its probabilistic nature.

Palabras clave: algoritmo EM, estimación de coste software, clustering, segmentación, ISBSG.

1. Introducción

El desarrollo de un proyecto software, tiene asociado un esfuerzo de dedicación en cuanto a tiempo y dinero se refiere. El poder estimar estos valores, preferiblemente, en las primeras etapas de construcción de software (cuando esta información es más valiosa para el desarrollo del proyecto), ayuda en gran medida en la consecución del mismo. Muchos han sido los proyectos software que han fracasado por una mala planificación en el tiempo y coste. Es por ello por lo que existen multitud de métodos de estimación de coste software, para ayudar en el desarrollo de los mismos.

La multitud de métodos de estimación existentes han sido agrupados en diferentes clasificaciones, la primera se debe a Barry Boehm [6], en la que clasifica a los métodos de estimación de coste software en: modelos algorítmicos, juicio de expertos, analogía, parkinson, precio para ganar¹, de arriba hacia abajo² y de abajo hacia arriba³. A esta clasificación le sigue la de DeMarco [7], la de Conte [8]: modelos históricos-experimentales, modelos estadísticos, modelos teóricos, y modelos compuestos. Más tarde, en 1991 Kitchenham [9], los clasifica en opinión de expertos, analogía, descomposición, y ecuaciones de estimación, los diferentes métodos de estimación. Fairley [10] en 1992 ofrece esta otra clasificación: modelos empíricos, basados en regresión, y basados en la teoría. En 1997, Walkerden y Jeffery [11] ofrecen otra clasificación, modelos empíricos paramétricos, modelos empíricos noparamétricos, modelos basados en analogías, modelos teóricos y modelos heurísticos. Recientemente, Boehm [12] ofrece otra clasificación, actualización de la realizada en 1981, en la cual aparecen nuevas clasificaciones, las agrupa en: técnicas basadas en modelos, técnicas basadas en expertos, orientados al aprendizaje, modelos dinámicos, modelos basados en técnicas de regresión, y modelos compuestos bayesianos. Dentro de los métodos orientados al aprendizaje, Boehm menciona las redes

¹ *Price to win.*

² *Top-down.*

³ *Bottom-up.*

neuronales y el estudio de casos (originados a partir de los modelos por analogía). Por último, en el año 2001, Wieczorek [13] clasifica los métodos de estimación en, métodos basados en modelos y métodos no basados en modelos, dentro de los primeros tenemos genéricos (propietarios y no propietarios) y específicos (conducidos por los datos y compuestos).

Si se centra la atención en los métodos orientados al aprendizaje, según la clasificación de 2001 de Boehm, actualmente además de las redes neuronales y el razonamiento basado en casos (CBR) (Shepperd [16]), se están aplicando otras técnicas de aprendizaje automático⁴ para intentar mejorar el proceso de estimación. Tales técnicas son las de programación genética, sistemas difusos, inducción de reglas, CART (Árboles de Regresión y Clasificación⁵) y combinaciones de éstas. Ejemplos de ello, pueden ser los artículos de Ali Idri, en los que aplica sistemas difusos combinados con analogías [15, 14], o los de Dolado en los que utiliza programación genética [17, 18] para estimar el coste software. En el caso de inducción de reglas, C. Mair compara esta técnica con redes neuronales y razonamiento basado en casos [19]. L. Briand, ha realizado diferentes análisis y comparativas, en las que interviene CART, [21, 20]. Por último mencionar el artículo de A. Lee [22], en el que se utilizan técnicas de *clustering*⁶ para el entrenamiento de una red neuronal en la estimación de coste software.

El presente artículo es continuación de una serie de trabajos publicados anteriormente en los que se aplican técnicas de *clustering* para segmentar un conjunto de proyectos software sobre los que se aplican modelos matemáticos de estimación. Estos artículos son los de J. Cuadrado et al. [4, 5] y los de M. Garre et al. [1, 2, 3]. En estos trabajos se obtienen una serie de grupos de proyectos, a partir de la base de proyectos ISBSG versión 8 (*Internacional Software Benchmarking Standard Group*⁷), y sobre cada uno de ellos se aplica un modelo matemático multiplicativo de la forma $e = as^b$, donde e es el esfuerzo estimado y s alguna medida del tamaño del proyecto. Las constantes a y b se obtienen mediante análisis de regresión sobre los proyectos de cada grupo. Los resultados obtenidos respecto a MMRE (Magnitud media del error relativo⁸) y PRED(0,3) (Nivel de

⁴ *Machine learning.*

⁵ *Classification and Regresión Trees.*

⁶ Agrupamiento

⁷ <http://www.isbsg.org>

⁸ *Mean Magnitude of Relative Error.*

predicción⁹), mejoran los que se consiguen al utilizar una única ecuación para todos los proyectos. La medida MMRE indica el valor medio de los errores cometidos al realizar la estimación, en este caso del esfuerzo, para cada uno de los proyectos disponibles. Por otro lado $PRED(l)$, indicaría el porcentaje de estimaciones que cometen un error menor a l . Cuanto menor sea MMRE y mayor sea $PRED(l)$ mejor será el modelo obtenido, Conte [8] indica como valores aceptables para $MMRE \leq 0,25$, y $PRED(0,3) \geq 0,75$.

En los trabajos anteriormente citados [4, 5, 1, 2, 3] se ha utilizado EM como algoritmo para realizar el proceso de *clustering*, demostrando su bondad. En este trabajo se van a utilizar otros algoritmos diferentes a EM, con el fin de comparar los resultados que se obtienen con cada uno de ellos en el proceso final de estimación. Se compararán tres algoritmos, COBWEB, k-means¹⁰ y EM, ofreciéndose los resultados que se obtienen al aplicar cada uno de ellos en el proceso de agrupamiento. El resto del artículo se estructura de la siguiente forma: en la sección 2 se realizará una definición de la metodología de *clustering*, se mostrará una clasificación de los diferentes algoritmos de *clustering*, así como una descripción de los algoritmos COBWEB, EM y k-medias. En la sección 3 se aplican estos algoritmos a la base de datos ISBSG y se comparan los resultados ofrecidos por los mismos. Finalmente en la sección 4 aparecen las conclusiones obtenidas a partir de la comparativa realizada.

2. Clustering

El proceso de *clustering* consiste en la división de los datos en grupos de objetos similares. Para medir la similitud entre objetos se suelen utilizar diferentes formas de distancia: distancia euclídea, de Manhattan, de Mahalanobis, etc. El representar los datos por una serie de clusters, conlleva la pérdida de detalles, pero consigue la simplificación de los mismos. *Clustering* es una técnica más de Aprendizaje Automático, en la que el aprendizaje realizado es no supervisado. Desde un punto de vista práctico, el *clustering* juega un papel muy importante en aplicaciones de minería de datos, tales como exploración de datos científicos, recuperación de la información y minería de texto, aplicaciones sobre bases de datos espaciales (tales como GIS o datos procedentes de astronomía), aplicaciones web,

⁹ Prediction Level

¹⁰ K-medias

marketing, diagnóstico médico, análisis de ADN en biología computacional, y muchas otras. En el presente trabajo se utilizará como complemento a los modelos matemáticos en la estimación de coste software.

En los últimos años han surgido una gran variedad de algoritmos de *clustering*, a continuación se mostrarán más en detalle los que serán utilizados en este trabajo.

2.1 COBWEB

Se trata de un algoritmo de clustering jerárquico. COBWEB [23], se caracteriza porque utiliza aprendizaje incremental, esto es, realiza las agrupaciones instancia a instancia. Durante la ejecución del algoritmo se forma un árbol (árbol de clasificación) donde las hojas representan los segmentos y el nodo raíz engloba por completo el conjunto de datos de entrada. Al principio, el árbol consiste en un único nodo raíz. Las instancias se van añadiendo una a una y el árbol se va actualizando en cada paso. La actualización consiste en encontrar el mejor sitio donde incluir la nueva instancia, operación que puede necesitar de la reestructuración de todo el árbol (incluyendo la generación de un nuevo nodo anfitrión para la instancia y/o la fusión/partición de nodos existentes) o simplemente la inclusión de la instancia en un nodo que ya existía. La clave para saber cómo y dónde se debe actualizar el árbol la proporciona una medida denominada utilidad de categoría, que mide la calidad general de una partición de instancias en un segmento. La reestructuración que mayor utilidad de categoría proporcione es la que se adopta en ese paso. El algoritmo es muy sensible a otros dos parámetros:

- **Acuity**: este parámetro es muy necesario, ya que la utilidad de categoría se basa en una estimación de la media y la desviación estándar del valor de los atributos, pero cuando se estima la desviación estándar del valor de un atributo para un nodo en particular, el resultado es cero si dicho nodo sólo contiene una instancia. Así pues, el parámetro *acuity* representa la medida de error de un nodo con una sola instancia, es decir, establece la varianza mínima de un atributo.
- **Cut-off**: este valor se utiliza para evitar el crecimiento desmesurado del número de segmentos. Indica el grado de mejoría que se debe producir en la utilidad de categoría para que la instancia sea tenida en cuenta de manera individual. En otras palabras: cuando no es suficiente el incremento de la utilidad de categoría en el

momento en el que se añade un nuevo nodo, ese nodo se corta, conteniendo la instancia otro nodo ya existente.

Además COBWEB pertenece a los métodos de aprendizaje conceptual o basados en modelos. Esto significa que cada cluster se considera como un modelo que puede describirse intrínsecamente, más que un ente formado por una colección de puntos.

Al algoritmo COBWEB no hay que proporcionarle el número exacto de clusters que queremos, sino que en base a los parámetros anteriormente mencionados encuentra el número óptimo. La implementación utilizada en el presente trabajo, de este algoritmo, es la de Weka¹¹.

2.2 EM

EM pertenece a una familia de modelos que se conocen como *Finite Mixture Models*¹², los cuales se pueden utilizar para segmentar conjuntos de datos. Es un método de clustering probabilístico. Se trata de obtener la FDP (Función de Densidad de Probabilidad) desconocida a la que pertenecen el conjunto completo de datos. Esta FDP se puede aproximar mediante una combinación lineal de NC componentes, definidas a falta de una serie de parámetros $\{\theta\} = \cup\{\theta_j \forall j = 1..NC\}$, que son los que hay que averiguar,

$$P(x) = \sum_{j=1}^{NC} \pi_j p(x; \theta_j), \quad \sum_{j=1}^{NC} \pi_j = 1. \quad (1)$$

donde π_j son las probabilidades a priori de cada cluster cuya suma debe ser 1, que también forman parte de la solución buscada, P(x) denota la FDP arbitraria y p(x; θ_j) la función de densidad del componente j. Cada cluster se corresponde con las respectivas muestras de datos que pertenecen a cada una de las densidades que se mezclan. Se pueden estimar FDP de formas arbitrarias, utilizándose FDP normales n-dimensionales, t-Student, Bernoulli, Poisson, y log-normales. Aquí se modelarán los datos mediante distribuciones normales, por ser éstas las más comunes.

El ajuste de los parámetros del modelo requiere alguna medida de su bondad, es decir, cómo de bien encajan los datos sobre la distribución que los representa. Este valor de

¹¹ http://www.cs.waikato.ac.nz/_ml/weka/

¹² Modelos de mezcla finitos

bondad se conoce como el *likelihood*¹³ de los datos. Se trataría entonces de estimar los parámetros buscados θ , maximizando este *likelihood* (este criterio se conoce como *ML-Maximum Likelihood*). Normalmente, lo que se calcula es el logaritmo de este *likelihood*, conocido como *log-likelihood* ya que es más fácil de calcular de forma analítica. La solución obtenida es la misma, gracias a la propiedad de monotonidad del logaritmo. La forma de esta función *log-likelihood* es:

$$L(\theta, \pi) = \log \prod_{n=1}^{NI} P(x_n) \quad (2)$$

donde NI es el número de instancias, que se suponen independientes entre sí. El algoritmo EM, procede en dos pasos que se repiten de forma iterativa:

- **Expectation:** Utiliza los valores de los parámetros, iniciales o proporcionados por el paso *Maximization* de la iteración anterior, obteniendo diferentes formas de la FDP buscada.
- **Maximization:** Obtiene nuevos valores de los parámetros a partir de los datos proporcionados por el paso anterior.

Después de una serie de iteraciones, el algoritmo EM tiende a un máximo local de la función L. Finalmente se obtendrá un conjunto de clusters que agrupan el conjunto de proyectos original. Cada uno de estos cluster estará definido por los parámetros de una distribución normal.

La implementación utilizada no es la de Weka, ya que la implementación que lleva a cabo del algoritmo EM lleva asociada la premisa de la independencia de los atributos utilizados, cosa que en el caso del esfuerzo y de los puntos de función no se cumple. Por lo tanto se ha desarrollado una implementación de EM en lenguaje C que se adapta mucho mejor a las características del problema de estimación de coste software. Para una mayor descripción de la misma se puede consultar [3].

2.3 K-medias

Se trata de un algoritmo clasificado como Método de Particionado y Recolocación. El método de las k-medias [24, 25], es hasta ahora el más utilizado en aplicaciones científicas

¹³ Verosimilitud

e industriales. El nombre le viene porque representa cada uno de los clusters por la media (o media ponderada) de sus puntos, es decir, por su centroide. Este método únicamente se puede aplicar a atributos numéricos, y los *outliers*¹⁴ le pueden afectar muy negativamente. Sin embargo, la representación mediante centroides tiene la ventaja de que tiene un significado gráfico y estadístico inmediato. La suma de las discrepancias entre un punto y su centroide, expresado a través de la distancia apropiada, se usa como función objetivo. La función objetivo, suma de los cuadrados de los errores entre los puntos y sus centroides respectivos, es igual a la varianza total dentro del propio cluster. La suma de los cuadrados de los errores se puede racionalizar, como el negativo del *log-likelihood*, para modelos mixtos que utilicen distribuciones normales. Por lo tanto, el método de las k-medias se puede derivar a partir del marco probabilístico (ver subsección Clustering Probabilístico del libro de Mitchell [26]).

Existen dos versiones del método de las k-medias. La primera es parecida al algoritmo EM, y se basa en dos pasos iterativos: primero reasigna todos los puntos a sus centroides más cercanos, y en segundo lugar recalcula los centroides de los nuevos grupos creados en el anterior. El proceso continúa hasta alcanzar un criterio de parada (por ejemplo que no se realizan nuevas reasignaciones). Esta versión se conoce como algoritmo de Forgy [27]. La segunda versión [28] reasigna los puntos basándose en un análisis más detallado de los efectos causados sobre la función objetivo al mover un punto de su cluster a otro nuevo. Si el traslado es positivo, se realiza, en caso contrario se queda como está.

A diferencia de los anteriores algoritmos, COBWEB y EM, k-medias necesita la previa especificación del número de clusters que se desean obtener. La implementación utilizada en el presente trabajo, de este algoritmo, es también la ofrecida por Weka.

3. Caso de estudio

Para realizar el estudio de los diferentes algoritmos de clustering, primero es preciso realizar un filtrado de los proyectos de la base de datos ISBSG, que será utilizada. Esto se hará en la sección 3.1. Una vez preparados los datos, se aplican los algoritmos COBWEB, EM y k-medias, sobre los mismos, obteniéndose unos resultados que se ofrecen en la sección 3.2. La comparación de estos resultados se verá en la sección 3.3.

¹⁴ Valores desacordes con el resto

3.1 Preparación de los datos

Se utilizó la base de datos de proyectos ISBSG-8, la cual contiene información sobre 2028 proyectos heterogéneos, procedentes de diferentes organizaciones. Esta base de datos contiene información sobre tamaño, esfuerzo, y otras características de un proyecto, considerándose veintidós de estos atributos (los más importantes y representativos). El primer paso de limpieza consistió en eliminar de la base de datos todos los proyectos con valores numéricos no válidos o nulos en los campos esfuerzo (*'Summary Work Effort'* en ISBSG-8) y tamaño (*'Function Points'* en ISBSG-8). Además todos los proyectos cuyo valor para el atributo *'Recording Method'* fuese distinto de *Staff Hours* también fueron eliminados. La razón es que se considera que el resto de formas de considerar el esfuerzo son subjetivas. Por ejemplo *Productive Time* es una magnitud difícil de valorar en un contexto organizativo.

Otro aspecto a tener en cuenta para la limpieza de los datos es la forma en la que se obtuvieron los diferentes valores de los puntos de función. En concreto se examinó el valor del atributo *'Derived count approach'*, descartando todos los proyectos que no hubiesen utilizado como forma de estimar los puntos de función (sin ajustar) métodos como IFPUG, NESMA, Albretch o Dreger. Las diferencias entre los métodos IFPUG y NESMA tienen un impacto despreciable sobre los resultados de los valores de los puntos de función [29]. Las mediciones basadas en las técnicas Albretch no se eliminaron ya que, de hecho IFPUG es una revisión de estas técnicas. De la misma forma el método Dreger [30] es simplemente una guía sobre las mediciones IFPUG. Por último se procede a la eliminación de los proyectos con valores nulos para el atributo tiempo (*'Project Elapsed Time'*). Finalmente, tras el proceso de limpieza, el estudio se realiza sobre una base de datos de 1569 proyectos.

Para obtener más información sobre los atributos de la base de datos ISBSG se puede visitar su página web (www.isbsg.org) y buscar la descripción de los mismos.

3.2 Aplicación de los algoritmos de clustering

En esta sección se mostrarán los resultados que se obtienen al aplicar los tres algoritmos de clustering bajo estudio, sobre la base de datos ISBSG, una vez preparados los datos, como se ha visto en la sección 3.1.

3.2.1 COBWEB

Este algoritmo admite dos parámetros, como se ha visto en la sección 2.1, *cutoff* y *acuity*. Los valores que Weka propone para éstos es de *acuity*=1.0 y *cutoff*=0.0028. Con estos valores no se obtiene ningún cluster, por lo que es necesario modificarlos. Según el significado del *cutoff* parece lógico decrementar este valor, para obtener un mayor número de clusters, para ello se le asigna un valor de *cutoff*=0.00028, obteniéndose 1292 clusters.

Cada cluster está formado por unos pocos elementos, de manera que se tienen muchísimos clusters de muy pocos proyectos. El número tan elevado de clusters hace evidente que el valor elegido no es el más adecuado, es necesario incrementarlo. Tras varios intentos se utiliza un valor de *cutoff*=0.0018, obteniéndose 74 clusters.

Se hicieron nuevas pruebas, pero ninguna ofreció un número de clusters menor. De los clusters obtenidos, 73 cubren 90 proyectos repartidos uniformemente, mientras que el cluster restante engloba a 1479 proyectos.

En la figura 1 se puede ver el cluster que cubre a la mayoría de los proyectos. La escala de esta figura, así como las siguientes, es logarítmica, para poder apreciar mejor la forma de los segmentos obtenidos.

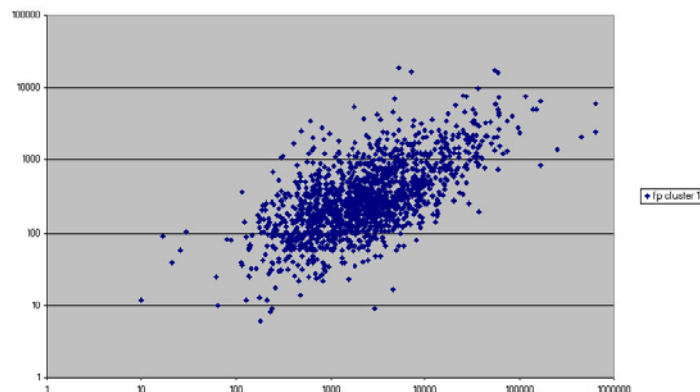


Figura 1: Cluster principal obtenido por el algoritmo COBWEB.

Los valores de MMRE y PRED(0,3) son los que se muestran en la tabla 1. Estos valores resultan de utilizar la recta de regresión del cluster obtenido, cuyos coeficientes *a* y *b* también se ofrecen en dicha tabla.

	Cluster 1
Nº Proyectos	1479
a	26,52
b	0,7899
MMRE	133,56%
PRED(0,3)	23,56%

Tabla 1. Valores MMRE y PRED(0,3) para COBWEB.

3.2.2 EM

Este algoritmo proporciona una segmentación de los proyectos en 9 clusters. La representación gráfica se puede ver en la figura 2.

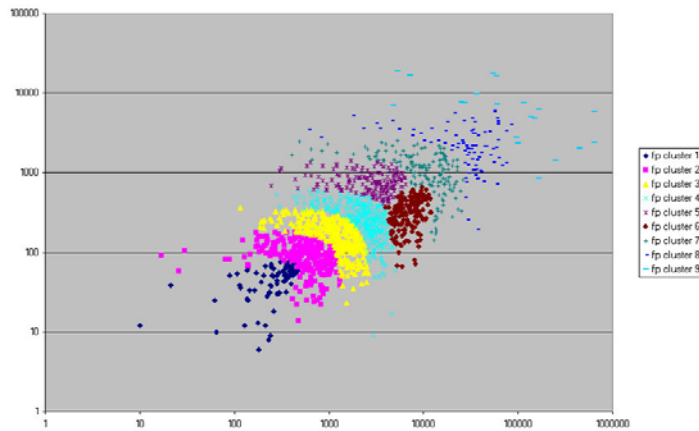


Figura 2: Clusters obtenidos por el algoritmo EM.

Los valores de los parámetros se ofrecen a continuación en la tabla 2. Los valores de MMRE y PRED(0,3) han resultado utilizando las respectivas rectas de regresión de cada cluster, cuyos coeficientes a y b también se ofrecen en dicha tabla.

Nº Cluster	1	2	3	4	5	6	7	8	9
Nº Proyectos	74	249	311	343	161	170	155	87	20
Probabilidad	0,046	0,1569	0,2655	0,2022	0,1198	0,1173	0,1027	0,0621	0,0133
Media e	284,68	616,15	0,179	2367,9	3072	6679,83	10984,97	31144,76	161796,42
Media fp	48,24	95,95	175,26	278,8	714,9	317,82	1123,3	2356,22	7509,93
Desv Stand e	128,96	309,55	594,94	1063,23	1590,45	2490,04	6226,94	18749,39	194440,38

Desv Stand fp	20,46	39,27	76,67	128,86	302,13	141,02	540,69	1383,26	5647
a	36,6	2829	34640	41730	44890	2500	672200	581400	-
b	0,5012	-0,3794	-0,675	-0,5351	-0,4322	0,1781	-0,6172	-0,4131	-
MMRE	64,88%	74,23%	41,54%	39,46%	71,47%	22,14%	80,42%	139,69%	-
PRED(0,3)	56,75%	46,37%	53,69%	59,18%	41,61%	69,41%	40%	41,37%	-

Tabla 2. Resultados del algoritmo EM.

Para el cluster 9, formado únicamente por 20 proyectos no se realiza ningún análisis por considerar que los resultados ofrecidos no serían fidedignos, por el pequeño número de elementos de este segmento.

3.2.3 K-medias

Al algoritmo de las k-medias hay que proporcionarle de antemano el número de clusters en los que se quiere segmentar la base de proyectos. Este dato se obtiene a partir del algoritmo EM, ya que éste obtiene este dato de forma óptima (véase [3]). Por lo tanto el algoritmo de k-medias se aplica sobre la base de datos de proyectos ISBSG para un número de clusters de 9 y un valor para la semilla de 10. La representación gráfica de los clusters obtenidos se puede ver en la figura 3.

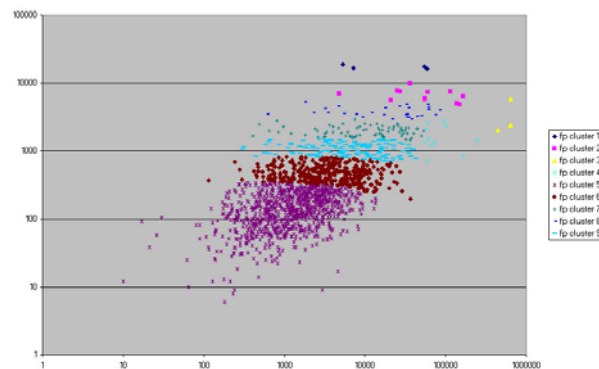


Figura 3: Clusters obtenidos por el algoritmo k-medias.

En este caso, el número de elementos por cluster es el siguiente: cluster 1, 4 proyectos; cluster 2, 12 proyectos; cluster 3, 3 proyectos; cluster 4, 11 proyectos; cluster 5, 920 proyectos; cluster 6, 361 proyectos; cluster 7, 74 proyectos; cluster 8, 28; y cluster 9, 156. Por lo tanto, solamente se analizarán los clusters 5, 6, 7, 8 y 9, los restantes se descartan por

tener un número de proyectos muy pequeño. Al igual que en el caso anterior, se pueden ver los resultados de aplicar este algoritmo, tras 50 iteraciones, en la tabla 3. Los valores de MMRE y PRED(0,3) resultan de aplicar rectas de regresión de cada cluster.

	Cluster 5	Cluster 6	Cluster 7	Cluster 8	Cluster 9
Nº Proyectos	920	361	74	28	156
Probabilidad	0,59	0,23	0,05	0,02	0,1
Media e	1811,23	5257,41	15869,70	28417,57	10664,88
Media fp	151,18	479,22	1980,13	3794,85	1069,98
Desv Stand e	1846,71	5066,92	13546,90	23373,81	9871,42
Desv Stand fp	81,10	137,29	324,03	647,19	212,73
a	7652	77390	730200	3699000	686300
b	0,5505	-0,5078	-0,5733	-0,6579	-0,6708
MMRE	109,68%	116,81%	193,5%	229,58%	155,92%
PRED(0,3)	24,34%	23,26%	13,51%	10,71%	14,74%

Tabla 3. Resultados del algoritmo k-medias.

3.3 Comparación de los resultados obtenidos

Si se comparan los valores de MMRE y PRED(0,3) de los clusters obtenidos por los algoritmos bajo estudio, se observa que el método EM destaca sobre los otros dos.

COBWEB ofrece un único cluster significativo, con resultados no muy diferentes de los que se obtendrían sin realizar segmentación alguna, considerando todos los proyectos para una única curva de regresión. De hecho el cluster obtenido es una representación de toda la base de datos, eliminando algunos *outliers*.

Este algoritmo no es adecuado para la segmentación de proyectos software, ya que tiende a agrupar a la mayoría en un solo segmento, cosa que no es deseable de cara a la mejora en la estimación de coste software.

EM y k-medias ofrecen segmentos de similares formas, aunque han agrupado los proyectos de diferente forma, ver figuras 2 y 3. Ello se debe al hecho de que ambos forman parte de una misma familia de algoritmos de clustering y tienen bases comunes. Sin

embargo EM destaca claramente sobre k-medias, ofreciendo mucho mejores valores de MMRE y PRED(0,3).

EM y k-medias, perteneciendo a la familia de algoritmos de particionado y recolocación, ofrecen mejores resultados que COBWEB, indicando que para tareas de segmentación de proyectos software son más adecuados que éste. Comparando EM y k-medias, se observa que los segmentos son diferentes, k-medias agrupa en un sólo cluster los mismos proyectos que EM divide en varios. Es decir, EM realiza una división más específica que k-medias.

4. Conclusiones

A continuación se mostrarán las conclusiones que se obtienen al aplicar cada uno de los algoritmos de *clustering* utilizados en este trabajo.

Este trabajo ha mostrado diferentes algoritmos de *clustering* que se han utilizado en la estimación de coste software. Para compararlos entre sí se han utilizado dos medidas ampliamente utilizadas y conocidas en la estimación, MMRE y PRED(0,3). Teniendo en cuenta estos datos, se observa que el algoritmo EM es el mejor algoritmo de los tres que se puede utilizar para la estimación de coste software, según el proceso descrito.

En general, siempre que se desee realizar una estimación de coste para un nuevo proyecto software, y se disponga de una base de datos heterogénea de proyectos históricos para ello, la mejor opción consistirá en utilizar el algoritmo EM tal y como se ha mostrado a lo largo del artículo. Los valores estimados utilizando EM son más parecidos a los reales, que si se utiliza COBWEB o k-medias para tal fin.

Se puede concluir que el algoritmo EM, siendo un algoritmo que realiza *clustering* probabilístico, es más adecuado que el algoritmo k-medias y COBWEB, para segmentar una base de datos de proyectos software, con el fin de mejorar la estimación del coste software.

Para próximos trabajos se profundizará en el estudio del algoritmo EM, utilizando otras formas de distribución de datos, así como la utilización de varios atributos con dependencia e independencia entre ellos. Así mismo se realizarán comparaciones entre EM con algoritmos de *clustering* de otras familias, para comprobar la bondad de los mismos.

Referencias

- [1] Garre, M., Cuadrado, J.J. y Sicilia, M.A., “Recursive segmentation of software projects for the estimation of development effort”, *Proceedings of the ADIS 2004 Workshop on Decision Support in Software Engineering*, CEUR Workshop proceedings, Vol. 120, 2004.
- [2] Garre, M., Charro, M., “Estimación del esfuerzo de un proyecto software utilizando el criterio MDL-EM y componentes normales N-Dimensionales. Aplicación a un caso práctico”, *Revista de Procesos y Métricas de las Tecnologías de la Información (RPM)* Vol. 2, nº 1, Marzo 2005, 2005, pp. 13-24.
- [3] Garre M., Cuadrado J.J., Sicilia, M.A., Charro M. y Rodríguez D., “Segmented Parametric Software Estimation Models: Using the EM algorithm with the ISBSG 8 database”, *Information Technology Interfaces*, Croacia, 20-23 junio 2005.
- [4] J. Cuadrado Gallego, Daniel Rodríguez, Miguel Ángel Sicilia. “Modelos Segmentados de estimación del esfuerzo de desarrollo del software: un caso de estudio con la base de datos ISBSG”. *Revista de Procesos y Métricas de las Tecnologías de la Información (RPM)*. Vol. 1, nº 2, agosto 2004, pp. 25-30.
- [5] Cuadrado-Gallego, J.J., Sicilia, M.A., Rodríguez, D. y Garre M., "[An empirical study of process-related attributes in segmented software cost-estimation relationships.](#)" *Journal of Systems and Software*, Vol. 3, nº 79, marzo 2006, pp. 351-361.
- [6] Boehm, B., *Software Engineering Economics*, Prentice-Hall, 1981.
- [7] DeMarco, T., *Controlling Software Projects*, Yourdan Press, 1982.
- [8] Conte, S.D., Dunsmore, H. E. y Shen, V.Y., “Software Engineering Metrics and Models”, Benjamin/Cummings, 1986.
- [9] Fenton, N.E., *Software metrics: a rigorous approach*, Chapman & Hall, Londres, 1991.
- [10] Fairley, R.E., “Recent advances in software estimation techniques”, *Proceedings of the 14th international conference on Software engineering*, Melbourne, Australia, 1992, pp. 382 – 391.
- [11] Walkerden, F. y Jeffery, D., “Software cost estimation: A review of models, process, and practice”, *Advances in Computers*, Vol. 44, 1997, pp. 59-125.
- [12] Boehm, B., Abts, C. y Chulani, S., “Software development cost estimation approaches -a survey”, *Annals of Software Engineering* 10, 2000, pp. 177-205.

- [13] Wieczorek, I. y Briand, L., *Resource estimation in software engineering*, Technical Report, International Software Engineering Research Network, 2001.
- [14] Idri, A., Abran, A. y Khoshgoftaar, T. M., *Fuzzy Case-Based Reasoning Models for Software Cost Estimation*, Springer-Verlag, 2004.
- [15] Idri, A., Abran, A. y Khoshgoftaar, T. M., “Fuzzy Analogy: A new Approach for Software Cost Estimation”, *Proceedings of the 11th International Workshop on Software Measurements*, Montreal, Canada, 2001, pp. 93-101.
- [16] Shepperd, M. y Schofield, C., “Estimating software project effort using analogies”, *IEEE Transactions on Software Engineering*, Vol. 23, nº 11, 1997, pp. 736-743.
- [17] Dolado, J.J., “On the problem of the software cost function”, *Information and Software Technology*, Vol. 43, 2001, pp. 61-72.
- [18] Dolado, J. J. y Fernández L., *Genetic Programming, Neural Networks and Linear Regression in Software Project Estimation*, INSPIRE III, Process Improvement through Training and Education, The British Computer Society, 1998, pp. 155-171.
- [19] Mair, C., Kadoda, G., Lefley, M., Keith, P. y Schofield, C. and Shepperd, M. and Webster, S., “An investigation of machine learning based prediction systems”, *The Journal of Systems and Software*, Vol. 53, 2000, pp. 23-29.
- [20] Briand, L., Langley, T. y Wieczorek, I., “Using the European Space Agency Data Set: A Replicated Assessment and Comparison of Common Software Cost Modeling”, *Proceedings of the 22th International Conference on Software Engineering*, 377-386, Limerick, Ireland, 2000.
- [21] Briand, L. C., El Emam, K., Maxwell, K., Surmann, D. y Wieczorek, I., “An Assessment and Comparison of Common Cost Software Project Estimation Methods”, *Proc. International Conference on Software Engineering*, ICSE 99, 1999, pp. 313-322.
- [22] Lee, A., Cheng, C. H. y Balakrishann, J., “Software development cost estimation: Integrating neural network with cluster analysis”, *Information & Management*, nº 34, 1998, pp. 1-9.
- [23] Fisher, D., “Knowledge acquisition via incremental conceptual clustering”, *Machine Learning*, Vol. 2, 1987, pp. 139-172.
- [24] Hartigan, J., *Clustering Algorithms*, John Wiley & Sons, New York, 1975.

- [25] Hartigan, J. y Wong, M., “Algorithm AS139: A k-means clustering algorithm”, *Applied Statistics*, Vol. 28, 1979, pp. 100-108.
- [26] Mitchell, T. M., *Machine Learning*. McGraw-Hill, 1997.
- [27] Forgy, E., “Cluster analysis of multivariate data: Efficiency versus interpretability of classification”, *Biometrics*, Vol. 21, 1965, pp. 768-780.
- [28] Duda, R. y Hart, P., *Pattern Classification and Scene Analysis*, Wiley & Sons, 1973.
- [29] NESMA, *NESMA FPA Counting Practices Manual (CPM 2.0)*, 1996
- [30] Dreger, J. Brian. *Function Point Analysis*. Prentice Hall, 1989.

Aplicación de modelos de competencias a la gestión de sistemas de información

Alfonso Urquiza

Universidad Francisco de Vitoria

a.urquiza.prof@ufv.es

Abstract

Today's development of the new economy shows that the expansion and wide usage of Information Systems and Technology has become a key driver in business performance improvements. Increasing relevance of the Chief Information Officer (CIO) and other lower level IS management roles becomes thus crucial in this scenario, particularly in large size technology driven organizations. Moreover, within the domain of Human Resources Management, competency based approaches boosting transformation of classical processes for employee life cycle management. Unfortunately most competency based implementations become long, high cost and manually supported adventures mainly due to absence of adequate competency frameworks which are effective for evaluating performance of key managerial employees, such as CIOs. This article addresses the challenge of creating a Competency Framework which includes a comprehensive competency model definition and an efficient method to evaluate IT management employees performance in large size organizations.

Keywords: skills, information system management, performance management, human assets, human resources management, CIO.

Resumen

En el ámbito de la nueva economía, el uso adecuado de los Sistemas de Información (SI) y las nuevas tecnologías se ha convertido en uno de los aspectos diferenciales clave en la consecución de resultados empresariales. La figura profesional y el rendimiento del Gestor de Sistemas de Información y Tecnología (*CIO*, *Chief Information Officer* en terminología inglesa) y de gestores en niveles inferiores, adquieren una relevancia cada vez más significativa, particularmente en las grandes organizaciones. Por otra parte la aplicación del paradigma competencial consigue niveles de aceptación significativos, transformando los procesos clásicos de gestión del 'ciclo de vida' de los empleados en las empresas. Pero su implantación es difícil, costosa y soportada todavía en gran medida en procesos manuales. Además, todavía no existen marcos de competencias específicos, para gestionar de forma eficiente el rendimiento de gestores en SI. Este trabajo propone un enfoque de construcción del marco para obtener posteriormente una definición de competencias y niveles y un método para evaluar el rendimiento laboral de estos profesionales de modo que pueda aplicarse universalmente en las grandes organizaciones..

Palabras clave: competencias, gestión de sistemas de información, gestión del rendimiento, capital humano, gestión de recursos humanos, CIO.

1. Introducción

Algunas investigaciones sobre la previsible evolución de la actividad informática en las empresas, publicadas en los años previos a la transformación que Internet produce en la sociedad, predecían que las funciones de gestión de los Sistemas de Información (SI) tanto para niveles intermedios como para el de mayor relevancia (*CIO, Chief Information Officer* en terminología inglesa) acabarían por descentralizarse y, en muchos casos, serían externalizadas [1]. Después de transcurridos los primeros años de adaptación social y empresarial al nuevo escenario de crecimiento y de consolidación de la Sociedad de la Información, el efecto de transformación observado más relevante en las organizaciones, es el *E-Business*. Éste se entiende como cualquier iniciativa (táctica o estratégica) que transforma las relaciones de negocio, bien entre empresas y consumidores (*B2C*), entre empresas (*B2B*) o incluso entre consumidores (*C2C*), soportada sobre las nuevas tecnologías vinculadas a Internet [2].

En este escenario, y lejos de las predicciones anteriores, nos encontramos ahora que la influencia de los gestores de SI crece considerablemente, particularmente en las grandes organizaciones [3]. Además se requiere de ellos nuevas habilidades empresariales y organizativas, conocimientos de gestión del negocio y capacidad de adaptación a nuevas arquitecturas y tecnologías.

Por otra parte, una de las sorpresas más interesantes que ha producido la transformación E-business en la actividad empresarial es que el uso de la tecnología, en vez de reemplazar a las personas, tal como algunos predecían que podría ocurrir, confirma que el talento constituye realmente el activo más importante en cualquier organización. Algunos incluso afirman que nos encontramos en una nueva revolución del conocimiento [4].

La gestión de este talento, que los profesionales de Recursos Humanos (RRHH) identifican como capital humano, se convierte en una necesidad creciente, sobre todo en aquellas organizaciones que se conocen como intensivas en conocimiento y en las que los profesionales y gestores de los SI son protagonistas esenciales. Este hecho, introduce cambios muy significativos en la forma clásica (transaccional) de gestionar los RRHH. En

el nuevo escenario el énfasis se aplica a la gestión de activos de conocimiento o competencias, entendidas como “conjunto de conocimientos, habilidades y actitudes personales requeridas en los empleados para realizar su trabajo de forma eficiente” [5]. Al final, siempre se busca mejorar los resultados por lo que el rendimiento de los empleados clave condiciona los resultados, tal como demuestra [6] al analizar la posible relación entre capacidades y competencias en tecnologías de la información, utilizando para ello datos obtenidos en muestras muy amplias de compañías pertenecientes a distintos sectores. Los resultados obtenidos confirman que las empresas que desarrollan competencias relevantes en tecnologías de información tienden a obtener resultados muy superiores frente a aquellas que sólo gestionan su actividad apoyándose en modelos clásicos de rendimiento tipo coste-beneficios.

En el ámbito laboral de las grandes organizaciones ya son habituales las prácticas de RRHH que utilizan esquemas de gestión del rendimiento y compensación basada en objetivos para sus empleados. Pero la implantación práctica de modelos basados en competencias no es una tarea fácil:

- se inicia con la definición inicial de un modelo con las competencias necesarias para alcanzar los objetivos organizacionales
- sigue con la asignación efectiva de las mismas a los empleados, de acuerdo a los perfiles de los puestos de trabajo necesarios (planificación de personal),
- se produce su ‘observación’ a través del comportamiento de las personas
- su evaluación de acuerdo a los ‘valores’ de la organización
- su desarrollo permanente para mejorar el rendimiento, a través de acciones concretas de formación, compensación u otros medios [7].

No existen todavía marcos de competencias específicos, útiles y no ambiguos para las figuras profesionales relevantes (como los gestores de SI) y en consecuencia, se avanza de forma muy costosa y lenta, por etapas y aproximaciones sucesivas.

Los procedimientos utilizados habitualmente para definir modelos de competencias para CIOs, se apoyan en la distribución masiva y posterior recogida y análisis estadístico de complejos cuestionarios entre miles de compañías para obtener así modelos teóricos y después, tratar de aplicarlos [8]. El enfoque que se propone en esta investigación es

diferente: Por una parte, y con carácter general, disponemos en la organización de esquemas básicos de gestión del rendimiento y compensación variable en función de resultados para los empleados. Por otra, y en el ámbito funcional de Sistemas de Información, las organizaciones deben definir un modelo concreto de gestión de la actividad, que es necesario actualizar permanentemente de acuerdo a los requisitos cambiantes del negocio (internos o externos) y a la propia evolución tecnológica. La definición y aceptación (con la correspondiente validación empírica) de un modelo de gestión de SI genérico para cualquier gran organización debería ser independiente de su actividad concreta y de la que ‘extraeríamos’ los indicadores clave para realizar la actividad de forma ‘excelente’ y definir a partir de ésta información el modelo de competencias. Con dicho modelo podríamos implantar de modo más sencillo y directo prácticas de gestión concretas basadas en competencias, progresando hacia objetivos de madurez en la gestión de RRHH sin dependencias ni opiniones externas a la organización. Si conseguimos así demostrar y crear una función de correlación entre el esquema de competencias y los indicadores del modelo de gestión para la actividad de gestión de Sistemas de Información, estableceremos una vía mucho más eficiente y directa para desarrollar el paradigma competencial en las grandes organizaciones.

El reto es ambicioso y relevante y constituye el objeto fundamental de nuestra línea de investigación, dentro del complejo escenario de gestión de la tecnología y los Sistemas de Información en la nueva economía. La validación del modelo quedará demostrada aplicándolo (con éxito) en el contexto de un proceso relevante de RRHH, como la evaluación del rendimiento, en más de una gran organización. Si el resultado es satisfactorio, se acelerará la vía de aplicación del modelo de competencias útiles y no ambiguos a funciones clave de gestión. El objetivo fundamental de la línea de investigación es, por tanto, proponer un método eficiente para conseguir ese marco competencial y la forma de aplicarlo en la evaluación del rendimiento laboral del CIO y los gestores de Sistemas de Información.

Esta publicación analiza las bases y fundamentos de la línea de investigación indicada, en particular sobre competencias y los modelos de gestionar los Sistemas de Información en el ámbito de la nueva economía.

En el siguiente apartado se analizan distintas aproximaciones sobre el paradigma competencial. El Apartado 3 describe y justifica el modelo de gestión de SI propuesto y en el 4, se esboza un cuadro de mando con indicadores de actividad, ajustados al modelo propuesto. Finalmente, en el Apartado 5, se resumen las conclusiones y se esbozan las actividades y líneas de investigación futuras relacionadas con esta investigación.

2. Diferentes aproximaciones y enfoques sobre el paradigma competencial

Obviando los primitivos enfoques con un sentido puramente lingüístico, las primeras aportaciones prácticas sobre el tema surgen del fundador de la consultora de Recursos Humanos *Hay*, David McClelland. Sus primeros trabajos demuestran que los resultados académicos y los tradicionales tests de inteligencia no predicen de forma fiable el éxito profesional [9]. Tratando de buscar nuevas variables enfocadas a la gestión del rendimiento laboral, aparece, por vez primera, el término de competencia.

La gestión del Capital Intelectual en ámbitos empresariales cercanos a la tecnología ha sido ampliamente investigada [10] [11], pero habitualmente se enfocan a disciplinas relacionadas con la gestión del conocimiento, siendo los modelos de competencias el referente más significativo para la gestión del rendimiento de los empleados, particularmente en organizaciones intensivas en conocimiento.

El interés creciente de la aplicación de modelos de competencias se constata también a través de iniciativas como People Capability Maturity Model o People-CMM [12] generado en el Software Engineering Institute. Este modelo constituye una excelente guía de referencia, contrastada en grandes organizaciones y definida para implantar nuevos modelos de gestión de recursos humanos encaminados a la mejora continua del rendimiento de los empleados. El enfoque competencial en este modelo es esencial, está presente en las fases más críticas de su desarrollo e identifica sin ambigüedad la necesidad de definir el marco de competencias que identifica los conocimientos, habilidades, actitudes necesarias para la gestión de los empleados y facilitar así su desarrollo profesional en la organización:



Figura 1. Niveles de Madurez y procesos en People-CMM

Las investigaciones recientes para obtener marcos de competencias aplicables a funciones de gestión indican que esta disciplina está muy poco desarrollada y también que los resultados obtenidos para evaluar el rendimiento de profesionales de la función de gestión son habitualmente muy ambiguos [13]. La ambigüedad conceptual se manifiesta como consecuencia de la propia complejidad de la actividad de gestión, de modo que los modelos existentes incorporan conceptos contradictorios, las competencias pueden ser dependientes o independientes en relación al comportamiento gerencial o incorporan mayoritariamente propiedades orgánicas o genéricas.

El ámbito anglosajón ha sido el más activo en aportaciones, y se puede hablar de dos enfoques de investigación diferentes, en función del origen de las mismas, Reino Unido o EEUU. En el primero, el término competencia se asocia a la definición de tareas que un trabajador de un determinado sector ocupacional debe saber realizar. En este sentido, y con el objetivo de mejorar el rendimiento de las empresas mejorando la calidad de los gestores británicos, la organización empresarial Management Charter Initiative (MCI) publica unos estándares de gestión en 1991, definiendo las habilidades que los gestores, en cualquier sector de actividad, deberían poseer. En contraste con este enfoque, la *American Management Association* define competencia como la característica intrínseca de una persona que genera rendimiento superior (concepto de búsqueda de la excelencia profesional) en un trabajo [14]. Las competencias no se identifican con las tareas asociadas

o requeridas para ocupar un puesto de trabajo, sino que se definen en función de las características de la persona que hacen que su rendimiento sea superior a la media. El enfoque británico está más dirigido a la acreditación o certificación, mientras que el americano se enfoca más al rendimiento.

Investigaciones como la realizada por [15] sugieren que ambos enfoques resultan incompletos para la creación de marcos de competencias orientados a la gestión del rendimiento de gestores en las organizaciones, y proponen avanzar en la creación de nuevos modelos que sepan aprovechar ideas y conceptos de ambas aproximaciones.

Considerando los requisitos de nuestros empleados clave, directivos y gerentes, proponemos que los modelos basados en el paradigma de las competencias, se deben implantar a partir de un marco de definición de las mismas que permita:

- Soportar los objetivos del Negocio, gestionando información objetiva para adquirir, mantener, influenciar, desarrollar y retener a los empleados adecuados.
- Alinear personas, procesos y tecnología alrededor de valores compartidos.
- Medir el valor para la organización de las inversiones en Capital Humano, en esas áreas estratégicas.
- Anticiparse a los cambios y aprender continuamente de las mejores prácticas, generadas a partir de nuevas iniciativas de investigación y del propio mercado, utilizando información contrastable e indicadores adecuados.

En nuestro trabajo, asumiremos un esquema competencial mayoritariamente aceptado por los responsables en gestión de recursos humanos, y que distingue dos tipos o familias de competencias:

1. Genéricas: denominadas clave en la literatura anglosajona, definen características referidas al comportamiento general del empleado, independientes de los conocimientos técnicos concretos que pueda o no tener. Ejemplos de este tipo son la capacidad de síntesis y análisis, la capacidad de organización, el compromiso ético, etc.
2. Específicas: son las asociadas a conocimientos y habilidades técnicas específicas de cada puesto de trabajo.

La gestión por competencias asume una definición inicial de puestos de trabajo, requeridos en términos de competencias genéricas y específicas. Para dimensionar

adecuadamente una plantilla laboral, se precisa además conocer la carga de trabajo prevista para los diferentes puestos. Y para poder evaluar el rendimiento de las personas, se necesita identificar al menos, los objetivos que se proponen (y asume) el empleado como alcanzables en el período de evaluación. Se cumplimentará así el perfil de exigencias para el puesto [16], en términos de competencias genéricas y específicas, para poder plantear la evaluación del rendimiento en función de los objetivos que se han definido. Las técnicas para realizarla pueden ser muy diversas, desde ejercicios que simulan situaciones reales en la empresa o evaluaciones 360⁰ a partir de información obtenida de superiores, subordinados o incluso clientes. Debemos investigar entre las diferentes opciones y elegir la más idónea para los gestores de SI. Por otra parte, nos encontramos en un escenario laboral en el que la aplicación práctica del paradigma competencial es muy novedoso, poco automatizado y difícil de implantar. Para los gestores de SI, no existe un marco de competencias completo, fiable y no ambiguo, con contenido y cuantificación de las competencias genéricas y específicas para conseguir la ‘excelencia’ en la profesión y finalmente, tampoco disponemos de un método concreto y fiable (además de soportado en una herramienta Software) para evaluar su rendimiento profesional.

3. Un modelo de gestión de la actividad de SI en la nueva economía

Las recientes teorías económicas así como las evidencias acerca del crecimiento que se observa a partir de la década de los noventa [17] identifican dos características definitorias del fenómeno de la nueva economía: la creciente globalización y la continua expansión de las tecnologías de la información. Estos dos aspectos no son síntomas de otros factores, son los factores que están propiciando el cambio y el crecimiento de la economía.

En este contexto, uno de los desafíos más importantes que han tenido las unidades o áreas de SI en las grandes organizaciones, ha sido el saber comunicar a las unidades de negocio el valor que su actividad aporta a la organización. Frecuentemente la comunicación se establece en términos de tecnología o de reducción de costes, y no en términos de aportación de valor para el negocio. Este problema de ‘alineamiento’ genera en las organizaciones escenarios muy variables de dependencia entre la credibilidad de SI y la percepción que el negocio tiene de su dependencia en la información para conseguir sus objetivos. Esta es una de las razones fundamentales por las que investigadores como [18]

estudian los aspectos estructurales de SI en las organizaciones de modo que sean entendidas por gestores y ejecutivos de negocios y faciliten el alineamiento entre SI y los objetivos empresariales. Los marcos de referencia para el ámbito de los SI, suelen orientarse a modelos estructurales complejos, nada adecuados para la gestión competencial. Otra forma clásica de enfocar la gestión [19] asume un modelo excesivamente simple de fases en el uso de los SI. El primer rol organizativo de los ordenadores fue la automatización de procesos intensivos en fuerza laboral para su realización. La segunda fase se orientó a los sistemas que generaban información útil para la toma de decisiones. En la tercera y actual, las organizaciones perciben las tecnologías de información como herramientas transformadoras, que generan ventajas competitivas en la creación de nuevos productos, servicios y procesos.

Del análisis del modelo propuesto por [18] concluimos que podría ser útil para construir una taxonomía en SI (en realidad se construyó como soporte a este objetivo), adaptándose a un número considerable de estructuras implantadas. Sin embargo, debemos considerarla excesivamente compleja ya que no encaja con algunas tendencias de gestión relevantes en la actualidad, como la orientación a procesos y la externalización de actividades, y, sobre todo, no se ajusta al objetivo de formular un marco de competencias para gestores de esta actividad.

El modelo que proponemos para cualquier gran organización es considerar la actividad de sistemas estructurada en forma de una cadena de valor, con la función 'informática' en el centro, suministradores de productos y servicios externos en un extremo (lado proveedor) y unidades internas de la organización a las que se prestan servicios (en el lado cliente) y con las tareas gerenciales definidas en el contexto de tres macro procesos:

- Dirigir la innovación: incluye la planificación estratégica, el diseño de la arquitectura de los sistemas y aplicaciones, así como los procesos y requisitos estratégicos del negocio.
- Gestionar el cambio: incluye el desarrollo de los SI y el soporte a los cambios (mantenimiento correctivo y evolutivo) que requieren los usuarios (clientes, en las unidades de negocio).
- Soportar la infraestructura: incluye las operaciones de los centros de procesamiento de datos, redes de comunicaciones y mantenimiento de los puestos de trabajo.

La forma en la que las empresas organizan su actividad de SI (federada, distribuida, centralizada, etc.) responde a sus prioridades, su cultura organizativa y sus objetivos particulares, pero ninguna función o actividad informática quedaría fuera de alguna de las tres áreas del esquema anterior. Por otra parte, esta agrupación identifica características competenciales homogéneas para funciones de gestión de la actividad, lo que facilitará enormemente la definición del marco de competencias. A partir de aquí, y sin olvidar los objetivos de rendimiento de dicho marco, es necesario considerar los factores que le condicionan. En [20] se proponen sólo cuatro: estrategia, entorno, tecnología y estructura. Nuestra propuesta no resulta tan limitada ya que incluye:

- Estrategia: formulación de acciones dirigidas a generar valor y lograr una ventaja competitiva sostenible. En nuestro modelo, la actividad estratégica del gestor se desarrolla en función del cliente, y por lo tanto debe enfocarse a la alineación de SI con el Negocio. El ámbito operativo para el desarrollo de la estrategia es el identificado en la dirección de la innovación (Plan de Sistemas, Arquitectura tecnológica y de aplicaciones,...).
- Valores compartidos: las ideas de lo que es correcto y deseable, en el ámbito individual y corporativo.
- Habilidades de la organización: capacidades que desarrolla el área en conjunto, como consecuencia del desarrollo de competencias individuales en el Grupo.

Y a partir de estas tres dimensiones, se construyen las siguientes:

- Estructura organizativa: áreas específicas de responsabilidad dentro de la organización de SI y la forma en que se integran y coordinan.
- Puestos de trabajo: definición funcional, en términos de roles y no de personas concretas que se necesitan para realizar el trabajo en la unidad.
- Procesos y sistemas: definición de cómo se realiza el trabajo (procedimientos) en el día a día.
- Estilo directivo: condiciona el uso del tiempo y las prioridades que se establecen en la gestión. Su contexto operativo es el Cuadro de Mando para la función de SI, que por su relevancia para el marco de competencias se analiza en el punto siguiente.

4. El cuadro de mando para el gestor de SI

Si pretendemos finalmente evaluar el rendimiento de la función de gestión de SI utilizando el paradigma competencial, en vez de definir el perfil profesional del gestor a partir de tareas y funciones teóricas que el puesto requiere, trabajamos con un esquema de objetivos e indicadores de rendimiento asociados a cada uno de ellos. Estos responden al modelo de gestión indicado en el punto anterior y permiten obtener un marco de competencias que responda a las expectativas establecidas para nuestro trabajo.

Este esquema puede tener una estructura de cuadro de mando funcional, que coloca a la estrategia en el centro de los procesos, y que ‘traduce’ la misión y la estrategia del gestor en objetivos relevantes y que pueden medirse, según el esquema de la figura:

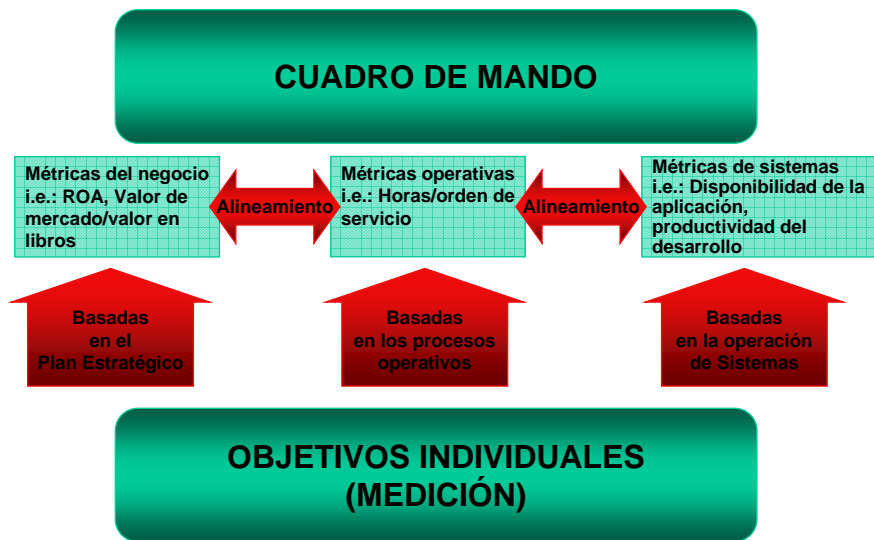


Figura 2. Construcción de un Cuadro de Mando para la gestión de SI

La metodología consiste en empezar identificando los objetivos individuales, estratégicos para el gestor, contruidos en forma de árbol que muestre sus relaciones, en función de los indicadores que inciden positiva o negativamente en su cumplimiento. Se obtienen manual o automáticamente a partir de información básica, indicadores y métricas asociadas a la actividad u operación del día a día. Las métricas deben ser relevantes y

modificables por decisiones o actuaciones del gestor. Además, para poder ser utilizadas en evaluación, deben ser comparables por lo que se puede establecer el nivel de ‘excelencia’ que se persigue en el modelo de gestión por competencias.

En el modelo propuesto, consideramos cuatro perspectivas o niveles, según tres macro-procesos que identificamos en el punto anterior. Cada perspectiva ayuda a la consecución de los objetivos del nivel superior, estableciendo así un vínculo causa-efecto entre los distintos objetivos. A continuación se identifican estos niveles y la razón por la que se proponen en el modelo:

1. La perspectiva financiera: mide el valor que la gestión de sistemas proporciona a sus clientes, las unidades de negocio. Es el nivel superior, y se justifica por la idea de hablar una lengua común entre SI y negocio, de modo que se puedan evaluar y priorizar inversiones, decidir cómo y cuando hacer cambios o qué puede o no puede cambiarse.
2. La perspectiva del cliente: responde al concepto de orientación al cliente que se indicó en puntos anteriores. Incorpora indicadores de valor para las unidades de negocio, tales como tiempos de ejecución o características y calidad de los servicios informáticos.
3. La perspectiva de procesos: permite evaluar si los procesos internos de SI responden a las expectativas de servicio que demandan las unidades de negocio. Calidad y tiempo de realización de aplicaciones o acciones de mantenimiento en los sistemas, son indicadores típicos para este nivel.
4. La perspectiva de los recursos: se mide en términos de resultados en la gestión de inversiones en personas, infraestructuras y tecnología, para alcanzar los objetivos planificados a medio y corto plazo en SI.

La figura 3 representa una estructura propuesta de objetivos que asocia los cuatro niveles definidos a los tres macro-procesos del modelo de gestión.

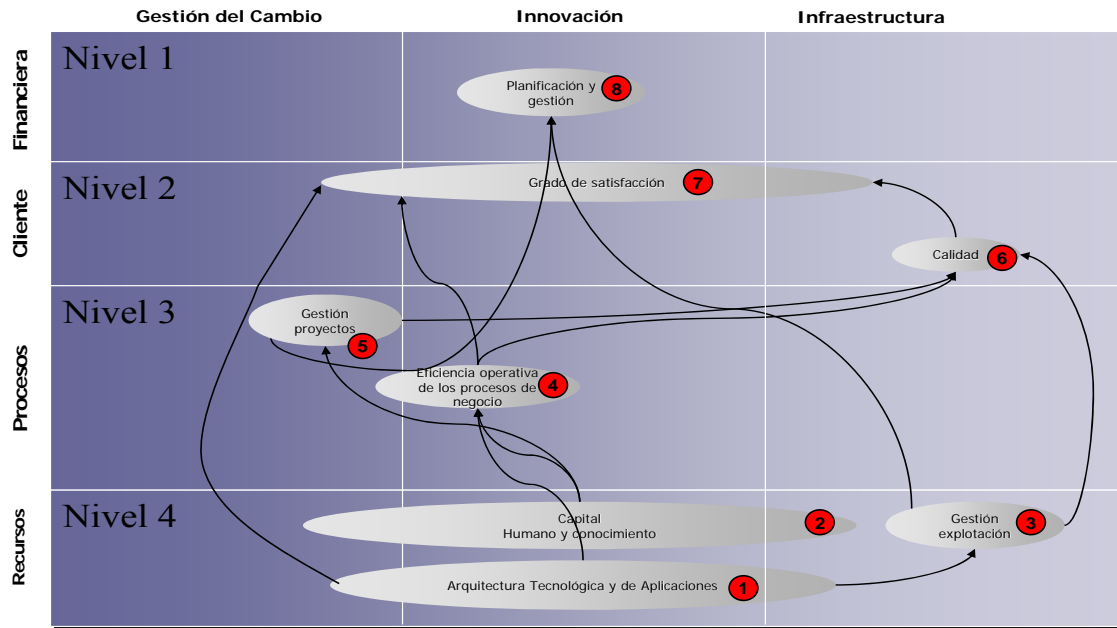


Figura 3. Árbol estratégico de objetivos para la gestión de Sistemas de Información

5. Conclusiones

La creación de un marco de competencias que posibilite la gestión del rendimiento laboral del CIO y otros gestores de SI y Tecnología es un reto complicado, pero factible. Para conseguirlo, se requiere aplicar adecuadamente una metodología para implantar Sistemas de gestión de RRHH soportados en el paradigma de las competencias y utilizar desde el inicio, información e indicadores de rendimiento que aseguren coherencia y alineamiento de su actividad profesional con los retos y expectativas que las grandes organizaciones esperan de su desempeño.

Como líneas futuras de investigación en esta área de conocimiento será necesario desarrollar y ampliar el esquema de indicadores asociados a objetivos incluidos en el modelo de gestión propuesto y desarrollar a continuación el marco completo de competencias. En función de los resultados obtenidos y validados en distintos escenarios empresariales, se podría ampliar el ámbito de aplicación del marco a nuevas figuras profesionales, particularmente en el ámbito de la gestión técnica y empresarial.

Referencias

- [1] Dearden, J., “The Whitering Away of the IS Organization”, *Sloan Management Review*, vol 28, nº4, pp. 87-91, 1987.
- [2] Hartman, A. y Sifonis, J., *Net Ready. Strategies for success in the E-economy* McGraw-Hill, 2000.
- [3] Reich, B.H y Nelson, K. M. “In their own words: CIO visions about the future of in-house IT organizations” *The DATA BASE for Advances in Information Systems*, vol 34, nº 4, pp.21-44, 2003.
- [4] Nordstrom, A.K. y Ridderstrale, J. (2000) *Funky Business. Talent makes capital dance*. Book House Publishing AB 11438 Stockholm, 2000.
- [5] Sagi-Vela, L., *Gestión por competencias: El reto compartido del crecimiento personal y de la organización* ESIC Editorial, 2004.
- [6] Bharadwaj, A.S., “A resource based perspective on Information Technology capability and Firm performance: An empirical Investigation” *MIS Quarterly*, vol. 24 nº1, pp 169-196, 2006.
- [7] Levy-Leboyer, C (1997) *Gestión de las competencias. Cómo analizarlas, cómo evaluarlas, cómo desarrollarlas*, Ediciones Gestión 2000,SA., 1997.
- [8] Ravarini A. et al.(2004) *Exploring the impact of CIO competencies on company performance*, Universita Cattaneo, LIUC, 2004.
- [9] McClelland, D., “Testing for competence rather than for intelligence” *American Psychologist*, vol 28 pp.1-14, 1973.
- [10] Lindgren, R., Stenmark, D.y Ljungberg, J. “Rethinking Competence Systems for Knowledge based organizations” *European Journal of Information Systems*, vol. 12 nº1, pp.18-29, 2003.
- [11] Ward,J. y Aurum,A., “Knowledge Management in Software Engineering: describing the process” *Proceedings of the 2004 Australian Software Engineering Conference (ASWEC’04)*, pp.137-146, 2004.
- [12] Curtis, B, *People Capability Maturity Model (P-CMM) Version 2.0*, CMU/SEI-2001-MM-01,Carnegie Mellon, The Software Engineering Institute, Pittsburg USA, 2001.
- [13] Grzeda, M.M., “In Competence we trust: addressing conceptual ambiguity”, *Journal of Management Development*, 2005, vol. 24 nº6, pp.530-545, 2005.

- [14] Boyatzis, R.E. *The competent manager: A model for effective performance*, John Wiley & Sons , New York, 1982.
- [15] Mei-I Cheng & Dainty, A., 2002 “The differing faces of managerial competency in Britain and America” *Journal of Management Development*, vol 22, nº6, pp. 527-537, 2002.
- [16] Pereda S. y Berrocal, F. *Técnicas de Gestión de Recursos Humanos por competencias*, Centro de Estudios Ramón Areces, Madrid, 2001.
- [17] Stiroh, K.J., “Is there a new economy?”, *Challenge*, vol 42, nº4, pp.175-191, 2000.
- [18] Bacon, C.J y Fitzgerald, B., “A systematic Framework for the Field of Information Systems”, *The DATA BASE for advances in Information Systems*, vol.32, nº2 pp. 46-67, 2001.
- [19] Ward, J. & Griffiths, P.M., *Strategic Planning for Information Systems*, New York, John Wiley & Sons, 1996.
- [20] Houtzagers, G., “Empowerment, using skills and competent management” *Participation & Empowerment International Journal*, MCB University Press, vol.7 nº2 pp.27-32, 1999.