Revista de Procesos y Métricas de las Tecnologías de la Información (RPM) ISSN: 1698-2029
VOL. 1, Nº 2, Agosto 2004, 25-30
Asociación Española de Métricas de Sistemas Informáticos (AEMES)

MODELOS SEGMENTADOS DE ESTIMACIÓN DEL ESFUERZO DE DESARROLLO DEL SOFTWARE: UN CASO DE ESTUDIO CON LA BASE DE DATOS ISBSG

J. Cuadrado–Gallego
Dpto. Informática. Universidad de Valladolid.
40005, Plaza de Santa Eulalia, 9. Segovia , Spain.
E-Mail : jjcg@infor.uva.es

Daniel Rodríguez
Departament of Computer Science, University of Reading
Reading, RG6 6AY, UK
E-Mail: d.rodriguezgarcia@rdg.ac.uk

Miguel–Ángel Sicilia
Departamento de Ciencias de la Computación, Universidad de Alcalá
28871, Ctra. de Barcelona, 33.6, Alcalá de Henares, España
E-Mail: msicilia@uah.es

Abstract: Parametric sofware effort estimation models use historical project databases to adjust the parameters of the required effort function. The use of databases with data coming from heterogeneous sources often entail that the resulting models are subject to excessively high mean errors, due to the fact that data are widely diverging in magnitude. In order to improve this situation, the segmentation of the input domain has been proposed, so that a regression model with different parameters is obtained for each segment. In this paper, the use of well-known clustering algorithms in a recursive way is described as a technique to obtain segmented models of the kind mentioned. Concretely, the ISBSG database and the EM algorithm are use as a demonstration of the results that can be obtained through that technique.

Resumen: Los modelos paramétricos de estimación del esfuerzo de desarrollo de software utilizan bases de datos de proyectos pasados para ajustar la función del esfuerzo requerido. El uso de bases de datos cuyas fuentes son heterogéneas hace que los modelos obtenidos mediante regresión a menudo tengan errores medios excesivamente altos, debido a que los datos son muy diferentes entre sí en cuanto a su magnitud. Para mejorar esta situación se ha propuesto el dividir el dominio de las entradas en varios segmentos, de modo que se obtenga un modelo de regresión con diferentes parámetros para cada uno de ellos. En este artículo se describe cómo se pueden utilizar algoritmos conocidos de agrupamiento de manera recursiva para obtener ese tipo de modelo segmentado. Concretamente, se utiliza la base de datos ISBSG y el algoritmo EM como demostración de los resultados que pueden obtenerse mediante dicha técnica.

Palabras Clave: Ingeniería del Software, Estimación de Esfuerzo, Modelos Matemáticos.

1. INTRODUCCIÓN

La estimación del esfuerzo y en consecuencia del coste, así como el tiempo, que se empleará en el desarrollo de un producto software es un aspecto directamente relacionado con el éxito del proyecto.

Uno de los métodos de estimación de más extendidos es la utilización de ecuaciones matemáticas en las que la variable dependiente es el esfuerzo o el tiempo, y las variables dependientes son diferentes aspectos del proyecto o del producto o de ambos. La forma ecuaciones matemáticas se obtienen principalmente mediante dos sistemas: o bien son definidas por expertos a partir de su experiencia, como es el caso, por ejemplo del conocido modelo de Boehm [Boehm, 2000]; o bien se deducen a partir de la utilización de técnicas estadísticas aplicadas a bases de datos de proyectos históricas.

Cuando las bases de datos de proyectos se han obtenido mediante la integración de datos heterogéneos, provenientes de diferentes organizaciones y tipos de proyectos, la generación de un modelo paramétrico único para toda la base de datos puede resultar en un ajuste técnicamente pobre, dado que se están combinando proyectos de diferente naturaleza que hacen difícil el que un modelo matemático como una función potencia del tipo

$$e = a \cdot s^b$$

(donde *e* es el esfuerzo estimado y *s* alguna medida del tamaño del proyecto) obtenga un ajuste de los parámetros adecuado para todo el dominio de los posibles tamaños. Esta dificultad se ve agravada por el hecho de que esas bases de datos no suelen tener una distribución uniforme de los puntos dentro del dominio de los tamaños, siendo mucho más dispersas en valores grandes del dominio.

Veamos un ejemplo de los problemas indicados. Tomaremos para ello los datos utilizados en la herramienta *Reality* de la base de datos de proyectos ISBSG version 8 (*International Software Benchmarking Standard Group*[1]). Esta herramienta toma una selección de los datos de proyecto de la base de datos ISBSG filtrada por la calidad estimada de los datos, y utilizando sólo datos obtenidos mediante IFPUG 4 o variantes. Los 709 datos de proyecto resultantes, si no consideramos más variables, se ajustan mediante la función potencia:

$$e = 47.73 \cdot s^{0,76} \qquad (1)$$

Donde el esfuerzo se expresa en horas, y el tamaño en puntos de función. Un análisis de la bondad del ajuste nos da como resultado MMRE=1.18 y PRED(.30)=25.6%.

El MMRE es el valor absoluto del error relativo medio de un conjunto de proyectos MMRE = $|\overline{RE}|$, siendo el error relativo medio la diferencia entre el valor real y el estimado, dividido por el valor real.

El PRED(*p*) indica da una medida del porcentaje de datos cuya estimación se ha desviado del valor real obtenido en una cantidad inferior a *p*. Por ejemplo si PRED (0,25)=0,8, esto indica que el 80% de los valores de estimados se diferencian en menos de un l 25% de sus correspondientes valores reales

Ambas medidas son difícilmente aceptables dado el alto grado de desviación sobre la inmensa mayoría de los datos, lo cual apunta a la necesidad de buscar mejores ajustes. Dado que la selección de los datos ya se supone ha eliminado posibles *outliers*, hay que buscar técnicas alternativas.

Algunos autores han sugerido [Riefer, 1998], [Abran, 2003] que la segmentación de los datos contenidos en las bases de datos históricas podría ser un camino adecuado para la obtención de ecuaciones matemáticas que proporcionen una mayor exactitud en las estimaciones.

En este trabajo, abordamos la generación de modelos paramétricos de estimación segmentados, es decir, aquéllos para los cuales se proporcionan diferentes funciones dependiendo de la magnitud de los *cost drivers* – en nuestro caso, del tamaño en puntos de función. En lugar de hacer una división arbitraria de las entradas, se utilizan algoritmos de agrupamiento (*clustering*) conocidos para dividir el dominio, y en caso de que el ajuste aún no sea "suficientemente" bueno, se procede a divisiones recursivas. Este esquema general admite muchas variantes que abren nuevas vías a la experimentación.

Otros trabajos utilizado algoritmos de agrupamiento borroso para determinar segmentos en las entradas [Xu, Taghi and Khoshgoftaar, 2004], pero no con el objetivo de mejorar el ajuste de modelos paramétricos de estimación.

El resto de este artículo se organiza de la siguiente manera. En la segunda sección se describe la técnica utilizada para la obtención del modelo de estimación segmentado. En la tercera sección se describe como caso de estudio un análisis cuantitativo de la

[1] http://www.isbsg.org/

efectividad de la técnica sobre una selección de la base de datos ISBSG. Finalmente, las conclusiones de la investigación y algunas posibles líneas de continuación futura se describen en la cuarta sección.

2. DESCRIPCIÓN DE LA TÉCNICA DE OBTENCIÓN DEL MODELO SEGMENTADO

La técnica de segmentación y regresión recursiva puede describirse esquemáticamente mediante el siguiente pseudocódigo:

```
Generar_modelo(CD:conjunto_datos,
               MMRE_deseado,
               PRED_deseado,
               PRED_PARAM: real;
       TIPO_MODELO: funcion): Modelo;
Inicio
   MODELO:=obtener_modelo(CD,
                          TIPO_MODELO);
   MMRE  := calcular_mmre(CD, MODELO);
   PRED  := calcular_pred(CD,
                PRED_PARAM, MODELO);

   Si (MMRE <= MMRE_deseado o
       PRED <=PRED_deseado
       o <<pocos datos>>) Entonces
        Clusters :=Dividir(CD);
     Desde i:=1
         hasta Clusters.longitud hacer

         Modelos[i]:=Generar_modelo(
                    CD{Clusters[i]},
                    MMRE_Deseado,
                    PRED_Deseado,
                    PRED_PARAM,
                    TIPO_MODELO);
     Fin_Desde
     Return Modelos;
   Fin_Si
   Return MODELO;
Fin
```

En el pseudocódigo anterior, se asume que la salida de el algoritmo de *clustering* (invocación a `Dividir`) es una estructura de datos con tantas posiciones como grupos obtenidos, en la cual, `Clusters[i][centro]` indica el "centro" estimado del cluster, `Clusters[i][alcance]` el alcance estimado del radio del grupo respecto al centro, y `Clusters[i][tamaño]` el tamaño aproximado en número de elementos del grupo. Esos datos se utilizan para filtrar el conjunto de datos original, lo cual se expresa mediante `CD{Clusters[i]}`. El algoritmo devuelve o bien una función de estimación ajustada directamente, o una función de estimación segmentada. Por ello, el tipo Modelo ha de ser recursivo, pero aquí no entramos en los detalles de la implementación.

Nótese que este tipo de algoritmos puede ser fácilmente automatizado, convirtiéndolo en una heurística de búsqueda de ajustes segmentados de propósito general.

El algoritmo descrito podría aún hacerse más genérico en varias direcciones. Por ejemplo, podrían especificarse otros tipos o criterios de medidas de calidad del ajuste, utilizar varios algoritmos de agrupamiento simultáneamente, examinando qué alternativa es mejor, o podrían utilizarse combinaciones de modelos matemáticos para las funciones de estimación.

3. CASO DE ESTUDIO

El caso de estudio descrito en esta sección utiliza los 709 datos seleccionados de la base de datos ISBSG que también usa la herramienta *Reality*, antes

mencionada. Como algoritmo de *clustering*, se ha utilizado la implementación del algoritmo EM proporcionada en las herramientas de fuente abierto WEKA[2]. El algoritmo de agrupamiento EM (Expectation-Maximization) [Dempster et al., 1977] es una variante de algoritmos de agrupamiento en árbol que calcula probabilidades de pertenencia a un cluster, basadas en una o más distribuciones de probabilidad, utilizando como objetivo la maximización de la probabilidad global. Este algoritmo es especialmente adecuado para casos en los que no se precisa pre-determinar el número de clusters que han de generarse como salida. Dado que en el contexto que aquí tratamos no se tiene ninguna información *a priori* sobre la forma o número de grupos en la base de datos de proyectos, el algoritmo EM es preferible a algoritmos que predeterminan el número de grupos de salida. Aunque el algoritmo EM asigna una probabilidad de pertenencia a cada elemento en cada cluster, en este estudio se utiliza una asignación de puntos a clusters determinados, determinada por el centro (media) de los clusters en el dominio del tamaño del software, que será el que se utilice al hacer estimaciones de proyectos concretos.

En lo que sigue se describen los datos obtenidos según el esquema algorítmico descrito en la sección anterior.

Después de los datos originales descritos en la introducción, se procede a la primera división. Los grupos obtenidos se describen en la Tabla 1, en dicha tabla se presentan la media y la desviación estándar junto a los puntos asignados a cada cluster, de acuerdo a la herramienta WEKA.

	Mean (fp)	Std. Dev (fp)	Approx. Size
Cluster 0	78,27	33.77	170
Cluster 1	194.39	85.57	255
Cluster 2	286.23	135.31	185
Cluster 3	1256.44	719.68	88
Cluster 4	4774.68	4469,85	11

Tabla 1. Resultados de la primera segmentación

La calidad de los modelos de regresión para cada uno de los clusters obtenidos se resume en la Tabla 2. Los parámetros de calidad propuestos han sido el PRED y el MMRE. Se presentan también dos columnas que contienen los datos que se obtienen cuando se aplica la ecuación ofrecida por *Reality* sobre esos grupos de datos.

	Modelo	MMRE	PRED(.3)	MMRE (original)	PRED(.3) (original)	Aceptable
Cluster 0	$e = 251.5 \cdot fp^{0.19}$	0.66	31.1%	2.1	20.6%	No
Cluster 1	$e = 9647 \cdot fp^{-0.3}$	0,39	46.2%	1.03	30%	Si
Cluster 2	$e = 26670 \cdot fp^{-0.27}$	0,37	53%	0.78	20.5%	Si
Cluster 3	$e = 29120 \cdot fp^{-0.083}$	0,53	43%	0.74	18%	Si
Cluster 4	$e = 29120 \cdot fp^{-0.083}$	0,28	72%	0.6	9%	Si
Media		**0,446**	**50%**			

Tabla 2. Calidad de los modelos de regresión obtenidos en la primera segmentación

Es importante observar que en término medio, la segmentación resultante mejora muy significativamente a la versión original que propone la utilización del mismo modelo para todos los puntos. También cabe resaltar que los modelos para los clústers 3 y 4 son idénticos, por lo que podrían combinarse en un solo grupo.

Si se utiliza como criterio de calidad aproximado MMRE<=0.5 y PRED(.3)>=40%, sólo el cluster 0 no cumple con las condiciones. Por supuesto que ese criterio puede considerarse como excesivamente laxo, y debería haberse considerado el conocido MMRE<=0.25 y PRED(.25)>=75% [Dolado, 2001] pero el objetivo de esta investigación es demostrar la bonanza de la segmentación iterativa, lo cual queda demostrado a partir del análisis de un solo cluster. Y el criterio escogido permite hacerlo así. Queda para posteriores trabajos el alcanzar el criterio más restrictivo.

Con todo lo anterior si se toman los datos del clúster 0 y se vuelve a aplicar el algoritmo EM, se obtienen los siguientes grupos de segundo nivel, como se puede ver en la Tabla 3. Dicha tabla tiene la misma estructura descrita para la Tabla 1.

	Mean (fp)	Std. Dev (fp)	Approx. Size
Cluster 0-0	41,72	19,19	32,00
Cluster 0-1	78,90	28,50	102,00
Cluster 0-2	152,80	60,48	36,00

Tabla3. Resultados de la segunda segmentación del cluster 0

Teniendo en cuenta los tres nuevos sub-grupos, el análisis de la calidad del ajuste queda como muestra la Tabla 4.

[2] http://www.cs.waikato.ac.nz/~ml/weka/

	Modelo	MMRE	PRED(.3)	MMRE (original)	PRED(.3) (original)
Cluster 0-0	$e = 246{,}9 \cdot fp^{-0.0225}$	0,47	34,4%	3,2	6%
Cluster 0-1	$e = 5716 \cdot fp^{-0.4994}$	0,37	55%	1,9	21,5%
Cluster 0-2	$e = 54220 \cdot fp^{-0.8557}$	0,13	94%	1,71	30,5%
Cluster 1	$e = 9647 \cdot fp^{-0.3}$	0,39	46.2%	1.03	30%
Cluster 2	$e = 26670 \cdot fp^{-0.37}$	0,37	53%	0.78	20.5%
Cluster 3	$e = 29120 \cdot fp^{-0.083}$	0,53	43%	0.74	18%
Cluster 4	$e = 29120 \cdot fp^{-0.083}$	0,28	72%	0.6	9%
Media		0,36	57%		

Tabla 4. Calidad de los modelos de regresión obtenidos tras la segunda segmentación

Como se puede observar en la Tabla 4, el MMRE obtenido para los tres sub-grupos obtenidos a partir del grupo 0 son significativamente mejores (0.47, 0.37, 0.13) que los obtenidos, tanto para el grupo 0 en el nivel inmediatamente superior (0.66) como para el conjunto total de datos del primer nivel (1.18).

Esta sustancial mejoría también se puede observar para el PRED que pasa a ser de un 25.6% del conjunto total de datos del primer nivel, a un 31% en el cluster 0. Pero experimenta una extraordinaria mejoría en el siguiente nivel obteniéndose un 34.4%, 55% y 94% para los clusters 0-0, 0-1 y 0-2, respectivamente. Lo cual indica mejoras que llegan a mas del 200% en el caso del cluster 0-2.

Otro aspecto interesante que se puede observar en la Tabla 4. Es la mejoría significativa que ofrecen también las ecuaciones matemáticas segmentadas frente a la ecuación global, incluso cuando ésta última se aplica sobre los clusters de datos. Llegando tener una mejoría en el PRED de más del doble para el cluster 0-1, un 55% de la ecuación segmentada frente a un 21.5% de la ecuación global; e incluso de más del triple para el cluster 0-2, un 94% de la ecuación segmentada frente a un 30,5% de la ecuación global

De este análisis de la Tabla 4 se puede concluir que, la técnica de segmentación de datos, para la obtención de las ecuaciones segmentadas de estimación del esfuerzo y el tiempo de desarrollo de productos software, es adecuada para producir resultados de una mayor exactitud.

Dado el modelo segmentado obtenido, la estimación para un proyecto concreto tendrá entonces dos pasos:

1) Seleccionar el clúster cuyo centro está más cerca del tamaño dado.
2) Utilizar el modelo de ese clúster para realizar la estimación.

En casos en los cuales un determinado punto esté razonablemente “entre más de un clúster”, se puede ampliar el procedimiento anterior de modo que se utilicen los modelos de los clúster más próximos, generando una estimación intermedia entre las estimaciones parciales, por ejemplo, mediante una media aritmética. De este modo se obtiene una solución de compromiso entre estimaciones de clústers a los cuales potencialmente podría pertenecer el punto dado.

4. CONCLUSIONES Y TRABAJO FUTURO

En el trabajo presentado se haa demostrado que la utilización de una técnica de segmentación o “clusterización” de los datos pertenecientes a bases de datos heterogéneas, tanto en la forma como en el origen, es un método adecuado como paso previo, para la obtención de ecuaciones matemáticas de estimación de esfuerzo o tiempo de desarrollo, para proyectos de desarrollo de software.

Dicho método tiene una estructura iterativa, de tal forma que su aplicación continuada en diferentes niveles va proporcionando ecuaciones de precisión creciente, tal y como demuestra los indicadores PRED y MMRE. En este trabajo se ha presentado una primera versión del algoritmo de iteración.

Como trabajos futuros se proponen entre otros: la automatización del algoritmo de iteración en una herramienta o el análisis de la bonanza de diferentes métodos de “clusterización”, entre los cuales se encontraría la aplicación de técnicas de Soft Computing.

5. AGRADECIMIENTOS

Este trabajo ha sido realizado con el soporte del proyecto CICYT: IN2GESOFT. TIN2004-06689-C03-00. INnovación e INtegración de métodos para el desarrollo y GEstión cuantitativa de proyectos SOFTware

6. REFERENCIAS

Abran, A. (2003). Software Estimation: Black Box or White Box. Presentado en el Workshop. ADIS 2003

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977). Maximum Likelihood from Incomplete Data via the EM algorithm. Journal of the Royal statistical Society, Series B, 39(1): 1-38.

Dolado, J.J. (2001) On the problem of the software cost function. Information and Software Technology, 43, 61-71

Reifer, D., Boehm, B., Chulani, S. (1999). The Rosetta Stone. Making COCOMO 81 Estimates Work with COCOMO II. CrossTalk, 12 (2), 11-15

Xu, Z., Taghi, T.M. and Khoshgoftaar, M. (2004). Identification of fuzzy models of software cost estimation. *Fuzzy Sets and Systems*, 145(1), 141-163