

TAMAÑO DE LOS PROYECTOS DE SOFTWARE Y ECONOMÍAS DE ESCALA EN EL BANCO DE DATOS ISBSG

Marta Fernández-Diego

José-de-Paula Maciel

José-María Torralba-Martínez

Universidad Politécnica de Valencia

Daniel Marcelo Llácer

Enseñanza

Abstract

An analysis of the existence of software projects economies of scale is presented in this paper, based on the ISBSG (International Software Benchmarking Standards Group) dataset Release 11. The results obtained using linear regression suggest the existence of economies of scale. In fact, investigations of relevance point out the importance of replicating previous results, as there has been some debate between researchers about economies of scale. Thus, special care was taken to register all the process followed in this study in order to facilitate replication.

Keywords: *software projects; ISBSG dataset; economies and diseconomies of scale; project size in function points; effort in person-hours*

Resumen

Se ha realizado un análisis sobre la existencia de economías de escala en proyectos de software, en el banco de datos ISBSG (International Software Benchmarking Standards Group), versión 11. Los resultados obtenidos mediante regresión lineal, apuntan a la existencia de economías de escala. Como se indica en la metodología de investigación, es importante realizar repeticiones de las investigaciones, ya que no hay consenso científico sobre los resultados. Por ello, se ha puesto especial cuidado en registrar todo el proceso seguido en este estudio, con la finalidad de que pueda ser replicado.

Palabras clave: *proyectos de software; banco de datos ISBSG; economías y diseconomías de escala; tamaño en puntos de función; esfuerzo en horas/persona*

1. Introducción

La existencia o no de diseconomías de escala en proyectos de software, se viene analizando empíricamente desde la década de 1970 (Boehm, 1981), pero no se produce acuerdo entre los investigadores (Kitchenham, 2002; Premraj et al., 2005) sobre la existencia de diseconomías o, por el contrario, de economías de escala.

Boehm (1981), investigador pionero del tema, establece la existencia de diseconomías de escala, y se muestra taxativo sobre la forma de combatirlas: "... la más poderosa técnica de todas para reducir diseconomías de escala... es REDUCIR LA ESCALA".

1.1 Objeto

Se presenta un análisis de la existencia de economías de escala en proyectos de software, en el banco de datos ISBSG (International Software Benchmarking Standards Group), versión 11¹.

El resto de la comunicación se organiza en los siguientes apartados:

- Trabajos relacionados;
- Metodología;
- Resultados: Distribución de la variable tamaño; estudio de la relación entre el esfuerzo y el tamaño;
- Conclusiones.

2. Trabajos relacionados

2.1 Precisiones sobre el concepto de economías de escala

Las economías de escala en las empresas se estudian desde hace tiempo, situando los primeros pasos (Maital, 1995) en el siglo XIX, indicando que: “El historiador empresarial de Harvard Alfred Chandler Jr.,... establece los orígenes de las economías de escala en relación con... la Standard Oil Company de EE.UU., en 1882”. Por ello, su estudio constituye un tema constante en la bibliografía de Microeconomía, cuando explica el comportamiento de la producción de las empresas. Sin embargo, el que el origen del concepto se refiera a la empresa, mientras que el tema aquí considerado se refiera a un proyecto de software—no a la empresa que lo desarrolla—, recomienda tratar separadamente los dos ámbitos: empresa y proyecto.

2.1.1 Economías de escala en la empresa

Villamil (1971) indica:

... se puede decir que las economías de escala surgen cuando existe un decrecimiento de los costes medios a medida que aumenta la escala de producción, significando, a su vez, un mejor aprovechamiento de los inputs que se utilizan. Las deseconomías de escala se plasman en una elevación más que proporcional de los costes al incrementarse el producto final obtenido.

El planteamiento de las economías de escala se realiza en supuestos de largo plazo, cuyo significado es que todos los factores se pueden modificar, son todos variables; esto es, se puede modificar el tamaño de la empresa. Mientras que el significado de corto plazo es que hay algún factor fijo.

Frank (2005) señala que:

Cuando la función de producción tiene rendimientos decrecientes de escala, un aumento proporcional dado de la producción requiere un aumento proporcional mayor de todos los factores y, por lo tanto, un aumento proporcional mayor de los costes totales... El caso de los rendimientos decrecientes tiene la propiedad general de que da lugar a una curva de coste total a largo plazo (CTLP) ascendente y a unas curvas de coste medio a largo plazo (CMeLP) y coste marginal a largo plazo (CMALP) de pendiente positiva.

¹ Se trabaja con la versión del banco que se facilita a las Universidades.

Y en caso de rendimientos crecientes de escala, la curva de CTLP es descendente y las curvas de CMeLP y CMALP son de pendiente negativa.

Puede darse la situación de que haya una combinación de los dos aspectos: “Sin embargo,..., el grado de rendimientos de escala de una función de producción no tiene por qué ser el mismo en todos los niveles de producción”.

2.1.2 Economías de escala en un proyecto de software

El concepto de economías de escala en la empresa, por lo indicado, se refiere a cómo varían los costes medios de cada unidad de producto, en un planteamiento a largo plazo—esto es, que se puede modificar el tamaño de la empresa—. Mientras que el concepto usual de economía de escala en un proyecto de software se refiere a cómo varían los costes del proyecto en función de su tamaño.

El concepto de economías de escala en proyectos no hace referencia, en principio, a un aspecto que se explicita en el caso de la empresa: si el planteamiento es a corto o a largo plazo. Mientras que el concepto en proyectos hace referencia a un aspecto que no se explicita en la empresa: el tamaño del bien o servicio producido. Por ello, en la tabla 1 se esquematizan estas matizaciones que diferencian el concepto de economías de escala en la empresa y en los proyectos.

Tabla 1: Economías de escala en empresas y en proyectos

| | | Tamaño de la empresa (de desarrollo software) | |
|---------------------------------|---------|-----------------------------------------------|------------------------|
| | | Fijo (Corto plazo) | Variable (Largo plazo) |
| Tamaño del proyecto de software | Pequeño | A | C |
| | Grande | B | D |

La situación de corto plazo—que corresponde a los casos A y B de la tabla 1—es conocida como de rendimientos decrecientes en la producción y aparece en cualquier situación productiva, no siendo por tanto una excepción el caso del desarrollo de software. Por ejemplo, Krugman² y Wells (2006) sobre los rendimientos decrecientes de los programadores informáticos indican:

El concepto de rendimientos decrecientes de un factor productivo fue formulado por primera vez por los economistas del siglo XVIII... Sin embargo, la idea de los rendimientos decrecientes de un factor productivo puede aplicarse también a las modernas actividades económicas como, por ejemplo, el diseño de software. En 1975, Frederick P. Brooks Jr., un director de proyectos de IBM..., publicó un libro titulado *The Mythical Man-Month*... Brooks observó que multiplicar el número de programadores asignados a un proyecto no producía una reducción proporcional en el tiempo que llevaba escribir un programa informático... De hecho, por encima de un cierto número, añadir otro programador al proyecto incrementaría la duración de tal proyecto... Cada programador adicional aporta menos que el anterior y por encima de cierto nivel un programador más es contraproducente...

² Paul Krugman es Premio Nobel de Economía 2008.

La observación de Brooks, que se acaba de describir, corresponde a una función de producción con rendimientos decrecientes a corto plazo estudiada por la Microeconomía de siempre. Krugman y Wells (2006) siguen diciendo:

En otras palabras, la programación es una actividad sujeta a rendimientos decrecientes tan importantes que existe un punto a partir del cual más programadores tienen un producto marginal negativo. El origen de los rendimientos decrecientes se encuentra en la naturaleza de la función de producción del proyecto de programación: cada programador debe coordinar su trabajo con el de los demás programadores del proyecto, lo que provoca que cada persona emplee cada vez más tiempo en comunicarse con los otros miembros del proyecto a medida que aumenta el número de programadores. En otras palabras, si todo lo demás permanece constante, existen rendimientos decrecientes del trabajo.

Para terminar referiremos la opinión de Pressman (2006) de que existe un mito común que todavía creen muchos gestores responsables del esfuerzo del desarrollo del software: "Si nos retrasamos en la calendarización, siempre podemos incorporar más programadores y recuperarnos más adelante en el proyecto".

2.2 Investigaciones empíricas relacionadas

Los investigadores obtienen resultados, en principio, contrarios. Por ejemplo, Boehm (1981) y Boehm et al. (2000) se inclinan por diseconomías de escala, Walston y Felix (1977) por economías.

Banker y Kemerer (1989) realizan una importante aportación metodológica sobre la existencia de economías de escala.

Sobre la polémica de que los investigadores no se ponen de acuerdo, sobre si se dan economías o diseconomías de escala en proyectos de software, Kitchenham (2002) señala que se debe a diversos aspectos, entre ellos los siguientes:

- las investigaciones se basan en distintos bancos de datos;
- la elección del test de hipótesis;
- el uso de funciones de producción;
- el análisis de sensibilidad;
- las técnicas estadísticas.

Premraj et al. (2005), trabajando con otro banco de datos, no encuentran evidencias de diseconomías de escala, aunque los resultados son diferentes en significación para proyectos de nuevo desarrollo y para proyectos de mantenimiento. Y recomiendan que se realicen replicaciones de su estudio, lo que se considera una de las justificaciones del análisis empírico que a continuación se presenta.

Para terminar, indicamos que existen otras líneas de investigación, con nuevas metodologías de tratamiento de datos, redes neuronales, algoritmo MARS (Multivariate Adaptive Regression Spline), etc. (Villanueva, 2005; Alba, 2007), que esperamos aplicar en futuras investigaciones.

3. Metodología

3.1 Descripción del banco de datos

Para llevar a cabo una investigación cuantitativa en proyectos informáticos, con un análisis empírico de datos, es importante contar con un banco de datos fiable y que esté formado por una muestra relevante para el estudio (Maciel et al., 2009).

Como base empírica para investigación sobre su gestión, se utiliza una fuente disponible, el banco de datos ISBSG en su versión 11 que se facilita a las Universidades³, que incluye más de 5.000 proyectos de software de diversos países, con una gran variedad de tipos (Maciel et al., 2009).

El tratamiento estadístico de los datos se ha realizado utilizando SPSS versión 17.

3.2 Criterios de selección de los proyectos del banco de datos

Los criterios de selección utilizados vienen determinados por la necesidad de utilizar datos de calidad, de referencia temporal de los proyectos y de coherencia a la hora de comparar el tamaño en puntos de función y el esfuerzo en horas/persona⁴.

3.3 Distribución de los distintos métodos de medición del tamaño

ISBSG clasifica la calidad de la medición funcional en cuatro niveles (A, B, C y D) decrecientes de calidad. La tabla 2 presenta la distribución de los distintos métodos de medición del tamaño en función de la calidad de la medición del tamaño funcional, para la totalidad de los proyectos del banco de datos, es decir para los 5052 proyectos de la versión 11.

Tabla 2: Distribución de los distintos métodos de medición del tamaño

| Método | Calidad de la medición del tamaño funcional | | | | | Total |
|----------------|---------------------------------------------|------|-----|-----|---|-------|
| | s.d. | A | B | C | D | |
| Albrecht | 0 | 2 | 0 | 0 | 0 | 2 |
| Automated | 0 | 2 | 1 | 0 | 1 | 4 |
| Backfired | 0 | 0 | 0 | 7 | 1 | 8 |
| COSMIC | 345 | 0 | 0 | 0 | 0 | 345 |
| Dreger | 0 | 0 | 0 | 4 | 6 | 10 |
| Feature Points | 0 | 2 | 0 | 0 | 0 | 2 |
| FiSMA | 0 | 0 | 156 | 340 | 0 | 496 |
| Fuzzy Logic | 2 | 0 | 0 | 0 | 0 | 2 |
| IFPUG | 6 | 2270 | 765 | 755 | 3 | 3799 |
| In-house | 0 | 1 | 0 | 0 | 0 | 1 |
| LOC | 186 | 0 | 0 | 0 | 0 | 186 |

³ El banco de datos ISBSG en su versión 11 que se facilita a las Universidades se describe con detalle en otra comunicación al Congreso: “Evolución reciente del banco de datos ISBSG (International Software Benchmarking Standards Group) de proyectos de software”.

⁴ El detalle del filtrado realizado puede obtenerse en otra comunicación al Congreso: “Evolución de la productividad física de los proyectos de software del banco de datos ISBSG”.

| | | | | | | |
|-------------------|-----|------|------|------|----|------|
| Mark II | 0 | 16 | 4 | 15 | 1 | 36 |
| NESMA | 0 | 34 | 115 | 6 | 0 | 155 |
| Retrofitted | 0 | 0 | 0 | 0 | 2 | 2 |
| System Components | 1 | 0 | 0 | 0 | 0 | 1 |
| Unknown | 0 | 1 | 0 | 1 | 1 | 3 |
| Total | 541 | 2328 | 1041 | 1128 | 15 | 5052 |

Nota: s.d. significa sin dato.

Estamos trabajando con puntos de función y no con cualquier otro tipo de medida del tamaño, ya que es necesario que el método de medida sea el mismo para poder hacer comparaciones. Como método de medida del tamaño en puntos de función se ha elegido IFPUG (International Function Point Users Group) por ser el más representativo. Por otro lado, dentro de IFPUG existen varias versiones. Parece que a partir de la versión 4 (1994) y posteriores es posible realizar una comparación de los tamaños medidos según las distintas versiones. Y se han considerado únicamente los proyectos con calidad de medición funcional de nivel A.

4. Resultados

4.1 Distribución de la variable tamaño

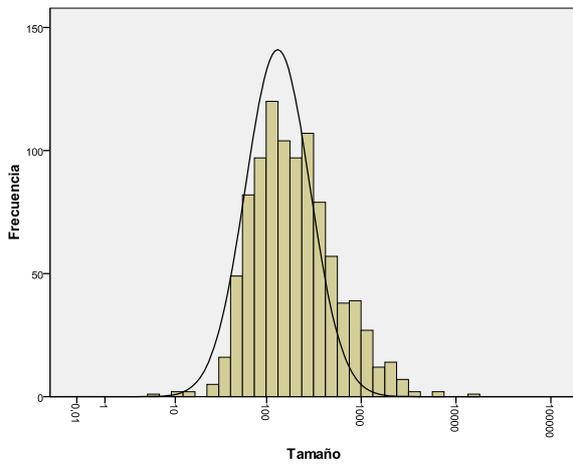
Para el siguiente análisis contamos con un total de 960 proyectos, puesto que las condiciones sobre la variable esfuerzo no son por ahora indispensables.

La distribución de la variable tamaño presenta una media de 355,6 puntos de función IFPUG y una desviación típica de 680,561. Entre los 960 proyectos analizados, destaca uno con 13.580 puntos de función, el de tamaño mayor, precedido de lejos por uno con 7.400 puntos de función.

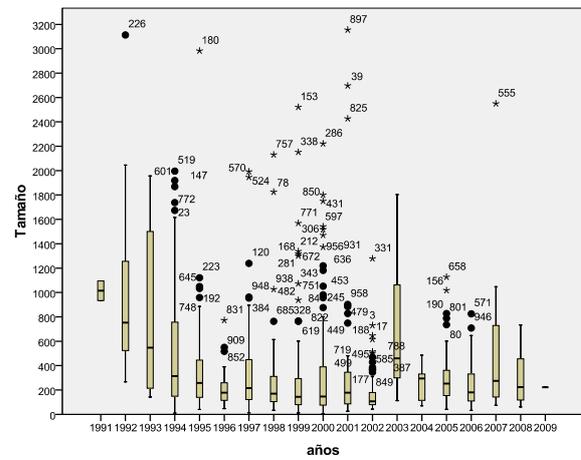
Tras una transformación logarítmica sobre la variable en cuestión, la figura 1.a muestra una distribución aproximadamente normal. Esta transformación podrá permitir más adelante hacer uso de esta variable en modelos de regresión lineal, por ejemplo.

Utilizando un diagrama de caja y bigotes, la figura 1.b presenta la evolución del tamaño de los proyectos en función del tiempo para los 960 seleccionados. Aquí hay que indicar que el horizonte temporal contemplado comprende 19 años, de 1991 a 2009. Sin embargo la distribución de proyectos por año en esta franja no es uniforme. Así, ciertos comportamientos pueden explicarse realizando simplemente un análisis de frecuencia de los proyectos por año. Por ejemplo, el pico de 2003 puede deberse a que sólo contamos con una muestra de 5 proyectos para ese año, lo cual no sería en absoluto significativo para ciertos análisis estadísticos (Pulido y Sánchez, 2001; Maxwell, 2002).

Figura 1: Distribución de la variable tamaño en puntos de función IFPUG



a) En escala logarítmica



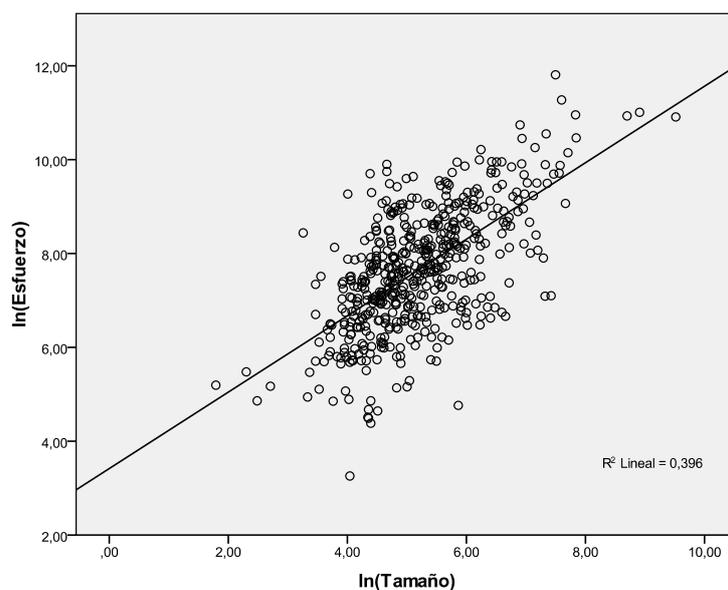
b) En función del tiempo

4.2 Estudio de la relación entre el esfuerzo y el tamaño

Al introducir la variable esfuerzo, medida en horas/persona, y las condiciones que se le exigen, en el siguiente análisis reducimos la muestra de 960 a un total de 521 proyectos que cumplen todas condiciones de selección.

Una vez comprobado que la variable esfuerzo presenta también una distribución bastante normal tras una transformación logarítmica, a continuación presentamos en la figura 2 el diagrama de dispersión con la línea de ajuste total.

Figura 2: Diagrama de dispersión esfuerzo-tamaño



Resumiendo el modelo, el coeficiente de correlación de Pearson, que en este caso es de 0,629, indica una determinada correlación entre las dos variables y el coeficiente de determinación que el 39,6% de la variabilidad en el esfuerzo se explica por el tamaño del proyecto.

Por lo tanto existen otras variables que modifican y explican la variabilidad del esfuerzo de los proyectos. En futuros trabajos, la introducción de más variables con técnicas de análisis multivariado nos permitirá identificar la importancia que otras variables pueden tener sobre el esfuerzo.

El modelo planteado responde a la ecuación:

$$\ln(\text{Esfuerzo}) = 3,416 + 0,815 * \ln(\text{Tamaño}) \quad (1)$$

Lo que equivale, eliminando la transformación logarítmica realizada, a la siguiente ecuación:

$$\text{Esfuerzo} = 30,45 * \text{Tamaño}^{0,815} \quad (2)$$

En la tabla 3 se presenta la información sobre los coeficientes del modelo de regresión. Para el exponente de la ecuación, cuyo valor es $B = 0.815$ con $sig. = 0.000$, se obtiene un intervalo de confianza de 95% entre 0,728 y 0,901. Es decir que es distinto de cero, cuya interpretación es que estamos ante un resultado significativo de economías de escala.

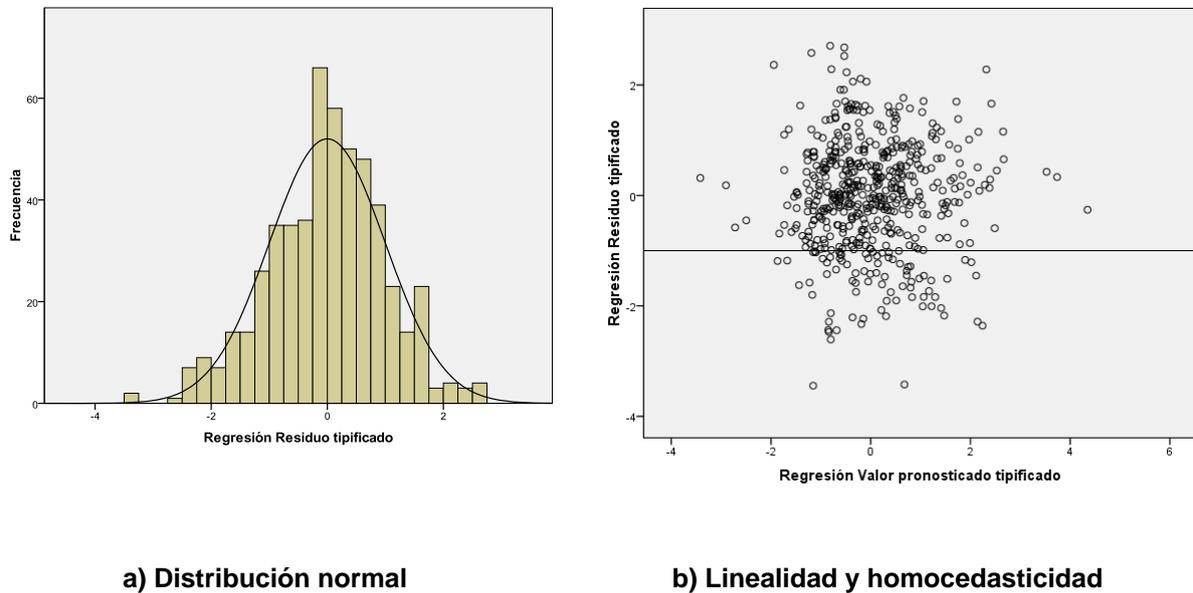
Tabla 3: Coeficientes del modelo de regresión

| Modelo | Coeficientes no estandarizados | | Coeficientes tipificados | t | Sig. | Intervalo de confianza de 95,0% para B | |
|------------|--------------------------------|------------|--------------------------|--------|------|----------------------------------------|-----------------|
| | B | Error típ. | Beta | | | Límite inferior | Límite superior |
| Constante | 3,416 | ,233 | | 14,635 | ,000 | 2,957 | 3,874 |
| Ln(Tamaño) | ,815 | ,044 | ,629 | 18,445 | ,000 | ,728 | ,901 |

4.2.1 Validación del modelo

En los modelos lineales, se asume que los residuos presentan una distribución normal. En efecto la figura 3.a muestra un patrón de los residuos conforme a una distribución normal.

Figura 3: Distribución de los residuos

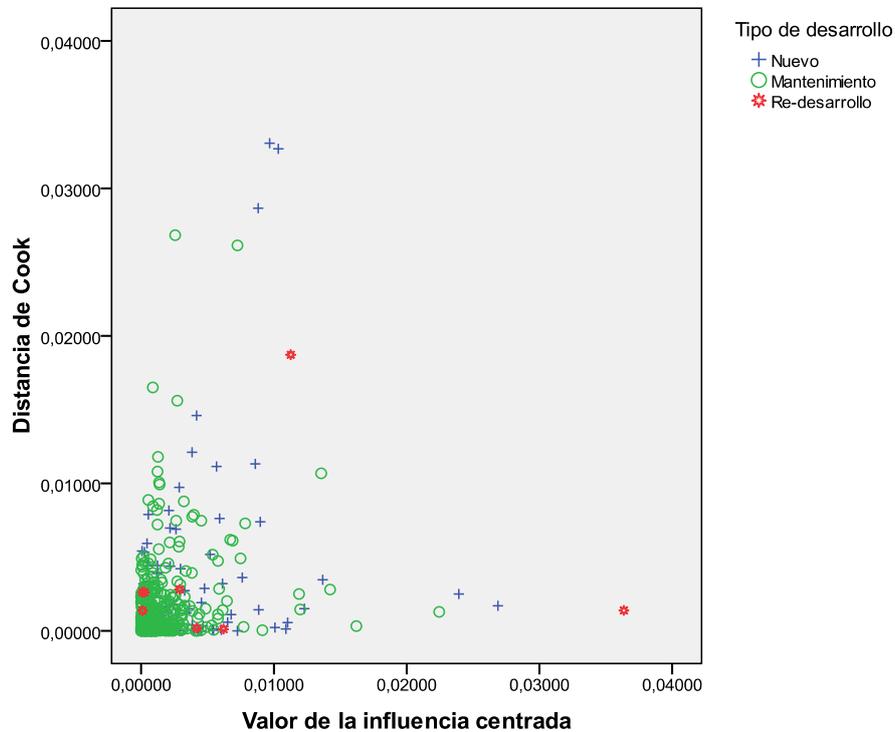


Además, debe cumplirse la hipótesis de linealidad y de varianza constante de los residuos, es decir la condición de homocedasticidad. Como puede contrastarse al representar los residuos tipificados frente a los valores pronosticados tipificados en la figura 3.b, la nube de puntos no presenta una tendencia o patrón de comportamiento, por lo tanto ambas hipótesis quedan verificadas.

Con el estadístico de Durbin-Watson estudiaremos la autocorrelación de los residuos. En nuestro caso vale 1,832, por lo que asumiremos que los residuos están relativamente incorrelados.

Finalmente en la figura 4, el diagrama de dispersión identifica los puntos de influencia. Los puntos con una mayor distancia de Cook corresponden a tres proyectos de nuevo desarrollo, sin embargo no poseen un alto leverage. Así que, aunque añaden mucha variabilidad a las estimaciones de la regresión, no afectan a la pendiente de la ecuación de regresión (Romero y Zúñica, 2010; Peña, 2008).

Figura 4: Puntos de influencia



5. Conclusiones

Como se indica en las investigaciones de referencia, es importante realizar replicaciones de las mismas, ya que no hay consenso científico sobre los resultados. Por ello, se ha puesto especial cuidado en especificar todo el proceso seguido en este estudio, con la finalidad de que pueda ser replicado.

Los resultados obtenidos suponen una modesta aportación a la cuestión que se debate, sobre la que seguimos trabajando, incorporando nuevas variables y metodologías que permitan mejorar este trabajo.

Los autores agradecen a los revisores las oportunas recomendaciones que realizaron sobre la propuesta de comunicación.

6. Referencias

- Alba, C. (2008). *Predicción y clasificación del nivel de riesgo en proyectos de sistemas de información*. Unpublished doctoral dissertation, Universidad de Oviedo, España.
- Álvarez, P. A. (2001). Concepto y medición de la eficiencia productiva. En Álvarez, P. A. *La medición de la eficiencia y la productividad (capítulo 1)*. Madrid: Pirámide.
- Banker, R. D., & Kemerer, C. F. (1989). Scale economies in new software development. *IEEE Transactions on Software Engineering*, SE-15 (10) 1199-1205.
- Boehm, B., Abts, C., Brown, A. W., Chulani, S., Clark, B. K., Horowitz, K., Madachy, R., Reifer, D., & Steece, B. (2000). *Software Cost Estimation with COCOMO II*. Prentice Hall.
- Boehm, B. W. (1981). *Software Engineering Economics*. Prentice-Hall.
- Brooks, F. P. (1975). *The Mythical Man-Month*. Addison Wesley.
- Frank, R. H. (2005). *Microeconomía y Conducta*. McGraw-Hill.
- Internacional Software Benchmarking Standards Group (ISBSG). www.isbsg.org (27-03-2010).
- Kitchenham, B. (2002). The question of scale economies in software-why cannot researchers agree? *Information and Software Technology* (44.1: 13-24).
- Krugman, P., & Wells, R. (2006). *Introducción a la Microeconomía*. Reverté.
- Maciel, J. P., Fernández-Diego, M., Vázquez-Barrachina, E., Neves-Silva, A., & Torralba-Martínez, J. M. (2009). Fuente de datos para investigación sobre Gestión de Proyectos Informáticos: Repositorio ISBSG - International Software Benchmarking Standards Group. En *I Congreso Iberoamericano SOCOTE*. Universidad Politécnica de Valencia, España.
- Maital, S. (1995). *Coste, Precio, Valor. Cómo gestionar mejor el triángulo del beneficio*. Deusto.
- Maxwell, K. (2002). *Applied statistics for software managers*. Prentice-Hall.
- Peña, D. (2008). *Fundamentos de Estadística*. Alianza Editorial.
- Premraj, R., Shepperd, M., Kitchenham, B., & Forselius, P. (2005). An Empirical Analysis of Software Productivity over time. En *11th IEEE International Software Metrics Symposium, IEEE*. Como, Italy.
- Pressman, R. (2006). *Ingeniería del Software. Un enfoque práctico*. McGraw-Hill.
- Pulido, A., & Pérez, J. (2001). *Modelos Económicos*. Madrid: Pirámide.
- Romero, R., & Zúñica, L. (2010). *Métodos Estadísticos en Ingeniería*. Valencia, España: Editorial de la Universidad Politécnica de Valencia.

Villamil, A. (1979). *Teoría y Política de las Economías de Escala. Una aplicación al caso español*. Ariel.

Villanueva, B. J. (2005). *Estimación de costes y plazos en proyectos de sistemas de información*. Unpublished doctoral dissertation, Universidad de Oviedo, España.

Walston, C. E., & Felix, C. P. (1977). A method of programming measurement and estimation. *IBM systems Journal* (vol. 16, pp 54-73).

Correspondencia (Para más información contacte con):

Marta Fernández-Diego
Universidad Politécnica de Valencia
Departamento de Organización de Empresas
Camino de Vera s/n, 46022 Valencia, Spain
Phone: +34 96 387 76 85
E-mail : marferdi@omp.upv.es