

# Uso del análisis ROC para anticipar la conveniencia de trazar proyectos de software

Juan Giró, Juan Vázquez, Brenda Meloni, Leticia Constable y Andrea Jornet

Departamento de Ingeniería en Sistemas de Información  
Facultad Regional Córdoba, Universidad Tecnológica Nacional  
Maestro López esq. Cruz Roja Argentina, Córdoba  
{juanfgiro, jcvazquez, bemeloni, leticiaconstable}@gmail.com, andreaornet@yahoo.com.ar

**Resumen.** El desarrollo de modelos que permitan anticipar la conveniencia de aplicar procesos de trazabilidad a proyectos de software está inspirado en la escasez de evidencias de que los progresos en este campo son efectivamente aprovechados por la industria. Así surgió la idea de identificar los factores de mayor impacto en el éxito de los procesos de trazabilidad y de estudiar modelos que posibilitan hacer predicciones en este campo a partir de esos factores. En este trabajo se presenta un modelo de predicción basado en el Análisis ROC, que permite obtener una buena estimación con poco esfuerzo, tomando como base una técnica que por el momento ha tenido poca difusión en la ingeniería de software.

**Keywords:** ingeniería de software, análisis ROC, trazabilidad de requerimientos.

## 1 Introducción

Al hablarse de trazabilidad de requerimientos en un proyecto de desarrollo de software se está haciendo referencia a una gestión que vincula las numerosas etapas de su ciclo de vida, brindando la necesaria garantía de coherencia, completitud y corrección al software producido y posibilitando su eficaz mantenimiento correctivo y preventivo.

La importancia y trascendencia de la trazabilidad de requerimientos como soporte del proceso de desarrollo de software hoy no se discute [1]. Sin embargo, en general las soluciones no llegan a ser efectivamente aplicadas en la industria del software [2], no pudiendo confirmarse las cualidades que pregonan sus autores ni los beneficios que aportarían tanto a los proyectos como a sus productos.

Además, al analizarse los temas de estudio en el campo de la trazabilidad, se comprueba que en su mayor parte están orientados a desarrollar nuevas metodologías y herramientas, y un esfuerzo mucho menor está destinado a estudiar la aplicación de las mismas en la industria y los motivos de las dificultades ya señaladas. Aquí cabe comentar que los escasos documentos destinados a analizar las causas de los dificultades de la trazabilidad hacen un abordaje cualitativo del problema [2][3], que indudablemente es útil, pero que la mayoría de las veces resulta demasiado general.

Indagando con más detalle en las experiencias desfavorables del campo de la trazabilidad se comprueba que pueden ser reunidas en tres grupos: *i*) las que fueron

abandonadas o no cubrieron las expectativas desde un punto de vista técnico, *ii*) las que fueron técnicamente exitosas pero el costo de su implementación fue mayor que el beneficio obtenido y *iii*) las que combinan ambas consecuencias negativas, es decir un resultado pobre o nulo a un elevado costo.

Por lo expuesto, surge la presunción de que no es fortuito que ciertos proyectos puedan ser exitosamente trazados y otros no, lo que lleva a pensar en una combinación de condiciones objetivas que conducen a uno u otro resultado. Es decir, debería ser posible reconocer factores determinantes que anticipen que un proyecto de desarrollo de software podrá ser efectivamente respaldado por un sistema de trazabilidad.

A partir de este razonamiento se plantea la hipótesis de que existen factores que condicionan el éxito de un proceso de trazabilidad y que es factible identificarlos. En base a estos factores será posible anticipar la conveniencia de aplicar trazabilidad a cierto proyecto concreto en ciertas condiciones y en un momento dado. Estos factores pueden ser cambiantes, por estar vinculados al “estado del arte” de la ingeniería de software y a la tecnología computacional disponible en cada momento.

Estas ideas inspiran la actividad que viene siendo cumplida en el proyecto “Aseguramiento de la Trazabilidad en Proyectos de Desarrollo de Sistemas de Software” [4], que persigue demostrar la hipótesis enunciada. En el marco de este proyecto se elaboraron dos documentos [5][6], que tuvieron por finalidad describir los factores propuestos para representar la efectividad de la trazabilidad en los proyectos de software y proponer modelos que reúnan estos factores para anticipar los resultados que pueden esperarse en cada caso. El último trabajo [6] presentó un modelo perfeccionado a partir de las experiencias realizadas y enseñanzas obtenidas con un modelo anterior [5].

La organización de este documento es la siguiente: en la sección 2 se resumen los factores de trazabilidad elegidos y los modelos propuestos, en la sección 3 se describe un nuevo modelo basado en el análisis ROC y se discuten los resultados obtenidos y en la Sección 4 se presentan las conclusiones de este trabajo y actividades futuras.

## **2 Antecedentes de modelos propuestos**

Las tres entidades involucradas en un proceso de desarrollo son: *a*) el producto software, *b*) el proyecto en el que se lo construye y *c*) la organización a cargo de las tareas, y los factores buscados estarán asociados a dimensiones de estas tres entidades. En el trabajo anterior ya citado [6] se presentaron y justificaron ocho factores elegidos para tal fin, que fueron agrupados según su pertenencia a las tres entidades y que son resumidos en el apartado siguiente. Al seleccionárselos se tuvo en cuenta que debían: *i*) ser cuantificables, *ii*) resultar familiares para facilitar su relevamiento, *iii*) ser ortogonales entre sí, *iv*) admitir valores en el rango [0..5] mediante métricas apropiadas y *v*) sus valores crecientes favorecer la trazabilidad.

### **2.1 Factores de la trazabilidad y sus métricas**

Los ocho factores propuestos para predecir la posibilidad de realizar la trazabilidad de requerimientos en proyectos de desarrollo de software son los siguientes:

- **Tamaño del producto (*t*)**

Para definir la métrica aplicable al factor *tamaño* (*t*) se adoptan puntos de función **PF** (ISO/IEC 20926, IFPUG 4.3.1) en el rango [100, 1000], para la representación de sistemas “chicos” y “muy grandes” respectivamente. Se expresa entonces:

$$t = \begin{cases} 5(PF/1000) & \text{si } PF \leq 1000 \\ 5 & \text{si } PF > 1000 \end{cases} \quad (1)$$

Nótese que, tal como está definido, todo proyecto con un tamaño mayor de 1000 puntos de función quedará representado por un factor  $t = 5$ , es decir que  $0,5 \leq t \leq 5$ .

- **Vigencia del producto (*v*)**

En la definición de la métrica del factor *vigencia* (*v*) se adopta para la vida útil **VU** un rango [0,5; 10] años, con lo que se tiene:

$$v = \begin{cases} 5(VU/10) & \text{si } VU \leq 10 \\ 5 & \text{si } VU > 10 \end{cases} \quad (2)$$

Es decir que  $0,25 \leq v \leq 5$ . Tal como esta expresado, la influencia del factor vigencia en la trazabilidad es proporcional a la vida útil del sistema hasta los diez años, a partir de los cuales el factor se mantendrá constante en el valor máximo de cinco.

- **Reutilización del producto (*r*)**

Se asume que en la trazabilidad el factor *reutilización* (*r*) tiene un comportamiento polinomial de segundo grado con respecto al porcentaje de futura reutilización **RE** [%], es decir privilegia los porcentajes mayores. La métrica propuesta es la siguiente:

$$r = \begin{cases} 1 + 0,025(RE + 0,0125RE^2) & \text{si } RE \leq 80 \% \\ 5 & \text{si } RE > 80 \% \end{cases} \quad (3)$$

- **Confiabilidad del producto (*c*)**

Para determinar el factor de *confiabilidad* (*c*) se toma como base un indicador, denominado **CO**, que representa la confiabilidad requerida del sistema y que es definido en el rango [0..5] con la escala siguiente: 0) *no significativa*, 1) *baja*, 2) *media*, 3) *alta*, 4) *muy alta* y 5) *absoluta*. Para definir la métrica *confiabilidad* (*c*) se adopta la expresión:

$$c = 1 + 0,16CO^2 \quad \text{para } CO = [0..5] \quad (4)$$

- **Plazo de ejecución del proyecto (*p*)**

Se asume que la duración de un proyecto informático **DP** está en el rango de [0; 5] años y el factor *plazo de ejecución* (*p*) queda directamente definido como:

$$p = DP \quad \text{para } DP = [0;5] \quad (5)$$

- **Equipo de proyecto (*e*)**

Para reflejar la efectividad **EF** del equipo se adopta una escala en el rango [1..5], que se define: 1) *Pobre*, 2) *Baja*, 3) *Media*, 4) *Alta* y 5) *Muy alta*. El factor *equipo de proyecto* (*e*) es asignado directamente:

$$e = EF \quad \text{para } EF = [1..5] \quad (6)$$

### - Madurez de la organización (*m*)

El factor *madurez (m)* de una organización es clasificado según la escala adoptada por CMMI del SEI (Ver. 1.3, TR-033/2010), que para el modelo de representación por etapas reconoce cinco niveles de madurez, que son: 1) *Inicial*, 2) *Gestionado*, 3) *Definido*, 4) *Cuantitativamente gestionado* y 5) *Optimizado*. El factor se define:

$$m = \text{Nivel de madurez CMMI } [1..5] \quad (7)$$

### - Dependencia de la organización (*d*)

El factor *dependencia (d)* define nivel de autonomía *NA* que tiene una organización para operar y tomar decisiones. Para ello se adopta la siguiente escala de clasificación: 1) *absolutamente libre*, 2) *adhesión a estándares propios*, 3) *dependencia de estándares de clientes*, 4) *dependencia de estándares de casa matriz* y 5) *dependencia de estándares de clientes y casa matriz*. Se define:

$$d = NA \quad \text{para } NA = [1..5] \quad (8)$$

## 2.2 Modelos Básico y Avanzado

Se propusieron dos modelos que tienen como entradas los ocho factores ya descritos. En ambos casos se trata de modelos dicotómicos destinados a clasificar anticipadamente los proyectos en “trazables” y “no trazables”, en los que sus parámetros son ajustados con información obtenida en la industria del software.

El primer modelo propuesto, denominado básico, equivale a reconocer que cada uno de los ocho factores incide de manera independiente sobre el diagnóstico buscado, y está destinado a un rápido reconocimiento de las expectativas que ofrece la trazabilidad de un proyecto. Para visualizar el modelo se recurre a un “diagrama radar” [6], que es un esquema plano con una cantidad de ejes radiales igual a la dimensión del modelo, en este caso ocho. En la Figura 1 se muestra uno de tales diagramas, donde los polígonos corresponden a: *a*) un proyecto trazable, *b*) un proyecto no trazable, *c*) un proyecto atípico y *d*) la “zona gris” que representa la frontera discriminante entre ambos casos.

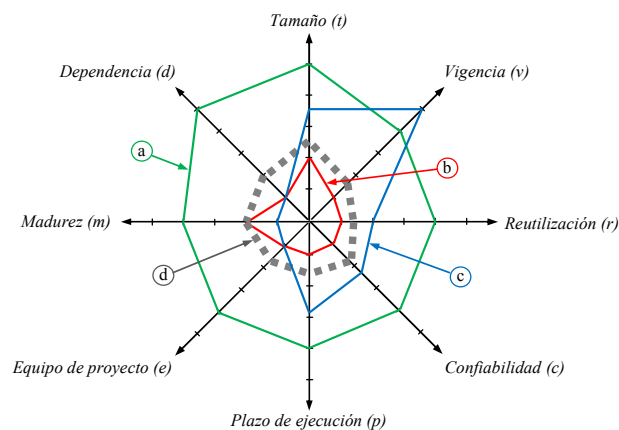


Figura 1. Representación de la trazabilidad de proyectos en un diagrama de radar

Los proyectos atípicos deben ser objeto de un análisis particular por admitir diversas interpretaciones, tales como: *i)* está fuera del alcance de este tipo de modelo, *ii)* el modelo lo cubre pero todavía carece de la sensibilidad requerida o *iii)* los valores asignados a los factores que definen este caso presentan inconsistencias y el modelo estaría advirtiendo esta circunstancia. Un proyecto bien planteado que caiga sobre la “zona gris” de la frontera (caso *ii)* estaría evidenciando que el modelo no alcanzó todavía suficiente sensibilidad como para clasificarlo correctamente.

El segundo modelo propuesto, que fue denominado avanzado, considera los ocho factores en forma combinada, lo que implica reconocer la existencia de una función altamente compleja cuyos argumentos son precisamente los ocho factores seleccionados. La expresión está destinada a clasificar los casos en “trazables” ( $\xi=1$ ) y “no trazables” ( $\xi=0$ ), que en forma general es expresada:

$$\xi = f(t, v, r, c, p, e, m, d) ; \xi \in [0,1] \quad (9)$$

Por conocerse los argumentos de la función propuesta  $\xi$ , pero no la expresión matemática capaz de reproducir un comportamiento acorde a las evidencias provenientes de la industria, se implementó el modelo en base a una red neuronal multicapa de perceptrones [6], cuyo esquema es representado en la Figura 2.

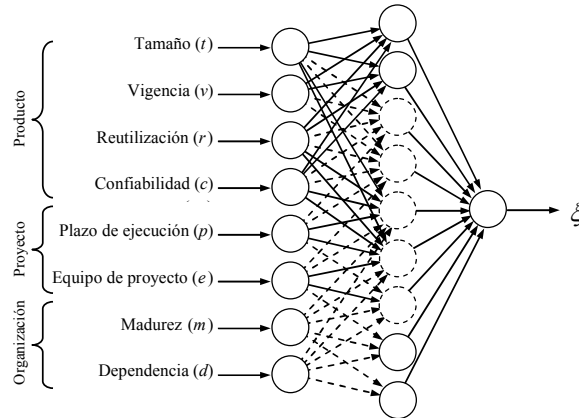


Figura 2. Modelo multicapa de perceptrones para la predicción de la trazabilidad de proyectos

### 3 Nuevo Modelo basado en el Análisis ROC

Para visualizar las poblaciones de datos utilizadas en las pruebas de los modelos se asignó una variable a cada una de las tres entidades involucradas: el *producto* ( $\eta_1$ ), el *proyecto* ( $\eta_2$ ) y la *organización* ( $\eta_3$ ). Al calcular el módulo de la resultante de los factores de cada entidad se tuvo en cuenta su ortogonalidad, lo que permitió reducir de ocho a tres la dimensión del problema:

$$\eta_1 = \sqrt{(t^2 + v^2 + r^2 + c^2)} \quad (\text{producto}) \quad (10)$$

$$\eta_2 = \sqrt{(p^2 + e^2)} \quad (\text{proceso}) \quad (11)$$

$$\eta_3 = \sqrt{(m^2 + d^2)} \quad (\text{organización}) \quad (12)$$

A partir de ésta reducción de dimensiones surgió la idea de explorar la conveniencia de utilizar el módulo de la resultante de los ocho factores, denominado “ $\rho$ ”, como parámetro representativo o “indicador” de cada caso considerado:

$$\rho = \sqrt{(\eta_1^2 + \eta_2^2 + \eta_3^2)} = \sqrt{(t^2 + v^2 + r^2 + c^2 + p^2 + e^2 + m^2 + d^2)} \quad (13)$$

El indicador “ $\rho$ ” representa el radio de una fracción de casquete esférico en el hiperespacio de ocho dimensiones, y el objetivo es determinar el valor más apropiado que separe de la mejor forma las poblaciones de los proyectos no trazables de los trazables. Para esto se recurre al análisis ROC, que se describe a continuación.

Debe observarse que el modelo básico descrito con anterioridad equivale a un poliedro convexo que separa ambas poblaciones, donde cada una de sus ocho caras es normal al eje de un factor. En el otro extremo, el modelo avanzado puede visualizarse como una hipersuperficie compleja que cumple la misma función con más precisión, por lo que el casquete esférico representa un modelo intermedio entre ambos.

### 3.1 Conceptos del Análisis ROC

El análisis ROC es una técnica destinada a evaluar clasificadores dicotómicos y tiene dos finalidades: *i*) poder elegir objetivamente el mejor entre varios modelos de clasificación y *ii*) optimizar la sintonía del modelo elegido. Esta técnica fue desarrollada durante la segunda guerra mundial para medir la eficacia en la detección de objetos enemigos en pantallas de radar. Luego se la trasladó al ámbito civil y fue progresivamente aplicada a la toma de decisiones clínicas en medicina, diagnóstico por imágenes y psicología. Posteriormente ganó popularidad en campos muy variados [7] [8] tales como farmacología, bioingeniería, aprendizaje automático y minería de datos. Mucho más recientemente comenzó a usarse en la evaluación de atributos de calidad de la ingeniería de software y sus métricas [9]. La poca difusión del análisis ROC en el ámbito de la ingeniería de software justifica una breve presentación de este tema.

En este tipo de análisis la exactitud de un diagnóstico se expresa a través de dos parámetros: la *sensibilidad* y la *especificidad*. La *sensibilidad* es la probabilidad de clasificar correctamente una muestra cuya condición real ha sido definida como positivo, y queda representada como la fracción de verdaderos positivos (FVP). Por el contrario, la *especificidad* es la probabilidad de clasificar correctamente una muestra cuyo estado real ha sido definida como negativo. Su valor se obtiene de restar a uno la fracción de falsos positivos (FFP). En la Tabla 1 se resumen estos parámetros.

Tabla 1: Resultados obtenidos con un clasificador y definición de parámetros

Modelo Clasificador		Condición real		
		Positivos	Negativos	
Resultados del modelo	Positivos	Verdaderos positivos (VP)	Falsos positivos (FP)	$Sensibilidad = FVP = \frac{VP}{VP + FN} \quad (14)$
	Negativos	Falsos Negativos (FN)	Verdaderos negativos (VN)	$Especificidad = FVN = \frac{VN}{VN + FP} \quad (15)$ $Especificidad = FVN = 1 - FFP \quad (16)$ $Exactitud = \frac{VP + VN}{VP + FN + VN + FP} \quad (17)$

A partir de estos parámetros la curva ROC (Figura 3.b) representa globalmente el desempeño de un modelo clasificador, es necesariamente creciente y refleja el compromiso existente entre *sensibilidad* y *especificidad*. En efecto, si para obtener mayor *sensibilidad* se modifica el valor de corte  $\rho_c$  entre las poblaciones de las muestras negativas y positivas (Figura 3.a), se lo hace a expensas de disminuir al mismo tiempo la *especificidad*. Esto obliga a seleccionar dicho valor de corte con mucho cuidado y presupone la elección apropiada del indicador que representa el problema estudiado, dependiendo de esta elección la menor superposición posible de ambas poblaciones.

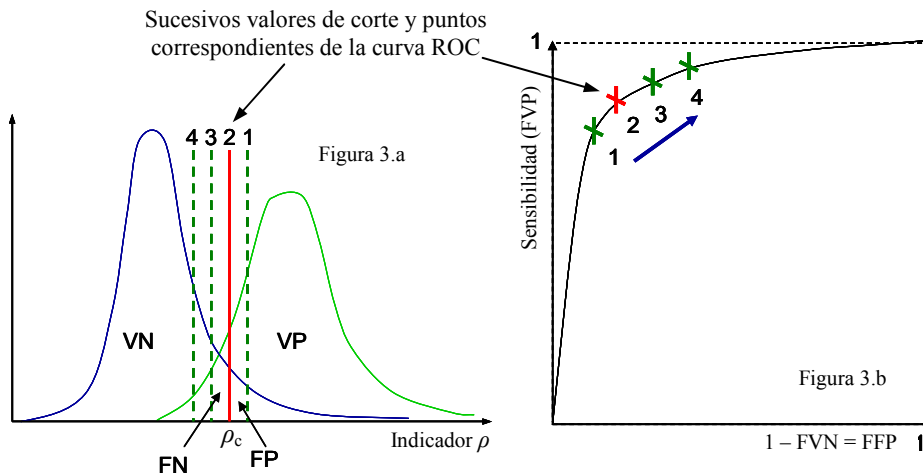


Figura 3: a) Curvas de distribución de resultados negativos-positivos y sucesivos valores de corte  $\rho_c$  para separar ambas poblaciones y b) Puntos definidos sobre la curva ROC a partir de los correspondientes valores de corte sobre las curvas de distribución.

Si el indicador  $\rho$  del modelo no fuese capaz de discriminar entre los dos grupos, es decir las poblaciones están superpuestas, la curva ROC sería una diagonal que une los vértices inferior izquierdo y superior derecho del gráfico, como muestra la Figura 4.

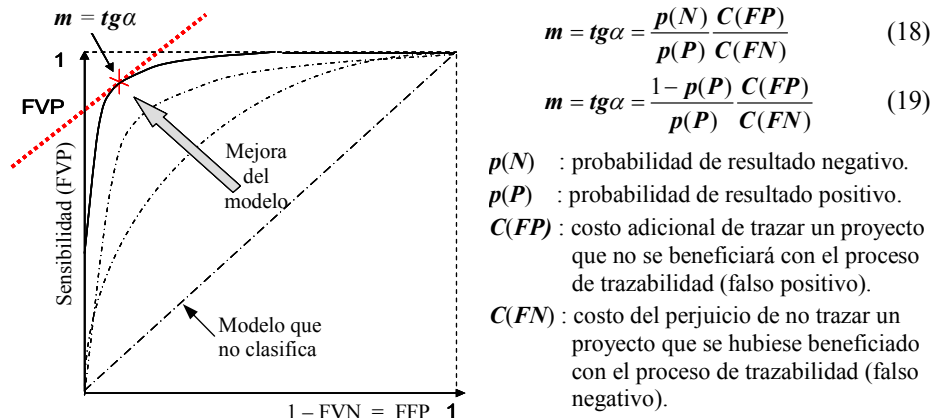


Figura 4: Curvas ROC de distintos modelos, que encierran mayor área al tratarse de mejores clasificadores, e identificación del punto de pendiente  $m$  que es el mejor valor de corte.

La exactitud del modelo aumenta a medida que la curva se desplaza desde la diagonal hacia el vértice superior izquierdo. Por este motivo para comparar dos clasificadores se utilizan las áreas encerradas bajo la curva ROC y el mejor es el que encierra la mayor área. Una vez identificado el mejor clasificador, su ajuste es realizado seleccionando el punto sobre la curva que tenga una pendiente  $m$ , como muestra la Figura 4.

### 3.2 Caso de estudio y discusión de resultados

A efectos de ilustrar sobre la aplicación del modelo propuesto se utilizó un lote de 102 casos conteniendo 55 proyectos trazados exitosamente (positivos) y 47 proyectos con resultados negativos. El lote de datos está compuesto con resultados relevados de la industria, casos reconstruidos a partir de informes incompletos y testimonios obtenidos de la literatura. La finalidad de este lote de datos fue conformar un caso de estudio que permitiera comprobar el desempeño del modelo ROC propuesto.

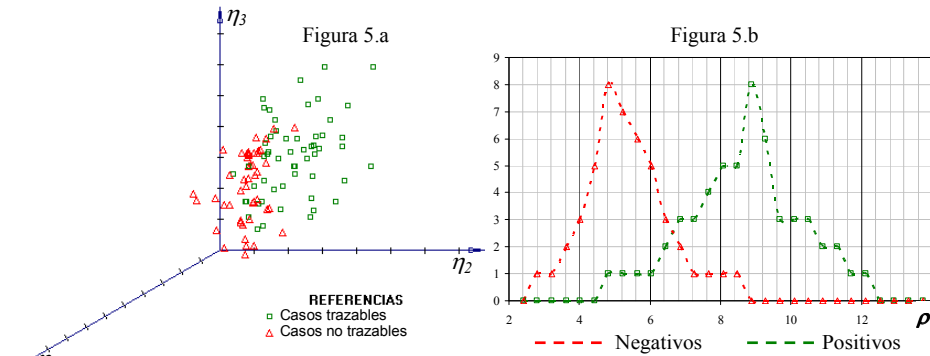


Figura 5: a) Representación de la población del lote de datos del caso de estudio y b) Curvas de distribución de la población de datos en función de “ $\rho$ ”

En la Figura 5.a se muestra la población del caso de estudio representada sobre un sistema de ejes  $\eta_1, \eta_2, \eta_3$  y en la Figura 5.b se presentan sus curvas de distribución de los casos negativos y positivos, con el indicador  $\rho$  en el eje de las abscisas. Luego, en la Figura 6.a se presenta la curva ROC obtenida a partir de estos datos y en la Figura 6.b la curva de la exactitud del modelo, ambas obtenidas para diferentes valores de corte.

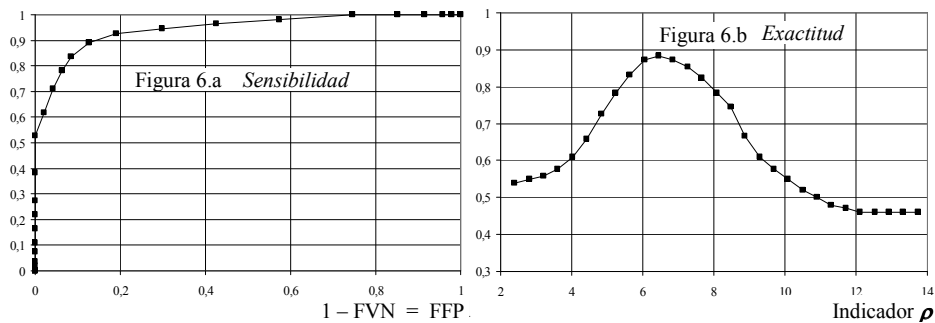


Figura 6: a) Curva ROC que representa la sensibilidad en función de FFP (1-especificidad) y b) Curva de exactitud (Ecuación 17) del modelo para los datos del caso de estudio.



Habiendo adoptado un modelo en base al indicador  $\rho$ , surge ahora la necesidad de conocer el valor de corte  $\rho_c$ , que como se dijo corresponderá al punto en que la curva ROC tenga pendiente  $m$ . Tal como lo expresa la Ecuación 19,  $m$  depende de la probabilidad de obtenerse un caso positivo y de los costos relativos que corresponden a los diagnósticos erróneos. Para facilitar la tarea de identificar el mejor valor de corte  $\rho_c$  se ajustaron expresiones polinomiales para representar la sensibilidad, su pendiente  $m$  y el punto de corte  $\rho_c$  en función de FFP (1-especificidad), cuyas curvas se muestran en la Figura 7.a. Una vez comprobado el buen desempeño de los polinomios, fueron reordenados para representar  $\rho_c$  en función de  $m$ , como lo muestra la curva en la Figura 7.b.

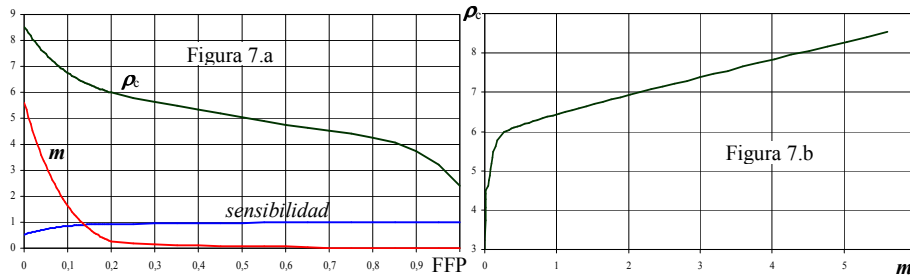


Figura 7: a) Evolución de la sensibilidad,  $m$  y  $\rho_c$  en función de FFP y b) Representación gráfica de  $\rho_c$  con la pendiente  $m$  en las abscisas.

Como puede observarse, la curva  $\rho_c = f(m)$  en el intervalo  $0,25 \leq m \leq 5,60$  es muy suave, casi una recta, lo que facilita representarla por un polinomio de tercer grado:

$$\rho_c = 0,0045m^3 - 0,051m^2 + 0,6235m + 5,8524 \quad (20)$$

Es decir que se dispone de una expresión simple que, dado un cierto valor de  $m$ , permite anticipar el valor del indicador  $\rho_c$  que corresponde a la separación de los proyectos trazables de los no trazables. Para ejemplificar sobre la aplicación del modelo se presentan en la Tabla 2 resultados que corresponden a la probabilidad de un resultado positivo en el caso de estudio y las relaciones de costos  $C(FP)/C(FN) = 3, 2, 1, 1/2$  y  $1/3$ . En la Figura 8 se muestran las poblaciones de los valores verdaderos y falsos, positivos y negativos, para el caso el que  $C(FP)/C(FN) = 1$ . Esto equivale a decir que el impacto de errar en uno u otro sentido es el mismo, y en este caso al valor de  $\rho_c$  lo ha permitido anticipar el gráfico de exactitud de la Figura 6.b.

Tabla 2: Ejemplos de cálculo de valores de corte del indicador  $\rho_c$  (Fórmula 20)

$p(P)$	$\frac{C(FP)}{C(FN)}$	$m = \text{tg}(\alpha)$	Valor corte $\rho_c$
0,8545	3	2,5635	7,1914
	2	1,7090	6,7914
	1	0,8545	6,3507
	1/2	0,4273	6,1098
	1/3	0,2848	6,0259

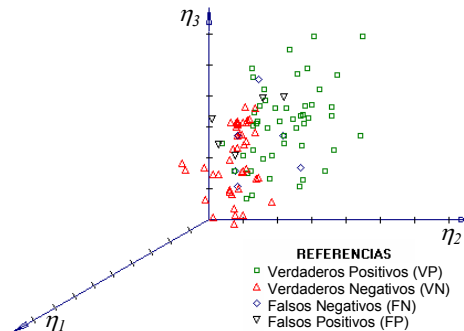


Figura 8: Clasificación para  $\rho_c = 6,3507$

Se dispone así de un indicador apto para predecir la conveniencia de trazar proyectos de desarrollo de software. El modelo se caracteriza por su facilidad de cálculo y clara interpretación, por ser su indicador la resultante de ocho factores que representan las cualidades de todo proyecto. Como ocurre con todos los modelos de predicción en la ingeniería de software, su mejor desempeño y utilidad se obtiene al ser aplicado en el seno de las organizaciones, con datos provenientes de sus propias experiencias. No obstante, los resultados obtenidos con datos de dominio público representan referencias generales de gran utilidad, a los que necesariamente se les debe asignar un valor orientativo.

#### 4. Conclusiones

Los trabajos que se vienen realizando condujeron a proponer un indicador, en este caso “ $\rho$ ”, destinados a anticipar con cierta facilidad el resultado que puede esperarse de la trazabilidad de un proyecto. Más aún, el indicador pone de manifiesto la aptitud de un proyecto para ser trazado, según la distancia que lo separa del valor de corte “ $\rho_c$ ” en uno u otro sentido. Para ello se utilizó el análisis ROC, que es una técnica muy difundida para la evaluación del desempeño de modelos dicotómicos. Sin embargo, y tal como ocurre con todos los modelos de predicción, su exactitud se apoya en la disponibilidad de datos históricos específicos en calidad y cantidad suficiente. Debido a que en el ámbito empresarial se asigna un valor estratégico a las predicciones, no es fácil reunir información confiable de dominio público para ajustar y validar este tipo de modelos en un contexto general. La actividad futura pondrá el foco en este aspecto y en estimular la adopción del modelo por parte de las organizaciones.

#### Referencias

1. Kannenberg, A., Saiedian, H.: Why Software Requirements Traceability Remains a Challenge. *CrossTalk: The Journal of Defense Software Engineering*. July/August, 14-19 (2009).
2. Ramesh, B.: Factors influencing Requirements Traceability Practice. *Communications of the ACM*. 41(12), 37-44. (1998)
3. Blaauboer, F., Sikkel, K., Aydin, M.: Deciding to adopt requirements traceability in practice. *Proc. of the 19th Int. Conf. on Advanced Infor. Systems Engineering*. Springer-Verlag (2007).
4. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A.: Aseguramiento de la Trazabilidad en Proyectos de Desarrollo de Sistemas de Software. Proyecto de Investigación, Secretaría de Ciencia y Tecnología, Código SCyT 155/10. (2010)
5. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A.: Modelos para anticipar la factibilidad de que un proyecto de desarrollo de software sea trazable. *Workshop de Ingeniería de Software, CACIC 2011*. Universidad Nacional de La Plata. 837-846 (2011)
6. Giró, J., Vazquez, J., Meloni, B., Constable, L., Jornet, A.: Hacia una respuesta al interrogante de si será factible trazar un cierto proyecto de desarrollo de software. Informe Técnico 2012/01, Proyecto 155/10, SCyT, FRC, UTN (2012).
7. Powers, D.: Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. Technical Report SIE-07-001, School of Informatics and Engineering, Flinders University, Adelaide, Australia (2007).
8. Fawcett, T.: An introduction to ROC analysis. *Elsevier ScienceDirect, Pattern Recognition Letters*, 27 - 861-874 (2006).
9. Shin Y., Huffman Hayes J., Cleland-Huang J.: A framework for evaluating traceability benchmark metrics. TR:12-001, DePaul University, School of Computing (2012).