

# Sistema experto de ayuda al mantenimiento de máquinas enfundadoras basado en aprendizaje automático

J. Gómez-Sanchis \*, A.J Serrano-López, M. Martínez-Sober, J.J Garcés-Iniesta, E. Soria-Olivas, G. Vergara, and José.M. Martínez-Martínez

IDAL, Universidad de Valencia,  
Av. de la Universidad SN., 46100 Burjassot, España  
<http://idal.uv.es/>

**Abstract.** El presente trabajo presenta un sistema experto para la emisión automática de ordenes de trabajo para el mantenimiento predictivo de máquinas enfundadoras. A partir de las alarmas emitidas por la máquina, se determinará el momento en el que hay que actuar sobre la máquina. El problema de aprendizaje automático planteado presenta un fuerte desbalanceo en las clases, así pues se empleará la técnica SMOTE (*Synthetic Minority Over-sampling TEchnique*) para balancear las clases. En cuanto al problema de clasificación, se aplicarán varias técnicas de aprendizaje automático: modelos lineales, redes neuronales y árboles de decisión. Se presenta la metodología utilizada y los resultados obtenidos encontrando que la técnica que mejores resultados obtiene en este caso concreto son los árboles de decisión con un valor de área bajo la curva ROC de 0.847.

**Keywords:** SMOTE, Aprendizaje automático, Árboles de decisión, Redes neuronales artificiales, Modelos lineales generalizados, Clases desbalanceadas

## 1 Introducción

En los últimos años la conectividad de todo tipo de dispositivos, y en particular los industriales, ha experimentado un auge debido a la mejora y abaratamiento de buses basados en *ethernet*. Este hecho, sumado al bajo precio de los dispositivos de almacenamiento, ha hecho que la tendencia actual sea que casi cualquier dispositivo industrial desde un PLC (*Programable Logic Controller*) hasta una célula de fabricación completa, esté conectada a la red y envíe datos relacionados con el proceso de fabricación (productividad, estado de la máquina, etc.). Así pues, el análisis de datos relacionados con la mejora de procesos industriales es una realidad y prueba de ello es el extenso estado del arte sobre dicha cuestión [8]. La incorporación de técnicas avanzadas de análisis de datos y modelización de sistemas permite ir más allá de la simple visualización de parámetros de

---

\* [Juan.Gomez-Sanchis@uv.es](mailto:Juan.Gomez-Sanchis@uv.es)

proceso, consiguiendo información importante para la optimización del mantenimiento y gestión de las mismas. Tradicionalmente se ha investigado mucho sobre técnicas para que células de fabricación industriales no se averiasen llevando a cabo acciones de mantenimiento preventivo [4]. No obstante, en la actualidad está empezando a surgir otro tipo de estrategias basadas en el mantenimiento predictivo [7], que tiene en cuenta aspectos novedosos como la monitorización en tiempo real de alarmas y el estado de la máquina para determinar el momento óptimo de llevar a cabo acciones de mantenimiento. La recolección de datos sobre el estado de la máquina y su almacenamiento se lleva a cabo a través de interfaces de comunicaciones conectados entre la máquina y servidores informáticos, en los que se alojan los algoritmos predictivos para ayudar a planificar la mejor estrategia de mantenimiento.

En este trabajo se va a abordar el problema de establecer estrategias de mantenimiento predictivo en un conjunto máquinas envasadoras de una empresa proveedora de servicios de *packaging* a centros logísticos. Las máquinas envasadoras son dispositivos que se dedican a la aplicación de film “a palet parado”. Están diseñadas para envolver cargas sin necesidad de que éstas tengan que girar en una plataforma, con el fin de asegurar la carga, protegerla del polvo o del agua, reducir el volumen de la carga, reforzar el embalaje, etc. Todas las máquinas de la empresa están conectadas en red y reportan periódicamente una serie de alarmas asociadas a determinados tipos de funcionamientos anómalos. Todas las variables de estado de cada máquina son recibidas en un servidor central y almacenadas por una base de datos. A pesar de disponer de todas las variables de estado de cada una de las máquinas en tiempo real existen varios desafíos para desarrollar un sistema experto que realice predicciones sobre cuando la empresa debe mandar a un técnico para realizar las reparaciones oportunas. Estos desafíos están relacionados con varias cuestiones: 1) resolver el fuerte desbalanceo de las clases en el conjunto de datos para construir un clasificador no sesgado hacia ninguna de las clases, ya que las situaciones en las que es necesario enviar a un técnico son muy pocas en comparación con las que no hay que tomar ninguna acción; 2) la extracción de las alarmas relevantes de la base de datos y la construcción del conjunto de entrenamiento; 3) la definición de una estrategia para fijar la importancia de alarmas pasadas en el sistema de clasificación; 4) la frecuencia de actualización de los modelos; 5) la integración de los algoritmos de clasificación en el sistema completo de gestión de la empresa.

Así pues, el objetivo del presente trabajo consistirá en el desarrollo e integración de un sistema experto que determine, a partir de las alarmas reportadas por las envasadoras, el momento óptimo para enviar a un técnico a realizar labores de reparación y mantenimiento. Para ello se compararán clasificadores lineales y no lineales, con el fin de determinar cuáles proporcionan mejores resultados utilizándose técnicas para conseguir suavizar el efecto del importante desbalanceo de las clases presentes en el conjunto de datos.

El resto del trabajo se organizará en una sección de materiales y métodos donde se presentará la estrategia para construir el conjunto de entrenamiento así como los algoritmos de aprendizaje automático responsables de la clasificación y

de solventar el problema del desbalanceo de las clases. Una sección de resultados y discusión, donde se presentarán los resultados de clasificación con cada uno de los clasificadores y se discutirán los mismos. Por último, se presentarán las conclusiones del trabajo en la última sección.

## 2 Datos y métodos

### 2.1 Datos

En esta sección se va a presentar el modo en el que se construye la matriz de entrenamiento del sistema experto a partir de los eventos reportados por cada máquina en la base de datos. El conjunto de entradas al sistema experto está relacionado con la frecuencia de ocurrencia de las diferentes alarmas que recoge el sistema y el tiempo que ha transcurrido desde la última reparación. La figura 1 (a) muestra un ejemplo de la matriz de eventos que se extrae de la base de datos. Se puede observar los diferentes tipos de eventos que envían las máquinas conforme van ocurriendo. Estos eventos son alarmas, contadores, partes, órdenes y tareas. Un evento de tipo alarma indica la alarma que se ha producido, un evento de tipo contador indica la cantidad de film plástico que ha utilizado la máquina ese día. Los eventos de tipo orden y parte indican cuando se ha emitido una orden de trabajo y cuando se ha realizado la reparación respectivamente. Las especificaciones del sistema experto indican que el sistema debe dar una predicción cada día con el fin de hacer la planificación diaria de reparación de máquinas. Esta restricción hace que el mínimo intervalo temporal para agrupar la información que se recibe de la matriz de eventos sea de un día. Así pues, el periodo en el que los patrones de cada máquina son muestreados es diario. Se ha recopilado la información diaria de una máquina a nivel de eventos (ocurrencia de alarmas, contadores, ordenes, partes y tareas asociadas a dichos partes) y, posteriormente se ha organizado dicha información en un vector que contiene toda la información que proviene de ese día. Cada fila de la matriz de patrones contiene campos como la dirección MAC de la máquina, la fecha del evento que se está almacenando la información, la frecuencia de cada una de las 22 alarmas de ese día (A1..Am), si ese día se ha emitido una orden o un parte y, por último, las tareas llevadas a cabo ese día (es evidente que solo días que tengan un parte tendrán tareas asociadas). A pesar de que en este trabajo no se han utilizado las tareas, se han exportado de la base de datos también para aumentar la funcionalidad del sistema experto en el futuro.

Del mismo modo, la figura 1 (b) muestra un ejemplo de cómo se organiza la información en la matriz de patrones a partir de la matriz de la eventos. La figura muestra un ejemplo sencillo para explicar cómo se reordena la información derivada de los eventos de una máquina, en patrones útiles para los algoritmos de aprendizaje automático. La dirección MAC de esa máquina es 00.50.C2.E4.17, los eventos que ha reportado esa máquina van desde el 9/2/2015 hasta 13/2/2015. Puesto que la máquina ha estado enviando información 4 días, la matriz de patrones resultante tendrá cuatro filas (una correspondiente a cada día). Las alarmas (campos en rojo) recopilan la información de entrada al sistema experto



Fig. 1. Esquema de construcción del conjunto de datos para desarrollar el sistema experto

(aún sin preprocesar temporalmente), los campos en amarillo y naranja, constituirán la señal de salida de cada uno de los módulos del sistema experto. Desde la empresa se resaltó la importancia de tener en cuenta, de un modo u otro, el histórico de la máquina a nivel de alarmas. De este modo, se ha optado por realizar un filtrado IIR de primer orden sobre las alarmas para tener en cuenta la información de todo el histórico de la máquina. La salida del filtrado aplicado viene dada por la siguiente expresión.

$$y(n) = x(n) + \alpha y(n - 1)$$

Para este estudio, se acordó con los expertos de la empresa que se probarán dos factores de olvido,  $\alpha = 0.63$  y  $\alpha = 0.95$ , como paradigmas de filtrado dando mucho y poco peso respectivamente, a las muestras pasadas. Finalmente, descartando el bloque de tareas se consigue el conjunto de datos para el sistema experto de emisión automática de ordenes (figura 1 c) con 725 muestras. Este conjunto tiene como variables de entrada las alarmas filtradas como se ha indicado anteriormente y como variable de salida si ese día se ha emitido una orden o no. Se han construido 4 conjuntos de datos (Grupo1 a Grupo 4), como se puede ver en la tabla 1, teniendo en cuenta si se ha ponderado el efecto acumulativo de las alarmas producidas ( $\alpha$ ) y si se ha tenido en cuenta la información de los contadores o no. Cada grupo de datos se ha dividido en dos subconjuntos de manera aleatoria: entrenamiento, compuesto por 2/3 del total de muestras y el de validación, compuesto por las restantes muestras. Para evaluar el rendimiento de los clasificadores se ha empleado el parámetro AUC (Área bajo la curva ROC) en el conjunto de validación. Se han realizado 100 simulaciones, cada una de las veces

con diferentes conjuntos de entrenamiento y validación (reparto entrenamiento/validación), obteniendo un área bajo la curva ROC para cada una de las 100 simulaciones con el fin de estudiar la robustez de los modelos ante la división entrenamiento/validación. Esto será repetido para cada modelo y cada grupo de datos mostrado en la tabla 1.

## 2.2 SMOTE

Después de un análisis rápido del conjunto de entrenamiento se constató que las clases estaban muy desbalanceadas (1.3% positivo, 98.7% negativo). Así pues se ha optado por utilizar la técnica SMOTE para minimizar el efecto del desbalanceo de las clases en el clasificador.

El método SMOTE [3] consiste en obtener más patrones de la clase minoritaria de forma sintética sin usar el método del reemplazamiento al tiempo que se muestrea la mayoritaria. Tiene su base en [5] y consiste en crear patrones adicionales de la clase minoritaria usando un procedimiento de k-vecinos (de forma usual se escoge  $k=5$ ), generando 5 muestras ( $N=5$ ) en sus intervalos, al tiempo que se muestrea la clase mayoritaria para evitar el desbalanceo de las clases (se eligen la mitad  $R=0.5$ ). Para generar nuevas muestras se selecciona de forma aleatoria uno de los k-vecinos de cada clase minoritaria (y que también tienen que pertenecer a la clase minoritaria); el siguiente paso es obtener un nuevo patrón en el segmento que va del patrón original al vecino escogido de forma aleatoria. Se han generado 2 conjuntos (B1 y NO), el primero utilizando SMOTE con los parámetros arriba indicados y el segundo sin aplicar SMOTE como conjunto de control.

Nombre	Explicación
Grupo 1	$\alpha = 0.95$ , sin incluir contadores
Grupo 2	$\alpha = 0.95$ , incluyendo contadores
Grupo 3	$\alpha = 0.63$ , sin incluir contadores
Grupo 4	$\alpha = 0.63$ , incluyendo contadores
B1	Datos balanceado con SMOTE y con una reducción de la clase mayoritaria: $N = 5$ , $k = 5$ , $R = 0.5$
NO	Datos en las proporciones originales, sin balancear

**Table 1.** Codificación de grupos según  $\alpha$ , inclusión o no de contadores y uso de SMOTE variando parámetros del balanceo. La nomenclatura queda de la forma siguiente: 3\_B1 indica grupo 3 con SMOTE B1 y 2\_NO indica grupo 2 sin SMOTE.

En este trabajo se han empleado 3 métodos de clasificación, en concreto se ha empleado una regresión logística, un árbol de clasificación con el algoritmo CART y un perceptrón multicapa. La razón para la elección de estos tres clasificadores ha sido diferente para cada clasificador. La regresión lineal se ha utilizado como paradigma de clasificador simple que proporcione una línea de referencia en la comparación. En cuanto al árbol de decisión se eligió como paradigma de

clasificador con fronteras de decisión lineales. Por último, el perceptrón multicapa se eligió como paradigma de clasificador con fronteras de clasificación no lineales. Todos los clasificadores han sido entrenados utilizando el software R y la librería caret [6].

### 2.3 Modelos lineales (RL).

Como modelo lineal se ha empleado una regresión logística, este modelo se caracteriza por ser un modelo lineal generalizado que estima la probabilidad de que una muestra pertenezca a una clase. Así pues se calcula la probabilidad condicional de que, dada la muestra  $X$  pertenezca a la clase  $Y = 1$ .

$$P(Y = 1|X) = \frac{1}{1 + e^{-\sum_{i=1}^n \omega_i x_i}}$$

siendo  $x_0..x_n$  las características de entrada de la muestra. Los coeficientes de la regresión logística  $\omega_i$  se han obtenido a través del procedimiento de optimización clásico como si fuera un modelo lineal generalizado.

### 2.4 Redes neuronales artificiales (RNA).

Se ha utilizado un perceptrón multicapa con 2 capas ocultas [1]. Se ha utilizado la regla Delta como algoritmo de actualización, iterando hasta alcanzar el mínimo error o un máximo de iteraciones. La bondad de la red obtenida depende de los parámetros de ajuste, que en este caso depende de la inicialización, y el número de neuronas en cada capa. Puesto que no se conocen los valores óptimos, se han evaluado una batería de 100 inicializaciones aleatorias con el fin de evitar mínimos locales de la función de error evaluándose diferentes arquitecturas variando el número de neuronas de 1 a 8 en la capa oculta.

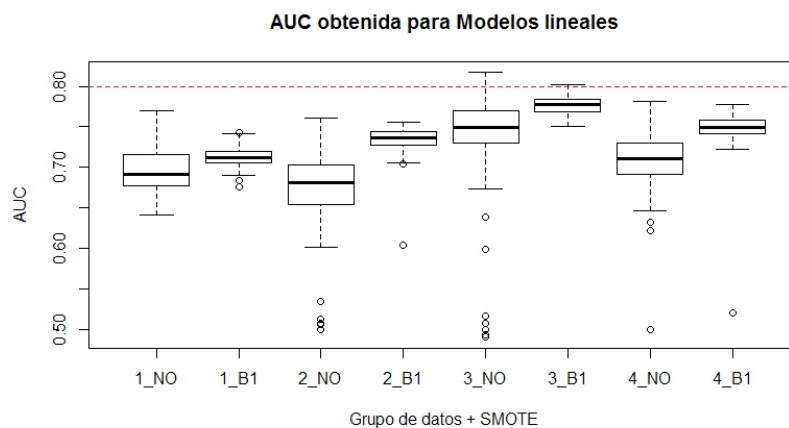
### 2.5 Árboles de decisión (DT).

Los árboles de clasificación y regresión constituyen un método estadístico no paramétrico utilizado tanto para clasificación como para regresión [2]. Si la variable dependiente del problema es continua, DT produce un árbol de regresión. Por el contrario, si la variable es categórica, DT producirá un árbol de clasificación. En este trabajo, DT se ha utilizado para producir árboles de decisión con el fin de clasificar patrones. Los árboles de decisión proporcionan conocimiento sobre las interacciones entre las variables del problema y los fenómenos que explican la relación entre dichas variables. Los modelos de clasificación generados con los árboles de decisión se caracterizan porque: 1) El modelo es independiente de la distribución estadística de las variables independientes y de las dependientes; 2) El éxito de la clasificación no está especialmente afectado por *outliers*. Los patrones anómalos son aislados en un nodo del árbol. De este modo DT se muestra robusto ante la presencia de datos anómalos; 3) Proporciona un modo de descubrir relaciones ocultas entre las variables presentes en el problema de

clasificación analizando las divisiones que produce cada nodo; 4) Proporciona modelos completamente interpretables, es decir, en problemas en los que no hay teoría al respecto, DT se puede utilizar como una herramienta analítica. En este trabajo se han empleado dos aproximaciones a su uso: los árboles sin podar obtenidos en entrenamiento y los árboles podados por validación cruzada.

### 3 Resultados y discusión

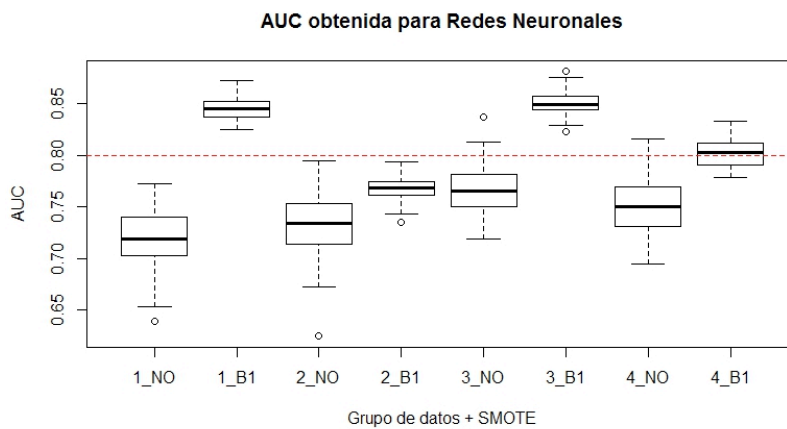
La figura 2 muestra la distribución de áreas bajo la curva para cada uno de los ensayos realizados en el conjunto de validación. En la misma figura se observa que los modelos lineales proporcionan resultados muy discretos, no obteniendo en ningún caso una AUC por encima de 0.8. El valor más alto obtenido de AUC es de 0.77, que se da para el grupo 3 ( $\alpha = 0.63$ , sin conteo de vueltas) con el grupo de datos B1 generado con SMOTE. No obstante, sí se aprecia una mejora con el uso de SMOTE, que se traduce en valores más altos de media y una menor dispersión en los resultados obtenidos.



**Fig. 2.** Área bajo la curva ROC obtenida para los modelos lineales. Valores referidos a los resultados de conjuntos de validación. La línea discontinua indica el nivel de AUC de 0.80, valor que la empresa indicó como aceptable.

A priori la capacidad de modelización de las redes neuronales es mucho más elevada que la de los modelos lineales. La figura 3 pone de manifiesto esta afirmación. Se observa que los resultados de AUC para los modelos neuronales son mucho mejores que en el caso de los modelos lineales. La misma figura muestra como en tres de los ensayos se supera un valor de AUC de 0.8, obteniéndose el mejor resultado en el caso 3\_B1 con AUC media de 0.85, seguido del caso 1\_B1 con 0.84 y el caso 4\_B1 con 0.802. Todos ellos son para grupos tratados

con SMOTE. Ninguno de los ensayos sin utilizar SMOTE ha permitido obtener alguna red neuronal con suficiente capacidad de clasificación en el problema tratado. La desviación estándar de la AUC para cada uno de los ensayos sin SMOTE es mayor que con SMOTE. Esta tendencia se repite tanto para los modelos lineales como para los árboles de clasificación. Así pues, se demuestra que usando SMOTE en este problema, se obtiene clasificadores más robustos ante la división del conjunto de datos. Otro hecho que se pone de manifiesto es que la inclusión de la información proporcionada por los contadores introduce ruido en los modelos neuronales, ya que estos proporcionan peores resultados. Esta tendencia se repite tanto en los modelos neuronales como en los árboles de clasificación.

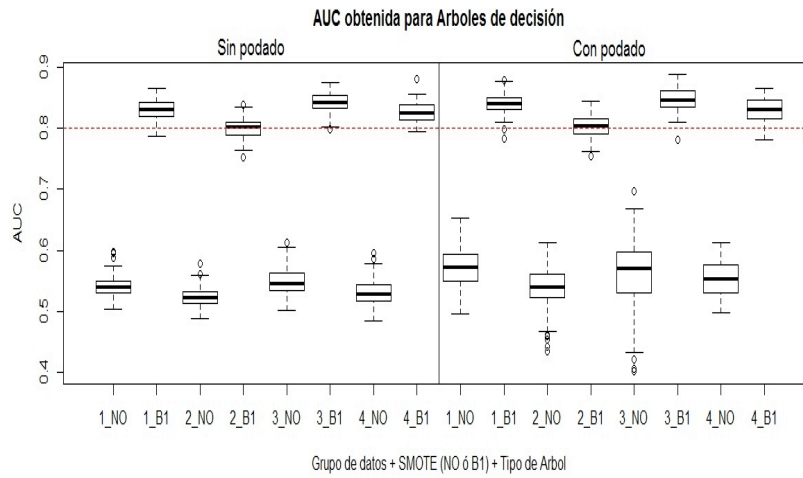


**Fig. 3.** Área bajo la curva ROC obtenida para los modelos neuronales ensayados. Valores referidos a los resultados de conjuntos de validación. La línea discontinua indica el nivel mínimo propuesto de 0.80.

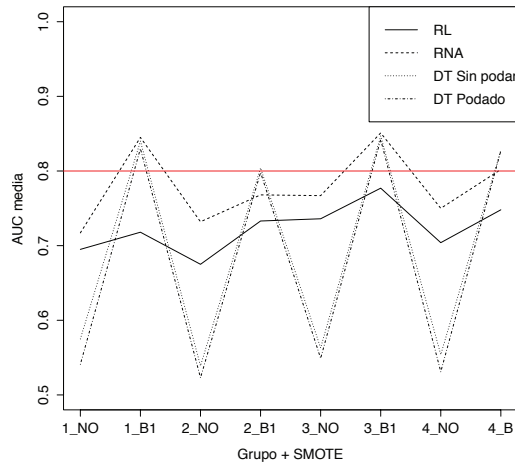
La figura 4 muestra que los árboles de decisión proporcionan mejores resultados que las redes neuronales en el problema estudiado. Casi todos los casos en los que se ha aplicado SMOTE han conseguido una media AUC superior a 0.80. El mejor resultado se obtiene en el ensayo 3\_B1 (0.842 para árbol sin poda y 0.847 para árbol podado). Con los árboles de clasificación también se confirma la tendencia general en cuanto a los resultados usando SMOTE, el mejor de los ensayos sin SMOTE proporciona una AUC de 0.57.

El gráfico de coordenadas paralelas mostrado en la figura 5 muestra una comparación los modelos y ensayos realizados. Se observa que los modelos que proporcionan mejores resultados son los modelos neuronales y los árboles de decisión. Del mismo modo, se observa que los ensayos en los que se ha utilizado SMOTE proporcionan mejores resultados que en los que no se ha utilizado SMOTE.





**Fig. 4.** Área bajo la curva ROC obtenida para los árboles de decisión (en este caso, de clasificación). En la parte izquierda de la figura se muestran resultados sin aplicar la poda y en la derecha con poda. Valores referidos a los resultados de conjuntos de validación. La línea discontinua indica el nivel mínimo propuesto de 0.80.



**Fig. 5.** Comparativa de Area bajo la curva ROC obtenida para las diferentes técnicas de modelización.

## 4 Conclusiones

Se han estudiado 3 técnicas clásicas de clasificación para un problema de emisión automática de ordenes de trabajo en máquinas enfardadoras. La principal difi-

cultad encontrada para resolver el problema es el elevado desbalanceo de las clases en el conjunto de entrenamiento. La técnica para balancear las clases SMOTE se ha mostrado eficaz y su uso proporciona los mejores resultados en el problema tratado. Empleando SMOTE, en el mejor de los ensayos, se pasa de una proporción de muestras minoritarias de 1.3% a 13.1%, este aumento en la clase minoritaria ha sido suficiente para conseguir clasificadores con la exactitud requerida por la empresa. En cuanto a la elección del factor del olvido  $\alpha$ , se ha observado que el valor de 0.63 ha proporcionado mejores resultados que el de 0.95 en la aproximación sin considerar los contadores. Los modelos lineales son una técnica de obligada evaluación por su sencillez y robustez, pero las redes neuronales y los árboles de clasificación han proporcionado resultados notablemente mejores. Los árboles de decisión han proporcionado unos resultados ligeramente mejores que las redes, siempre y cuando se empleen datos balanceados. La inclusión de información de contadores en líneas generales introduce ruido en los modelos de clasificación en contra de la opinión previa de los expertos de la empresa.

**Acknowledgments.** Este trabajo ha sido financiado por la empresa Aranguren Comercial de Embalaje SL (ARANCO) en el ámbito del proyecto "Desarrollo de un sistema experto para la clasificación de eventos en máquinas y optimización de tareas de mantenimiento" entre la Universidad de Valencia y la empresa en el marco de un proyecto CDTI.

## References

1. Bishop, C.: *Neural Networks for Pattern Recognition*. Clarendon Press (1995)
2. Breiman, L., Friedman, P., Olshen, R., Stone, C.: *Classification and Regression Trees*. Wadworth, Belmont-CA (1984)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.* 16(1), 321–357 (Jun 2002), <http://dl.acm.org/citation.cfm?id=1622407.1622416>
4. Chouikhi, H., Khatab, A., Rezg, N.: A condition-based maintenance policy for a production system under excessive environmental degradation. *Journal of Intelligent Manufacturing* 25(4), 727–737 (2014), <http://dx.doi.org/10.1007/s10845-012-0715-9>
5. Ha, T.M., Bunke, H.: Off-line handwritten numeral string recognition. *Pattern Recognition* 31, 257–272 (1997)
6. Kuhn, M., Johnson, K.: *Applied Predictive Modeling*. Springer (2013)
7. Wang, J., Zhang, L., Duan, L., Gao, R.: A new paradigm of cloud-based predictive maintenance for intelligent manufacturing. *Journal of Intelligent Manufacturing* pp. 1–13 (2015), <http://dx.doi.org/10.1007/s10845-015-1066-0>
8. Yin, S., Li, X., Gao, H., Kaynak, O.: Data-based techniques focused on modern industry: An overview. *IEEE Transactions on Industrial Electronics* 62(1), 657–667 (Jan 2015)