

# MINERÍA DE DATOS EDUCATIVOS PARA LA DETECCIÓN DE RECURSOS CLAVE

## EDUCATIONAL DATA MINING TO DETECT KEY RESOURCES

Eva Gibaja Galindo  
Amelia Zafra Gómez  
María Luque Rodríguez  
Antonio Arauzo Azofra  
Aurora Ramírez Quesada  
Juan Luis Olmo Ortiz

{egibaja, azafra, mluque, arauzo, aramirez, jlolmo}@uco.es

Universidad de Córdoba. Departamento de Informática y Análisis Numérico

### Abstract

This article describes an educational innovation project whose aim is designing and developing a new Moodle block to obtain decision tree based predictive models from usage data store in Moodle. Due to the features of these decision tree based models, it is possible to establish a relationship between student's work and the final mark, and to obtain a description of the key resources to pass/fail a certain subject. With this information, the instructor could detect students with a high probability of fail and then try to correct the situation. This block could be applied to any subject in the Moodle platform. After developing the block, it has been tested with some of the subjects given by teachers involved in the project.

**Keywords:** Educational data mining; key resources; classification, decision trees

### Resumen

Este artículo describe un proyecto de innovación educativa centrado en diseñar y desarrollar un nuevo módulo de Moodle que permita obtener modelos predictivos basados en árboles de decisión a partir de los datos de uso almacenados en Moodle. Debido a las características de los modelos basados en árboles de decisión, es posible relacionar el trabajo personal realizado por los alumnos en Moodle con la calificación final obtenida, proporcionándonos así una descripción de los recursos y actividades que resultan clave para superar/suspender una asignatura. Con esta información, el profesor podría detectar alumnos con alta probabilidad de no superar la asignatura e intentar reconducir la situación. Este módulo puede ser aplicado en cualquier asignatura de la plataforma Moodle. Concretamente, para comprobar su funcionamiento, ha sido aplicado la asignatura *Introducción a la Programación* impartida en el grado de Ingeniería Informática de la Universidad de Córdoba.

**Palabras clave:** Minería de datos educativos, recursos clave, clasificación, árboles de decisión

## 1. INTRODUCCIÓN

Las plataformas *e-learning* han tenido muy buena aceptación y se han convertido en una herramienta esencial en la comunidad educativa. Así, constituyen uno de los pilares fundamentales en la construcción del conocimiento, permitiendo nuevos canales de comunicación tanto del aprendizaje síncrono como asíncrono, y dotando de un gran potencial tanto a los docentes como a los alumnos (Boneu, 2007).

Con la llegada de estas plataformas, nos encontramos con una gran cantidad de información almacenada sobre la actividad del estudiante, cuyo análisis resulta de gran interés para comprender mejor su proceso de aprendizaje. El problema es que identificar y encontrar conocimiento útil en toda la información disponible es una tarea difícil. Una solución muy prometedora es el uso de técnicas de extracción de conocimiento o minería de datos. Cuando estas técnicas son aplicadas sobre datos educativos, se habla de minería de datos educativos (*Educational Data Mining*, EDM). La *Minería de Datos Educativos* (Romero y Ventura, 2010) se define como un proceso no trivial que nos permite transformar los datos almacenados en los sistemas de enseñanza en conocimiento útil que pueda utilizarse para tomar decisiones. Es una disciplina emergente que aplica minería de datos en el ámbito de la enseñanza con el objetivo de obtener una mejor comprensión del proceso de aprendizaje de los estudiantes, orientado a la mejora de la calidad y la rentabilidad del sistema educativo.

Una de las aplicaciones más populares de la EDM es la predicción y entendimiento de los resultados académicos de un estudiante. En este contexto, pretendemos predecir el rendimiento académico y los factores que influyen en él (*recursos clave*) en función de la actividad realizada por los alumnos en una plataforma de *e-learning*. Para llevar a cabo esta tarea, partimos de la información sobre la actividad de los estudiantes que está almacenada en los registros *log* de la plataforma. Estos alumnos son evaluados y etiquetados como aprobados o suspensos. A continuación, se construye un modelo predictivo utilizando un algoritmo de clasificación con los datos de la etapa anterior. El modelo nos permitirá predecir si un nuevo estudiante aprobará o suspenderá a partir de su información almacenada en la plataforma. Trabajaremos con modelos de clasificación interpretables que, además de dar una predicción, proporcionen una explicación con las evidencias en las que se han basado para llevar a cabo la

clasificación. En este caso, estas evidencias serán las actividades y recursos (A&Rs) realizados en la plataforma y que han resultado clave para poder superar o no una asignatura. Con esta información, sería posible detectar alumnos con un alto riesgo de no superar la asignatura, para así tratar de tomar acciones preventivas que reconduzcan su proceso de aprendizaje.

## 2. OBJETIVOS

El objetivo fundamental de este proyecto ha sido desarrollar un módulo en Moodle (*Module Object-Oriented Dynamic Learning Environment*) (Ros, 2008) para obtener información de cuáles son las A&Rs que resultan más relevantes para superar una asignatura. La selección de la plataforma Moodle para el desarrollo del módulo, se ha debido a que es la plataforma educativa que se encuentra implantada en la Universidad de Córdoba, y por tanto donde se encuentran almacenados todos los registros de actividad de los alumnos de todas las asignaturas de Grados y Másteres que se imparten en esta Universidad. Participantes de este proyecto, ya han desarrollado previamente otros proyectos basados en Moodle (Luque et al., 2014), (Zafra et al., 2014).

Los algoritmos de clasificación que se han aplicado están basados en árboles de decisión y están disponibles en el software de Weka (Hall et al., 2009):

- Algoritmo *J48* (Quinlan, 2014). Este algoritmo genera un árbol de decisión a partir de una tabla donde cada fila es un patrón descrito por un conjunto de atributos.
- Algoritmo *RandomForest* (Breiman, 2001). A diferencia de *J48*, este algoritmo genera varios árboles de decisión.

La selección de estos algoritmos se debe a que generan modelos predictivos muy interpretables. Así, además de proporcionar una predicción sobre el rendimiento académico, nos permiten estudiar las A&Rs que se están considerando más relevantes para determinar si un estudiante tiene una alta probabilidad de obtener una calificación satisfactoria en una determinada asignatura. De este modo, el módulo diseñado e implementado permite utilizar la información de la actividad llevada a cabo por el alumno en la plataforma educativa y las calificaciones almacenadas en Moodle para obtener un modelo predictivo que nos permita detectar si existen recursos que son determinantes para superar la asignatura. El módulo desarrollado presenta además las siguientes características:

- Cada uno de los algoritmos incorporados puede ser configurado considerando los parámetros más relevantes.
- La selección de los estudiantes y de la actividad que se quiere considerar es flexible permitiendo la selección de:
  - Un conjunto de estudiantes definidos por el profesor.
  - Un conjunto de A&Rs definidos por el profesor.
  - Un rango de fechas definido por el profesor.
- Se puede incorporar la calificación de los alumnos (en caso de que el profesor no esté utilizando el libro de calificaciones de Moodle).
- Se puede mantener el anonimato de toda la información personal utilizada. Esto facilitará la publicación de los resultados obtenidos en medios de difusión científica o docente.
- Es posible guardar los datos de log utilizados para generar los modelos en otros formatos para su posterior uso en otras herramientas de minería de datos.
- Es posible utilizar diversos niveles de desempeño para las A&Rs (accedido/no accedido, número de accesos, calificación).
- Es posible utilizarlo en inglés o en español.
- El módulo se puede utilizar en cualquier asignatura de la plataforma Moodle.

Desde un punto de vista técnico:

- El desarrollo ha seguido el formato estándar de Moodle, de manera que su manejo es lo más intuitivo posible y personas no expertas en minería de datos pueden aplicar estos algoritmos y obtener información relevante de las relaciones entre el trabajo de los alumnos y su calificación.
- El módulo incluye iconos de ayuda, con la finalidad de orientar al usuario.
- El módulo es robusto y fácil de ampliar y mantener en el futuro.

### 3. METODOLOGÍA

Para llevar a cabo este trabajo, se ha empleado un servidor del grupo de investigación KDIS de la Universidad de Córdoba. El bloque se ha implementado en PHP con XAMPP <sup>1</sup>, que convierte al ordenador en un servidor que ofrece: Apache, MySQL, PHP y phpMyAdmin.

Para alcanzar los objetivos previamente comentados, el proyecto se ha desarrollado en las siguientes fases:

1. *Estudio previo de la plataforma Moodle.* Se ha llevado a cabo un estudio de la base de datos del sistema Moodle para poder obtener la información necesaria de los registros *log*. También, ha sido necesario llevar a cabo un estudio de la API de desarrollo de Moodle, para poder aprovechar toda la funcionalidad que ya está incorporada en la plataforma educativa. Finalmente, se ha llevado a cabo un estudio de los módulos con los que interactuará el bloque a desarrollar.
2. *Diseño y desarrollo de la base de datos que se ha incorporado a Moodle.* Se ha definido la información que se debe almacenar o re-estructurar para poder aplicar los modelos predictivos.
3. *Diseño de un bloque de Moodle que permite aplicar las técnicas predictivas de clasificación.* Se ha llevado a cabo el diseño e implementación del módulo, de tal forma que permite: hacer anónima la información de los alumnos (para facilitar la posterior publicación de resultados), seleccionar la información que se quiere utilizar para generar el modelo predictivo (A&Rs, estudiantes, fechas), introducir calificaciones de los alumnos (en caso de no estar utilizando el libro de calificaciones de Moodle), y seleccionar/configurar el algoritmo de clasificación que se desea aplicar.
4. *Instalación del software.* El software se ha instalado en un servidor propio.
5. *Validación del sistema.* Se ha llevado a cabo un proceso de validación para certificar y garantizar que el software está libre de errores y que lleva a cabo la tarea a desempeñar de manera correcta y eficiente.
6. *Análisis de la utilidad del bloque.* Tras desarrollar la nueva funcionalidad, se ha comprobado su utilidad en la evaluación y seguimiento de alumnos y A&Rs en la asignatura Introducción a la Programación del Grado en Ingeniería Informática. Para ello, una vez finalizado un curso académico y teniendo las calificaciones finales de los alumnos, se han aplicado los dos algoritmos de árboles considerados, *J48* y *RandomForest* y se evaluaron sus resultados. El análisis demuestra que los árboles de decisión obtenidos por los modelos, permiten determinar los factores más relevantes. Los resultados obtenidos, serán comentados en la siguiente sección, donde se muestra un caso de estudio.

### 4. DESCRIPCIÓN

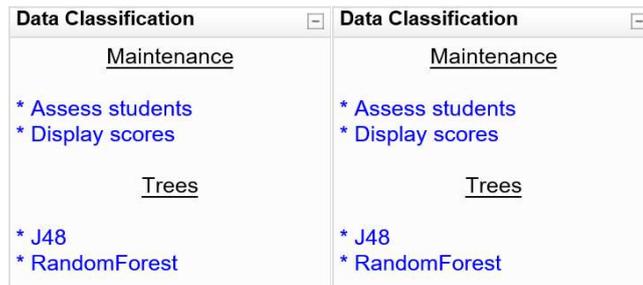
En este apartado se describen las características más relevantes del módulo desarrollado ilustradas con un caso de estudio en la asignatura Introducción a la Programación del Grado en Ingeniería Informática. La interfaz del módulo permite tanto el idioma inglés, como el español. En los ejemplos, se ha seleccionado tanto con la interfaz en inglés, como en español, para mostrar ambos funcionamientos.

#### 4.1 Descripción general del módulo

El módulo desarrollado (Figura 1) consiste en un bloque de Moodle que permite las siguientes opciones:

1. *Mantenimiento (maintenance).* Para obtener un modelo de clasificación de rendimiento académico es necesario disponer de una calificación para los alumnos considerados. En caso de que el profesor no haya utilizado el libro de calificaciones de Moodle, el módulo da la opción de introducir manualmente una calificación (*assess students*) y de visualizar las calificaciones (*display scores*). Estas calificaciones son totalmente independientes del libro de calificaciones de Moodle.
2. *Algoritmo basado en Árboles (trees).* Obtención de modelos predictivos mediante árboles de clasificación utilizando los algoritmos:
  - Algoritmo *J48*.
  - Algoritmo *RandomForest*.

1 <http://www.apachefriends.org/en/xampp.html>



1. Aspecto general del módulo.

#### 4.2 Módulo de Mantenimiento

Este módulo permite evaluar a los alumnos y visualizar las calificaciones. La primera de ellas, *evaluar alumnos*, permite evaluar a los estudiantes que no tienen calificación en el libro de calificaciones con la calificación de aprobado (*pass*) o suspenso (*fail*).

Podemos ver un ejemplo de esta funcionalidad en la Figura 2. En ella, se muestran todos los estudiantes pendientes de evaluación y se despliega un combo que permite calificarlos. En caso de error, también es posible recalificar a alumnos ya evaluados utilizando la parte inferior del formulario (*assessed students*).

Finalmente, la opción *reset* permite asignar el valor *no score* a la calificación de todos los alumnos del grupo.

**Assess students** You are logged in as [Administrador Laboratorio](#) (Logout)

[IP\\_MP](#) > [GII-1-PG2\\_C1](#) > Assess students

**Important**

Scores can be introduced here are completely independent of Moodle gradebook.

**Reset**

Reset scores No ▾

**Pending students to be assessed: 9**

Juan Luis	No score ▾
Daniel	No score ▾
Jose	No score ▾
Manuel	Pass ▾
Damian	Fail ▾
Inmaculada	No score ▾
Sergio	No score ▾
Pedro Javier	No score ▾
Javier	No score ▾
Ramon	No score ▾

**Students assessed: 69**

Oscar	Fail ▾
Pilar	Fail ▾
Antonio	Fail ▾
Mariano	Pass ▾
David	Pass ▾

**Assess students** You are logged in as [Administrador Laboratorio](#) (Logout)

[IP\\_MP](#) > [GII-1-PG2\\_C1](#) > Assess students

**Important**

Scores can be introduced here are completely independent of Moodle gradebook.

**Reset**

Reset scores No ▾

**Pending students to be assessed: 9**

Juan Luis	No score ▾
Daniel	No score ▾
Jose	No score ▾
Manuel	Pass ▾
Damian	Fail ▾
Inmaculada	No score ▾
Sergio	No score ▾
Pedro Javier	No score ▾
Javier	No score ▾
Ramon	No score ▾

**Students assessed: 69**

Oscar	Fail ▾
Pilar	Fail ▾
Antonio	Fail ▾
Mariano	Pass ▾
David	Pass ▾

## 2. Evaluar alumnos.

La segunda funcionalidad el módulo de mantenimiento es *visualizar calificaciones*. Esta funcionalidad (Figura 3) permite ver un listado de todos los alumnos de la asignatura junto con su correspondiente evaluación (*fail* o *pass*).

Display scores You are logged in as Administrador Laboratorio (Logout)

IP\_MP > GII-1-IPG2\_C1 > Display scores

Scores

Student	Score
Oscar	Fail
Pilar	Fail
Antonio	Fail
Mariano	Pass
David	Pass
Ignacio Javier	Fail
Barbara	Pass

Display scores You are logged in as Administrador Laboratorio (Logout)

IP\_MP > GII-1-IPG2\_C1 > Display scores

Scores

Student	Score
Oscar	Fail
Pilar	Fail
Antonio	Fail
Mariano	Pass
David	Pass
Ignacio Javier	Fail
Barbara	Pass

3. Visualizar calificaciones.

#### 4.3 Ajustes comunes del módulo

A continuación, se describe el módulo de clasificación desarrollado para la detección de recursos clave. Se describen primero los elementos comunes a los dos algoritmos considerados y después se hace una descripción detallada de cada uno de ellos. Cabe destacar que, al igual que hace Moodle, el usuario siempre tiene disponible información online sobre la utilización del módulo a través del icono de ayuda .

Uno de los aspectos comunes a los dos algoritmos considerados es la *selección personalizada de alumnos*. Esta opción se realiza mediante *check boxes* y permite además seleccionar a la vez todos los alumnos mediante el enlace *todos los alumnos* (Figura 4).

Opciones de visualización

[Mostrar nombre y apellidos del estudiante](#)  
[Mostrar nombre y apellidos cifrados del estudiante](#)  
[Mostrar índice del estudiante](#)

---

Seleccione los Alumnos

[Todos los Alumnos](#)

                   Oscar  
                    Pilar  
                    Juan Luis  
                    Daniel  
                    Jose Manuel  
                    Damian  
                    Inmaculada

---

Opciones de visualización

[Mostrar nombre y apellidos del estudiante](#)  
[Mostrar nombre y apellidos cifrados del estudiante](#)  
[Mostrar índice del estudiante](#)

---

Seleccione los Alumnos

[Todos los Alumnos](#)

                   Oscar  
                    Pilar  
                    Juan Luis  
                    Daniel  
                    Jose Manuel  
                    Damian  
                    Inmaculada

4. Selección personalizada de alumnos. Sin cifrado.

Otro aspecto común reseñable es el *cifrado de los nombres de los alumnos*. Esta funcionalidad permite mantener la privacidad de los alumnos en caso de que se quieran publicar los resultados obtenidos. Se permiten dos tipos de cifrado: numérico y encriptado. Así, la selección personalizada de alumnos da tres opciones. La primera consiste en mostrar el nombre y apellidos completos del estudiante, tal como se muestra en la Figura 4 (hemos ocultado los apellidos por motivos de privacidad). La segunda opción, consiste en cifrar los nombres de los estudiantes con el algoritmo MD5 (Figura 5). La última opción de cifrado consiste en asignar un índice a cada alumno y mostrarlos indicando su índice (*Estudiante1*, *Estudiante2*, etc.). Este último método genera nombres más intuitivos para el profesor que mediante MD5 (Figura 6).

<b>Opciones de visualización</b>	
<a href="#">Mostrar nombre y apellidos del estudiante</a> <a href="#">Mostrar nombre y apellidos cifrados del estudiante</a> <a href="#">Mostrar índice del estudiante</a>	
<b>Seleccione los Alumnos</b>	
<a href="#">Todos los Alumnos</a> <input type="checkbox"/> da2187a4c9277a885e40de738e467383 <input type="checkbox"/> 9a442410014d70101c087b2b2f8755df <input type="checkbox"/> 2ce2212703db37800070150d516a1d08 <input type="checkbox"/> dad9a20309712648f308e3a5a5950328 <input type="checkbox"/> 354b183b8036b83dfda18398d60438c8 <input type="checkbox"/> 7449ca9058000df5cba42d07a37d7253 <input type="checkbox"/> 9d78886588c6d9dd2060ededf63bc88d	
<b>Opciones de visualización</b>	
<a href="#">Mostrar nombre y apellidos del estudiante</a> <a href="#">Mostrar nombre y apellidos cifrados del estudiante</a> <a href="#">Mostrar índice del estudiante</a>	
<b>Seleccione los Alumnos</b>	
<a href="#">Todos los Alumnos</a> <input type="checkbox"/> da2187a4c9277a885e40de738e467383 <input type="checkbox"/> 9a442410014d70101c087b2b2f8755df <input type="checkbox"/> 2ce2212703db37800070150d516a1d08 <input type="checkbox"/> dad9a20309712648f308e3a5a5950328 <input type="checkbox"/> 354b183b8036b83dfda18398d60438c8 <input type="checkbox"/> 7449ca9058000df5cba42d07a37d7253 <input type="checkbox"/> 9d78886588c6d9dd2060ededf63bc88d	

5. Selección personalizada de alumnos. Cifrado MD5.

<b>Opciones de visualización</b>	
<a href="#">Mostrar nombre y apellidos del estudiante</a> <a href="#">Mostrar nombre y apellidos cifrados del estudiante</a> <a href="#">Mostrar índice del estudiante</a>	
<b>Seleccione los Alumnos</b>	
<a href="#">Todos los Alumnos</a> <input type="checkbox"/> Estudiante 1 <input type="checkbox"/> Estudiante 2 <input type="checkbox"/> Estudiante 3 <input type="checkbox"/> Estudiante 4 <input type="checkbox"/> Estudiante 5 <input type="checkbox"/> Estudiante 6 <input type="checkbox"/> Estudiante 7	

6. Selección personalizada de alumnos. Cifrado con índice.

<b>Opciones de visualización</b>	
<a href="#">Mostrar nombre y apellidos del estudiante</a> <a href="#">Mostrar nombre y apellidos cifrados del estudiante</a> <a href="#">Mostrar índice del estudiante</a>	
<b>Seleccione los Alumnos</b>	
<a href="#">Todos los Alumnos</a> <input type="checkbox"/> Estudiante 1 <input type="checkbox"/> Estudiante 2 <input type="checkbox"/> Estudiante 3 <input type="checkbox"/> Estudiante 4 <input type="checkbox"/> Estudiante 5 <input type="checkbox"/> Estudiante 6 <input type="checkbox"/> Estudiante 7	

El módulo de clasificación también permite la *selección personalizada de fechas*, que se realiza mediante despleables (Figura 7) y la *selección personalizada de A&Rs* (Figura 8). Realizada mediante *check boxes*, permite además seleccionar a la vez todos los recursos de un tipo concreto.

7. Selección personalizada de fechas.

8. Selección personalizada de A&Rs.

Además, es posible elegir *opciones de agrupamiento* (Figura 9). Así, se puede generar el modelo a partir del acceso y calificación del alumno respecto a A&Rs individuales (*grouping = No*) y Tipos de A&Rs (*grouping = Yes*).

9. Selección de opciones de agrupamiento.

10. Selección del tipo de información

Finalmente, cabe destacar que el módulo es capaz de generar el modelo a partir de *distintos tipos de información* (Figura 10):

- *Hits*. Numero de accesos (*hits*) a cada A&R.
- *Hits binary*. Si el alumno ha accedido o no a una determinada A&R independientemente del número de accesos.
- *Score*. En lugar del acceso, considera la puntuación. Según la configuración del agrupamiento y el tipo de A&R considera:

- *grouping = No*
  - Cuestionarios: la calificación obtenida.
  - Tareas: si se ha subido o no una tarea.
  - Foros y wikis: si se ha participado o no.
  - Consultas: si se ha realizado o no.
- *grouping = Yes*
  - Grupo de cuestionarios: la nota media.
  - Grupo de tareas: el número de tareas terminadas.
  - Grupo de foros o wikis: el número de posts.
  - Grupo de consultas: el número de consultas realizadas.
- *Hits & Scores*. Combina el método *Hits* y el método *Scores*.
- *Hits Binary & Scores*. Combina el método *Hits Binary* y el método *Scores*.

#### 4.4 Configuración de los algoritmos de clasificación

Se han incluido todos los parámetros que proporciona la herramienta weka para el algoritmo *RandomForest* (Figura 11):

- *MaxDepth*. Profundidad máxima de los árboles generados (0 para no poner límite a la profundidad).
- *NumFeatures*. Número de atributos del conjunto de datos original que serán seleccionados aleatoriamente para generar cada árbol.
- *NumTrees*. Número de árboles a generar.

*Seed*. Semilla para el generador de números aleatorios.

11. Configuración del algoritmo *Random Forest*.

Respecto al algoritmo *J48*, también se han incluido todos los parámetros que proporciona la herramienta weka (Figura 12):

- *SaveInstanceData*. Indica si los datos de entrenamiento se salvan para su posterior visualización.
- *Binary splits*. Indica si al generar el árbol se utilizarán particiones binarias en los atributos nominales.
- *Confidence factor*. Factor de confianza para la poda (pequeños valores producen mayor poda).
- *MinNumObj*. Número mínimo de instancias por hoja.
- *NumFolds*. En *reducedErrorPruning* un *fold* se utilizará para la poda y el resto para generar el árbol.
- *Unpruned*. Indica si se realizará poda del árbol o no.
- *SubtreeRaising*. Indica si se realizará *subtree raising* durante la poda.
- *ReducedErrorPruning*. Indica si se aplicará *reduced-error pruning* en lugar de la poda básica de C4.5
- *Seed*. Semilla para el generador de números aleatorios del proceso *reducedErrorPruning*.
- *useLaplace*. Indica si se utilizará en el conteo en las hojas del árbol la corrección de Laplace.

12. Configuración del algoritmo *J48*.

## 6. UTILIDAD/ANÁLISIS

Para mostrar la utilidad del módulo desarrollado se presenta un caso de estudio para la asignatura Introducción a la Programación de Grado en Ingeniería Informática. Esta asignatura incluye 6 cuestionarios que los alumnos deben ir realizando a lo largo del cuatrimestre, conforme se explica en clase la teoría necesaria para completarlos. Se enumeran en el mismo orden en que están secuenciados en la asignatura:

- Tipos de datos, operadores, expresiones, E/S
- Estructuras de control
- Modularización
- Tipos de datos compuestos
- Vectores y Matrices
- Matrices y estructuras

El objetivo de este caso de estudio es determinar cuáles de estos cuestionarios son recursos clave de la asignatura. Para ello, se han seleccionado como datos de entrada todos alumnos (cifrándolos por índice) y los 6 cuestionarios. Después, se ha ejecutado el algoritmo J48 con sus parámetros por defecto obteniendo el árbol de decisión de la Figura 13.

```
*Cuestionario modularización <= 5: Fail (22.0)
*Cuestionario modularización > 5
| *Cuestionario tipos de datos, operadores, expresiones E/S > 4.67: Pass (3.0)
| *Cuestionario tipos de datos, operadores, expresiones E/S <= 4.67
| | *Cuestionario modularización > 7.33: Pass (18.0/7.0)
| | *Cuestionario modularización <= 7.33: Fail (22.0/3.0)
```

13. Ejemplo de árbol de decisión.

De este árbol de decisión se puede deducir que:

- De los 6 cuestionarios considerados son dos los que pueden considerarse recursos clave: *modularización* y *tipos de datos, operadores y expresiones, E/S*.
- El cuestionario que además parece más determinante es el de *modularización*, debido a que si se suspende el correspondiente a tipos de datos, se requiere de una calificación relativamente alta (7.33) para superar la asignatura.

Como se puede apreciar, este modelo permite que el profesor obtenga conocimiento que puede ayudar a detectar alumnos en situación de riesgo de suspender la asignatura y poder así tomar medidas preventivas. Por ejemplo, todos los alumnos deberían tener muy claros los conceptos vistos en *modularización*.

```
@relation J48
@attribute "*Cuestionario tipos de datos, operadores, expresiones E/S" numeric
@attribute "*Cuestionario estructuras de control" numeric
@attribute "*Cuestionario modularización" numeric
@attribute "*Cuestionario tipos de datos compuestos (homogéneos y heterogéneos)"
numeric
@attribute "*Cuestionario vectores y matrices" numeric
@attribute "*Cuestionario matrices y estructuras" numeric
@attribute "Score" {"Pass","Fail"}

@data
7.33,8,7,0,8.17,6,"Fail" %Student 1
8,8,8,0,8,7,"Fail" % Student 2
7.33,4,9.33,0,0,8.5,"Fail" % Student 3
7.33,8,6.67,0,8,7.67,"Pass" % Student 4
8,6,7.33,0,7.5,7.33,"Pass" % Student 5
```

14. Ejemplo de dataset en formato *arff*.

Cabe destacar que el módulo da la opción de guardar los datos utilizados en el formato *arff* de weka, por lo que se facilita que puedan ser posteriormente procesados directamente desde la herramienta weka. La Figura 14 muestra un extracto del fichero que se generaría. Debido a que los nombres de los alumnos fueron cifrados, en el fichero *arff* no aparecen sus nombres, sino su alias. No obstante, el profesor tiene accesible desde el módulo la correspondencia entre los nombres cifrados y los reales (Figura 15).

Index	Name	Index	Name
Student 1	[redacted] Oscar	Student 1	[redacted] Oscar
Student 2	[redacted] Pilar	Student 2	[redacted] Pilar
Student 3	[redacted] Antonio	Student 3	[redacted] Antonio
Student 4	[redacted] Mariano	Student 4	[redacted] Mariano
Student 5	[redacted] David	Student 5	[redacted] David

15. Correspondencia entre los nombres de alumnos y los nombres cifrados.

## 7. CONCLUSIONES/DISCUSIÓN

Se puede concluir que se ha cumplido con todos los objetivos planteados. De este modo, es posible obtener información relevante (conocimiento), a partir de la información almacenada en la base de datos de Moodle aplicando algoritmos de clasificación basados en árboles. Con esta información, que hasta ahora estaba inaccesible a los profesores de forma directa, se puede dar respuesta a preguntas como “¿cuáles son los recursos más relevantes de nuestras asignaturas?”, que a su vez, permite detectar situaciones que podrían conducir al alumno a suspender la asignatura, y poder actuar en consecuencia. Así, una vez tengamos suficiente actividad de los alumnos en la plataforma y una calificación, el uso de este módulo puede determinar las A&Rs que podrían mejorar el rendimiento académico en los sucesivos cursos.

## AGRADECIMIENTOS

Este trabajo ha sido financiado con el proyecto de innovación educativa de la Universidad de Córdoba 2015-2-5005.

## BIBLIOGRAFÍA

- BONEU, J. M. (2007) “Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos”. *Universities and Knowledge Society Journal*, 4(1), p. 1 - 8.
- BREIMAN, L. (2001) “Random forests”. *Machine learning*, 45(1), p. 5-32.
- FRAWLEY, W. J., PIATETSKY-SHAPIRO, G., MATHEUS, C. J. (1992) “Knowledge discovery in databases: An overview”. *AI magazine* 13(3), p. 49 - 57.
- HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P., WITTEN, I. H. (2009) “The WEKA data mining software: an update”. *ACM SIGKDD explorations newsletter* 11(1), 10-18.
- LUQUE, M., GIBAJA E., ZAFRA, A., VENTURA, S. (2014) “Proyecto de innovación docente para la autoevaluación y mejora de la metodología docente usando encuestas personalizadas en Moodle”. In *P. I Seminario Iberoamericano de Innovación Docente de la Universidad Pablo de Olavide*. Sevilla, 20 y 21 de noviembre 2014. ISBN: 84-697-2223-9.
- QUINLAN, J. R. (1993) *C4. 5: programs for machine learning*. Morgan Kaufmann Publishers, San Mateo, CA.
- ROMERO, C., VENTURA, S. (2010) “Educational data mining: a review of the state of the art”. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 40(6), p. 601-618.
- ROS MARTÍNEZ DE LAHIDALGA, I. (2008) “Moodle, la plataforma para la enseñanza y organización escolar”. *Ikastorratza, e-Revista de didáctica* 2, p. 3-12.
- ZAFRA, A., GIBAJA E., LUQUE, M., VENTURA, S. (2014) “ExportQuiz: Módulo de Moodle para exportar cuestionarios a ficheros de texto”. In *P. I Seminario Iberoamericano de Innovación Docente de la Universidad Pablo de Olavide*. Sevilla, 20 y 21 de noviembre 2014. ISBN: 84-697-2223-9.