UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

TRABAJO FIN DE MÁSTER

Propuesta de un método basado en Deep Learning para Learning Analytics en MOOCs

Máster Universitario en Investigación en Innovación en las Tecnologías de la Información y las Comunicaciones

Autor: ISIDRO ESTRADAS, Cristina

Tutor: CARRO SALAS, Rosa María

Septiembre 2017

Resumen

Es notable la relevancia que están consiguiendo los cursos MOOC (*Massive Online Open Courses*) en el ámbito de la educación y el aprendizaje online. Estos cursos abren la puerta a una nueva era para el aprendizaje, ya que ponen accesible una gran cantidad de conocimiento a personas que pueden no tener los recursos necesarios para adquirirlo de otro modo. Po ello, el número de estudiantes que se matriculan en un MOOC puede alcanzar un número gigantesco.

Debido al carácter abierto que tienen los MOOCs, los estudiantes pueden llevar a cabo su proceso de aprendizaje de forma independiente, cada uno siguiendo su propio ritmo, y para los profesores puede resultar muy complicado llevar a cabo un seguimiento individualizado de la evolución de cada uno de ellos. Habitualmente, el porcentaje de estudiantes que aprueban estos cursos es mucho menor que el de quienes no lo consiguen. Además, las tasas de abandono suelen ser bastante más altas que en la docencia presencial. Sería interesante, por tanto, monitorizar la actividad y evolución de cada estudiante a lo largo del tiempo, de modo que por una parte, los profesores puedan tener algún tipo de retroalimentación y tomar medidas u ofrecer refuerzo a los estudiantes en caso necesario. Por otra parte, se podría considerar la posibilidad de intervenir o enviar mensajes de alerta a aquellos estudiantes para los cuales se detecten dificultades o se prevea que pueden abandonar los cursos.

Las interacciones de los estudiantes con los MOOCs quedan almacenadas y pueden ser consultadas. Por cada estudiante se genera un número muy grande de datos que pueden ser analizados para conocer la evolución del estudiante y tratar de detectar o predecir el riesgo de suspenso o abandono.

En este Trabajo de Fin de Máster se muestra cómo se puede analizar la información disponible sobre las interacciones de los estudiantes con los MOOCs para tratar de detectar o predecir situaciones de riesgo como el suspenso o abandono. Además, se analizará si los algoritmos de *Deep Learning* pueden aportar una mejora en comparación con los algoritmos convencionales de aprendizaje automático, en el ámbito de *Learning Analytics*.

Abstract

It is noticeable the relevance that are gaining some MOOC (*Massive Online Open Courses*) in the education and online learning fields. These courses open the door to a new era for learning, because they make a great deal of knowledge accessible to people who may not have the necessary resources to acquire them in any other way. This means that the number of students that get enrolled in a MOOC can reach a gigantic number. If we take into account that each student generates a very large amount of data, the quantity of information that can be extracted from its analysis is immense.

On the other hand, due to the open nature of the MOOCs, students can carry out their learning independently and it can be very difficult for the teachers to keep track of them. If we compare the students who pass these courses and those who do not, it is observed that the percentage of the latter group is very high. Therefore, it is necessary to analyse in some way the activity that the users of the course have, so that the teachers can have feedback and take measures or reinforcement for the students.

Therefore, in this Master's Dissertation, a method is proposed in order to provide feedback to the teachers in the context of online learning systems and in particular, of MOOCs. In addition, it will be analysed if the algorithms of Deep Learning can provide an improvement compared to the conventional automatic learning algorithms, in the field of Learning Analytics.

A la vida.

Por poner siempre cada cosa en su lugar.

AGRADECIMIENTOS

En primer lugar, agradezco a Rosa Carro que me acogiese como tutora y ofrecerme así la oportunidad de cursar este máster. Y junto con ella, agradecer a Álvaro Ortigosa por la dedicación que han tenido por guiarme en este trabajo.

A Juanda, por ser tercer tutor, amigo y tan buena persona. Gracias por estar siempre dispuesto a ofrecer una mano.

A Tania, por ser mi compañera de viaje. Por animarme siempre y mostrarme una sonrisa en los malos momentos. Gracias por estar siempre ahí. Te quiero.

Y, finalmente, a mis padres por ser un ejemplo a seguir, por animarme en todas las noches que me he quedado despierta y por poner tanto interés en lo que hago.

Gracias a todos.

Cristina.

Índice general

Índice de Figuras

Índice de Tablas

1 Introducción

1.1 Motivación

Con el paso de los años es mayor la importancia y relevancia que van adquiriendo los MOOCs (*Massive Online Open Courses*) en el ámbito de la educación [1]. La gran mayoría de universidades del mundo ofrece distintos cursos totalmente gratuitos en forma de MOOCs y cada vez son más las personas que se matriculan en ellos. Estos cursos ofrecen un sinfín de oportunidades a personas que no tienen los recursos necesarios para cursar estudios que impliquen una inversión económica.

Por otro lado, el que los MOOCs sean cursos online de carácter abierto implica que los estudiantes tienen total libertad para la realización de los mismos. En algunas ocasiones los cursos pueden estar abiertos en cualquier momento y en otras el temario va siendo accesible conforme los profesores lo consideran oportuno. En cualquier caso, el estudiante puede optar por adquirir los conocimientos sin someterse a ningún examen o, en cambio, examinarse y obtener la certificación de la realización de ese curso. Esto crea una posible situación de "abandono", que se considera que existe cuando el estudiante no termina las actividades del curso y no consigue aprobarlo. Sin embargo, esta situación de "abandono" puede darse no solo porque de verdad el estudiante no haya mantenido el interés en la realización de las actividades del curso, sino también porque no haya considerado la necesidad de examinarse, bastándole con aprender los conocimientos de una forma independiente. Por otra parte, de la gran cantidad de estudiantes que interactúan con un MOOC, cada uno lo hace a su ritmo. Sería deseable poder conocer su evolución, detectar y predecir el riesgo de fracaso o abandono.

Cuando un estudiante de un MOOC resuelve de manera errónea un ejercicio, es complicado saber si estaba interesado realmente en el ejercicio, pero no lo pudo resolver correctamente o si “pasaba por allí” y pulsó cualquier opción para ver qué ocurría. Para evitar confusiones es necesario analizar la actividad de los usuarios en el curso y poder identificar por un lado los que suspenden porque de verdad han abandonado y por otro los que suspenden por no examinarse aun teniendo interés en aprender con el curso. Si al profesor se le proporciona retroalimentación en el momento sobre situaciones de riesgo, se le ofrece la posibilidad de ayudar a los estudiantes que quieren abandonar o terminarán abandonando, e incluso a

aquellas que tienen interés pero que se prevé que, dada su evolución, tienen difícil aprobar, para que remonten y aprovechen la oportunidad de aprender e incluso de certificarse.

Debido a la gran actividad que puede realizar un estudiante en un MOOC, durante la interacción de los estudiantes con los MOOCs se genera una gran cantidad de información que se puede analizar. Es posible extraer diversas variables o atributos relevantes para realizar un análisis del comportamiento de estos. Este análisis se denomina análisis del aprendizaje, más comúnmente conocido como *Learning Analytics (LA)*y ha cobrado una gran importancia en los últimos años [2]. El análisis de los datos producidos por diferentes plataformas de enseñanza online puede constituir una potente herramienta que abra la puerta para mejorar la forma de enseñar de hoy en día a través de estas plataformas.

Por otro lado, en pleno auge de la investigación en el ámbito del aprendizaje automático, está clara también la importancia y utilidad de estas técnicas para la toma de decisiones basadas en miles y miles de datos. Aparece un nuevo concepto, el *Deep Learning* [3] [4] que se describirá en secciones posteriores y que permite, entre otras cosas, identificar patrones. Es por ello que se hace interesante analizar la actividad de los estudiantes con esta clase de algoritmos de aprendizaje.

En este TFM se plantea la posibilidad de analizar una gran cantidad de atributos extraídos de esta información mediante algoritmos de *Deep Learning*. El uso de algoritmos de *Deep Learning* en el ámbito de *Learning Analytics* no está muy explorado, o al menos no se tiene constancia. En este caso se ha considerado el uso de estos algoritmos justamente por la capacidad que tienen de encontrar patrones en una gran cantidad de datos. Además, ya que no se tiene constancia de trabajos que exploren esta línea, se pretende investigar si los algoritmos de *Deep Learning* proporcionan una mejora en comparación con los algoritmos convenciones de aprendizaje automático que se suelen usar en el ámbito de *Learning Analytics.*

1.2 Objetivos

1.2.1 Objetivo principal

Como se ha comentado anteriormente, el abandono de estudiantes en los MOOCs es masivo. También hay un gran número de estudiantes que no llega a superar las pruebas de evaluación necesarias para aprobar el curso. Esto puede deberse a diferentes factores, ya sean personales

o relacionados con el curso. En este TFM se explora la posibilidad de ofrecerle ayuda al profesor para detectar posibles riesgos de suspenso o abandono por parte de los estudiantes y, además, detectar en qué punto de los cursos pueden existir factores que favorezcan dicho abandono.

Así pues, el objetivo principal de este trabajo es proponer un método para ofrecer retroalimentación o *feedback* sobre el desempeño de los estudiantes de un curso online abierto y masivo (MOOC) y sobre los indicios de problemas o situaciones indeseables identificados para ese curso.

1.2.2 Objetivos parciales

Para alcanzar el objetivo principal del TFM se han planteado los siguientes objetivos parciales:

O.1 Procesado de los registros de los eventos generados por la actividad del usuario en el curso para extraer variables relevantes para el análisis a realizar.

O.2 Identificación de factores de riesgo de abandono o suspenso.

O.3 Elaboración de una propuesta para detectar y predecir situaciones de riesgo para el estudiante.

O.4 Comparación del desempeño y porcentaje de éxito de algunos **algoritmos convencionales de aprendizaje con respecto a la utilización de algoritmos de *Deep Learning.***

O.5 Análisis de los resultados de la aplicación del método propuesto en un curso concreto.

1.3 Contexto del Trabajo de Fin de Máster

Este Trabajo Fin de Máster se enmarca en el contexto de la obtención de información de entornos de enseñanza online del tipo cursos online masivos y abiertos (MOOCs), y tiene el principal propósito de identificar indicios de posibles problemas o situaciones indeseables que puedan producirse, mediante el análisis de los datos obtenidos de los estudiantes.

1.4 Estructura de la memoria

En este apartado se detallan la estructura de la memoria y el contenido de cada capítulo. Se compone de 5 capítulos y 2 anexos:

- En el **capítulo 1** se plantea un problema y se pone en contexto el TFM. Después se describen el objetivo principal y los objetivos parciales, se contextualiza el proyecto y se describe la estructura de este documento.
- En el **capítulo 2** se habla de los conceptos en los que se apoya este TFM. En primer lugar, se comentará de qué trata el ámbito de *Learning Analytics* y los MOOCs. Después se hablará del *Deep Learning*.
- En el **capítulo 3** se realiza un análisis del conjunto de datos del que partimos y el enfoque que se tomará para los análisis.
- En el **capítulo 4** se analizan los resultados obtenidos en los diferentes análisis realizados.
- En el **capítulo 5** se exponen las conclusiones obtenidas tras la realización de este Trabajo Fin de Máster y comentan los posibles trabajos futuros.
- En el **capítulo 6** aparece la bibliografía consultada.
- En el **anexo I** se muestran las diferentes distribuciones de las variables extraídas.
- En el **anexo II** se muestran los árboles de decisión generados para el análisis por semanas.

2 Estado del arte

Este trabajo se enmarca en el ámbito del análisis del aprendizaje, más comúnmente conocido como *Learning Analytics*, y se centra concretamente en análisis del aprendizaje en MOOCs, explorando distintas técnicas de aprendizaje automático, entre ellas *Deep Learning*. En esta sección se introducen los conceptos y definiciones más relevantes en este contexto, así como el estado del arte actual en este ámbito.

2.1 Learning Analytics

Existen muchas definiciones que se han ido debatiendo conforme se ha ido investigando en este ámbito a lo largo del tiempo. Finalmente se concluye que el concepto de *Learning Analytics* hace referencia al análisis y presentación de datos sobre estudiantes y sus contextos personales con el fin de entender su aprendizaje y poder mejorarlo [5]. Este análisis puede proporcionar información de diferentes asuntos de interés sobre el estudiante y el contexto en el que se encuentra aprendiendo. Desde un punto de vista del aprendizaje que se lleva a cabo a través de los MOOCs, esta información puede estar relacionada con la sucesión de actividades que lleva a cabo a lo largo de una sesión online, el tiempo que ha estado activo en el MOOC, los resultados de las actividades que ha realizado, la relación que tiene con otros estudiantes que se encuentran realizando el mismo curso u otra serie de datos que pueden resultar de interés para el análisis.

Dentro de este ámbito existe la necesidad de utilizar distintos métodos computacionales para realizar el análisis necesario para tratar de entender qué ocurre durante el proceso de aprendizaje de los estudiantes a partir de todos los datos recopilados sobre sus interacciones con los MOOCs. Estos métodos toman distintos enfoques según el contexto que se quiera analizar. En este trabajo se van a comentar tres de ellos, ya que suponen los más interesantes.

- *Network-Analytic Methods*

Estos métodos se enfocan en las relaciones que existen entre los distintos actores. Los actores se toman como nodos y la relación entre ellos se representa mediante arcos, dando lugar a grafos [6]. El tipo de conexión que existe entre un par de actores define la naturaleza de cada relación. Existen diferentes tipos de lazos entre actores, como lazos de amistad, profesionales o incluso enfocados a la información que comparten entre ellos sobre algún tema. En el contexto de este trabajo, es necesario destacar que estos "actores" que forman los nodos no

tienen por qué ser necesariamente estudiantes. También se pueden representar como nodos los distintos elementos del curso, y analizar las relaciones que existen entre ellos y que puedan proporcionar información interesante para el análisis del aprendizaje [6].

- *Process-Oriented Interaction Analysis*

Este enfoque se basa en analizar las interacciones que tiene un actor o estudiante mientras está realizando el curso [7]. Estas interacciones se registran en los archivos de *log* que genera la plataforma en la que está desarrollado el curso. Al analizar estos archivos se pueden tratar de detectar patrones que definan el comportamiento de un estudiante durante el proceso de aprendizaje [2].

- *Content Analysis Using Text-Mining Methods*

Este método se basa en el análisis del contenido textual que generan los estudiantes durante la realización de los cursos. Este contenido hace referencia, por ejemplo, a los comentarios que escriben en los foros acerca de un determinado tema o sobre una duda concreta. Con este tipo de análisis no solo es posible analizar la naturaleza del contenido textual, sino que además se puede extraer información semántica relacionada con los temas de un curso, pudiendo así clasificarlos o, por ejemplo, extraer relaciones entre las palabras de varios temas diferentes [8].

Estos tres enfoques ofrecen la posibilidad de analizar y entender la actividad llevada a cabo por los estudiantes en un contexto de aprendizaje no presencial.

En la red eMadrid [9] se han realizado trabajos relacionados con el análisis del aprendizaje. Por ejemplo, se ha propuesto una metodología para el diseño y desarrollo de MOOCs basada en las mejores prácticas analizadas en los primeros MOOCs implementados en las plataformas edX [10] y MiriadaX [11]. Por otro lado, también se ha trabajado en modelos de predicción y clustering, análisis de interacción social durante el proceso de aprendizaje y evaluación de diferentes experiencias educativas. Otro ejemplo de trabajo es realizado en el Amrita e-Learning Research de la Universidad de Amrita en India, donde el objetivo es encontrar la manera de identificar estudiantes que están en riesgo de no certificarse en un curso [12]. Este tipo de trabajos se enfoca principalmente en encontrar patrones relevantes utilizando reglas de asociación [13].En este trabajo nos vamos a centrar, como ya se ha comentado anteriormente, en el análisis del aprendizaje en cursos MOOCs desarrollados en la plataforma edX. Se quiere poder detectar aquellos casos en los que los estudiantes se encuentren en una situación que pueda poner en riesgo la obtención de la certificación para

dicho MOOC o que pueda suponer un riesgo de abandono.

2.1.1 MOOCs

MOOC es el acrónimo en inglés de *Massive Online Open Course* (o curso online masivo y abierto) [1]. Esto quiere decir que es un curso a distancia, accesible por internet para cualquier persona y casi sin límite de participantes. Algunos pueden estar abiertos continuamente y otros pueden ir abriéndose poco a poco, tal y como lo consideren los profesores que lo imparten. Se compone de un temario, vídeos, ejercicios individuales y/o grupales, exámenes, etc. Además de estos componentes, también pueden contener foros de debate y ayuda que proporcionan la posibilidad de construir una comunidad de estudiantes y profesores. Su carácter masivo hace que sea una potencial fuente de datos para realizar diferentes análisis del comportamiento de los usuarios durante la realización del curso. Además, la cantidad de usuarios que pueden acceder a estos cursos hace que sea muy interesante realizar análisis de la interacción social entre los estudiantes y profesores. Por otro lado, su carácter gratuito hace que sea muy llamativo para un número gigantesco de personas que desean aprender sobre un tema o ámbito sin necesidad de invertir económicamente en ello.

Muchas son las plataformas que alojan todo tipo de MOOCs como INTEF, Miriada X [11], Udacity [14], Udemy [15], Coursera [16] y, como en el caso del MOOC que analizamos en este TFM, edX [10].

2.2 Deep Learning

El aprendizaje profundo, *Deep Learning* en inglés, es un conjunto de algoritmos de aprendizaje automático [17]. Estos algoritmos utilizan estructuras lógicas que se asemejan en mayor medida a la organización del sistema nervioso de los mamíferos, teniendo capas de unidades de proceso que imitan el funcionamiento de las redes neuronales y que se especializan en detectar determinadas características existentes en los objetos percibidos. Una red neuronal se caracteriza por ser un conjunto de neuronas conectadas entre sí y que trabajan juntas. Conforme van ganando experiencia, las neuronas refuerzan las conexiones entre sí y "aprenden". Este enfoque nos permite que dentro del sistema haya redes de procesos que se especialicen en detectar determinadas características ocultas en los datos.

La arquitectura de este tipo de redes neuronales se basa en una serie de neuronas organizadas en diferentes capas, según la función que tenga cada una de ellas. En la figura se pueden observar tres tipos de capas: la de entrada, que se encarga de recibir los datos necesarios para el análisis; las neuronas de la capa oculta, que se encargan de procesar y realizar cálculos internos de la red; y, por último, la capa de salida contiene neuronas que se encargan de recibir el resultado final de los cálculos. Toda neurona dentro de una capa tiene una conexión con una neurona de la capa siguiente. Estas conexiones tienen asociado un número llamado peso.

En este trabajo se analiza la posibilidad de aplicar algoritmos basados en redes neuronales para tratar de detectar patrones de comportamiento entre los estudiantes que han cursado un MOOC. Para ello es necesario decidir qué datos de entrada se van a proporcionar a la red neuronal, así como el número de neuronas y capas que va a tener dicha red. Estas y otras cuestiones se analizarán en el apartado siguiente.

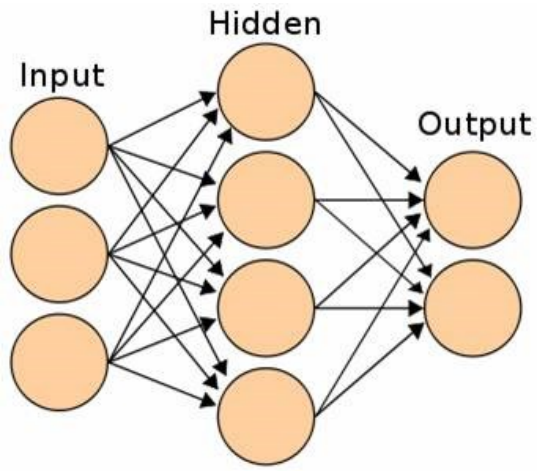

Figura 1. Arquitectura de una red neuronal. (Figura extraída de [18]**)**

3 Análisis de los datos

3.1 Conjunto de datos

Para este Trabajo Fin de Máster se nos ha proporcionado una serie de archivos correspondientes al MOOC *La España del Quijote* de la Universidad Autónoma de Madrid, alojado en la plataforma edX. A continuación, se describe el contenido de los distintos archivos:

- **ProfileQuijote501x1T2015**

Este archivo, con formato .csv, contiene datos demográficos de todos los usuarios registrados hasta el momento en el curso. Además, contiene información sobre su biografía y la foto de perfil.

- **CertificateQuijote501x1T2015**

Este archivo, con formato .csv, contiene información sobre la nota final de cada usuario en el curso.

- **SocialQuijote501x1T2015**

Este archivo, con formato .csv, contiene datos sobre la actividad social de los usuarios en el curso. Esto se refiere esencialmente a los comentarios escritos por cada usuario en los distintos foros disponibles.

- **Registro de los eventos**

Estos registros son archivos con formato .json donde se registran los eventos generados por la actividad del usuario durante su interacción con el curso. Por cada día se genera un registro con la actividad de todos los usuarios que se han conectado al curso en ese día. Para este TFM se cuenta con un total de 728 archivos generados desde el 29 de septiembre de 2014 hasta el 14 de febrero de 2017.

3.1.1 Estructura de los registros

A continuación se presenta la estructura de los registros de los datos utilizados para el análisis realizado en este TFM. La plataforma edX genera una serie de eventos que recogen

la actividad del usuario. Existen 21 tipos de eventos que pueden generar los estudiantes relacionados con las diferentes partes del curso.

La estructura interna de cada evento dependerá del tipo de evento que sea. Aun así, tienen en común elementos como la identificación del usuario, el agente que lanza el evento, el evento generado, el tipo de evento y el instante en que se produce (*user_id, agent, event, event_type y time*). Los eventos generados por la interacción de los estudiantes con los vídeos del curso han sido utilizados para extraer información sobre el número de vídeos con los que el estudiante ha tenido interacción (*n_videos*), el tiempo de video en segundos que ha visto (*t_*videos) y el número de movimientos hacia delante y hacia atrás que ha tenido el estudiante en los videos (*n_mov_haciadelante y n_mov_haciaatras.* De la interacción de los estudiantes con los diferentes ejercicios y exámenes, se ha extraído información sobre el número de aciertos (*n_aciertos*) e intentos (*n_intentos*) y el número de problemas (*n_problemas*). Cada tema del curso tiene un foro asociado donde los estudiantes pueden añadir comentarios. De estos eventos se ha extraído el número de comentarios (*n_comentarios).* Del registro de los eventos de cada estudiante se ha podido extraer la variable que indica el número de eventos (*n_eventos).* Una vez obtenido el número de eventos se ha llevado a cabo un proceso para extraer el número de sesiones que ha llevado a cabo el usuario, analizando los tiempos e intervalos entre los eventos (y se ha definido la variable *n_sesiones*). Finalmente, la variable clase *aprobado* se extrae del archivo .csv *CertificateQuijote501x1T2015* descrito anteriormente.

3.2 Enfoque de los modelos

Uno de los objetivos de este TFM es analizar los patrones de la actividad de los estudiantes de los MOOCs en relación con los resultados que obtienen, de forma que permita proporcionar *feedback* al profesor de cara a la próxima edición del curso o incluso para el mismo curso que están realizando. Principalmente se pretende poder predecir cuáles serán los resultados de un estudiante según su evolución durante la realización del curso, para poder tomar las decisiones o realizar las intervenciones que se consideren oportunas al respecto. Así pues, se ha decidido analizar poco a poco la trayectoria del estudiante para, de esta forma, poder alertar (al profesor o al propio estudiante) cuando se detecta un estudiante presenta riesgo de suspender, quedarse estancado o abandonar el curso. Para poder realizar estas predicciones y poder intervenir durante la propia realización del curso, se ha decidido tomar la semana como unidad de medida de tiempo a la hora de analizar los eventos y las

diferentes trayectorias de los estudiantes. Teniendo en cuenta un número máximo de semanas para realizar el curso, se diseñará un modelo por cada semana, correspondiente a la actividad acumulada desde el inicio del curso hasta la dicha semana. Así mismo, también se realizará un análisis de los valores totales de cada estudiante para tener una visión global de su desempeño. Además, se investigará, según los patrones encontrados en los estudiantes, posibles puntos del curso en los que los estudiantes se hayan podido quedar estancados, si fuera el caso.

La figura 2 muestra la distribución de los estudiantes que aprobaron el curso en función del número de semanas que necesitaron para terminarlo. Cabe mencionar que este MOOC está abierto y sin límite de tiempo. Como se puede apreciar, hay mucha diversidad, estando el punto más alto en más de 100 usuarios que tardaron entre 0 y 10 semanas. El grueso está entre las 0 y las 42 semanas. A partir de ese valor, ya son pocos los estudiantes que necesitan más semanas para terminarlo.

Para definir el número de semanas que se van a considerar para la generación de los modelos y, en consecuencia, el número de modelos que se van a generar, se ha observado el grueso de la figura anterior y se ha decidido considerar 42 semanas, que dará lugar a la generación de 42 modelos

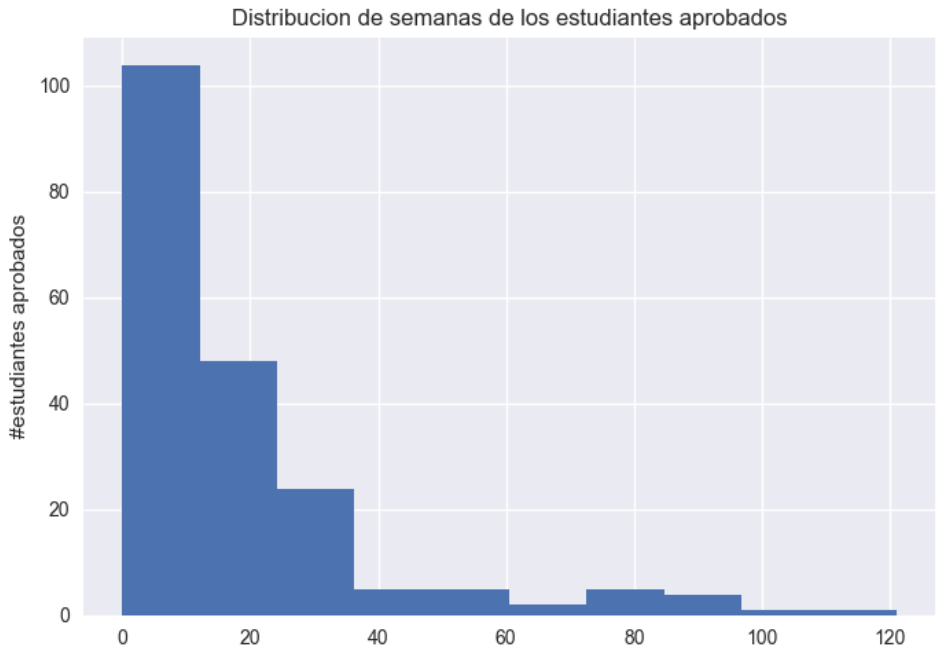

Figura 2. Distribución del tiempo tardado en semanas en completar el MOOC.

3.3 Tecnología utilizada

En este apartado se van a mencionar las tecnologías y lenguajes de programación que se han utilizado y su finalidad.

Python [19]

Lenguaje de programación utilizado para el desarrollo del código encargado de procesar los *logs* con el fin de extraer las variables relevantes para este estudio. También se ha utilizado Python para la implementación de los diferentes algoritmos aprendizaje automático.

Tensorflow [20]

Es una librería de código abierto para aprendizaje automático. Se ha utilizado principalmente para la implementación del algoritmo de *Deep Learning*.

Scikit-learn [21]

Librería con recursos de aprendizaje automático para el lenguaje Python. Se ha utilizado para la implementación de los algoritmos convencionales de aprendizaje automático, es decir: Näive Bayes, Árboles de decisión y SVM.

Sublime Text [22]

Editor de texto utilizado para la implementación de los algoritmos de aprendizaje automático.

Pycharm [23]

Entorno de desarrollo para Python utilizado para la implementación del código encargado del procesamiento de los logs.

Microsoft Word [24]

Procesador de texto utilizado para la realización de esta memoria.

Zotero [25]

Gestor de referencias utilizado para la realización de esta memoria.

4 Resultados

En esta sección se presentan los diferentes resultados que se han obtenido. En primer lugar, se especifica cuáles son las variables extraídas de los *logs* que contienen los eventos de los estudiantes, se comentan sus características y se muestra su correlación con la variable clase *aprobado*. En segundo lugar, se muestran los resultados de la clasificación supervisada que se ha llevado a cabo. Se presentan los resultados de los cuatro algoritmos utilizados y se realiza una comparación entre ellos. Por último, se muestran los resultados obtenidos del análisis, mediante árboles de decisión, de la progresión de los estudiantes a lo largo de las distintas semanas del curso.

4.1 Extracción de las variables

En primer lugar, se ha realizado la extracción de las variables o atributos que van a componer los diferentes modelos de predicción. Para ello, con ayuda de distintas librerías de Python, se ha llevado a cabo un procesamiento de los **729 registros** de los eventos de los estudiantes.

Como se comentó en la sección anterior, se han analizado los diferentes eventos que puede generar un estudiante para así poder extraer variables con cierta relevancia para el análisis. A continuación, se indican las **14 variables** que se han extraído:

- *n_videos:* número total de vídeos que el estudiante al menos ha visitado. El evento se genera en el instante en que visita el vídeo.
- *t_videos*: tiempo total en segundos que un usuario ha estado viendo vídeos del curso.
- *n_mov_haciadelante:* número de movimientos hacia delante que ha realizado un usuario mientras visualizaba un vídeo.
- *n_mov_haciaatras:* número de movimientos hacia atrás que un usuario ha realizado mientras visualizaba un vídeo.
- *n_aciertos:* número de aciertos que tiene un usuario en total en los exámenes y ejercicios del curso.
- *n_intentos:* número de intentos de un usuario durante la realización de los exámenes y ejercicios de un curso (puede realizar varios intentos por ejercicio).
- *n_problemas:* número de ejercicios que ha realizado un usuario.
- *n_comentarios:* número de comentarios que ha realizado un usuario en los foros del curso.

- *n_eventos:* número total de eventos que un usuario ha generado durante su interacción con el curso.
- *n_sesiones:* número total de sesiones en las que se ha conectado un usuario.
- *aprobado:* indica si un usuario ha aprobado o no el curso. Será la clase para la clasificación supervisada.

La extracción de los atributos se ha realizado de cara a analizar las causas o las razones por las que un usuario aprueba un curso o no. A continuación se muestra un análisis de los valores de estas variables.

4.1.1 Análisis de los valores de las variables

Con el objetivo de dar a conocer los valores de los distintos atributos en el MOOC analizado, a continuación, se muestra la información en dos tablas. En la primera de ellas se muestra, para cada uno de los atributos *n_videos, t_videos, n_mov_haciadelante, n_mov_haciaatras y n_aciertos*, el valor medio, la desviación típica, el mínimo, el máximo y el valor del atributo para distintos porcentajes de estudiantes. Por ejemplo, la media de vídeos vista por los estudiantes es 5.59, siendo la desviación típica 14.43. El estudiante que menos vídeos ha visto no ha accedido a ninguno, y el que más ha visto ha visualizado 70. El 25% de los estudiantes no ha visto ningún video, el 50% tampoco ha visto ninguno y el 75% ha visto 3.

	n_videos	*t_videos*	*n_mov_haciadelante*	*n_mov_haciaatras*	*n_aciertos*
Mean	*5.591709*	*2910.398747*	*3.196839*	*2.891441*	*7.885774*
Std	*14.433206*	*12639.266261*	*47.263032*	*15.465881*	*27.482177*
Min	*0.000000*	*0.000000*	*0.000000*	*0.000000*	*0.000000*
25%	*0.000000*	*0.000000*	*0.000000*	*0.000000*	*0.000000*
50%	*0.000000*	*0.000000*	*0.000000*	*0.000000*	*0.000000*
75%	*3.000000*	*361.000000*	*0.000000*	*0.000000*	*0.000000*
max	*70.000000*	*237340.000000*	*1850.000000*	*307.000000*	*219.000000*

	n_intentos	*n_problemas*	*n_comentarios*	*n_eventos*	*n_sesiones*
Mean	5.810916	1.592604	0.257083	286.315538	9.909931
Std	53.194094	5.088638	2.233291	793.786103	24.111198
Min	0.000000	0.000000	0.000000	1.000000	1.000000
25%	0.000000	0.000000	0.000000	1.000000	1.000000
50%	0.000000	0.000000	0.000000	18.000000	2.000000
75%	0.000000	0.000000	0.000000	106.000000	6.000000
max	1794.000000	23.000000	95.000000	7641.000000	349.000000

Tabla 1. Descripción de los atributos

Contamos con un número total de **3353 usuarios**. Como se puede observar en las tablas anteriores, los valores de los atributos en general están muy dispersos, ya que sus desviaciones típicas son muy grandes. Si observamos los percentiles 25, 50 y 75 se puede observar que en muchos de los atributos el valor es 0. Esto quiere decir que probablemente hay un gran porcentaje de usuarios que tienen muy poca actividad a lo largo del curso. Prueba de esto es que alrededor de un 94% de los usuarios de este curso no ha aprobado y solo el casi 6% sí.

Si nos quedamos únicamente con los datos de quienes sí han aprobado el curso, los valores de los atributos son los que se muestran en las tablas xx y xx. En este caso se puede observar que, de los estudiantes que han aprobado el curso, la media de vídeos vista por los estudiantes es 46.78, siendo la desviación típica 22.23. El estudiante que menos vídeos ha visto no ha accedido a ninguno, y el que más ha visto ha visualizado 70. El 25% de los estudiantes ha visto 31 videos, el 50% ha visto 55 y el 75% ha visto 65.

	n_intentos	*n_problemas*	*n_comentarios*	*n_eventos*	*n_sesiones*
Mean	81.060302	20.356784	2.492462	2650.266332	72.587940
Std	197.570410	3.254886	2.492462	1380.709240	51.576275
Min	10.000000	10.000000	4.164386	107.000000	2.000000
25%	23.000000	19.000000	0.000000	1733.000000	42.500000
50%	32.000000	21.000000	0.000000	2378.000000	60.000000
75%	50.000000	23.000000	1.000000	3303.000000	84.500000
max	1794.000000	23.000000	4.000000	7641.000000	349.000000

	n_videos	*t_videos*	*n_mov_haciadelante*	*n_mov_haciaatras*	*n_aciertos*
Mean	*46.783920*	*24313.834171*	*42.226131*	*24.542714*	*111.100503*
Std	*22.238345*	*30562.615068*	*189.245322*	*39.360378*	*21.176029*
Min	*0.000000*	*0.000000*	*0.000000*	*0.000000*	*60.000000*
25%	*31.500000*	*8666.500000*	*2.000000*	*3.500000*	*100.000000*
50%	*55.000000*	*18756.000000*	*6.000000*	*10.000000*	*111.000000*
75%	*65.000000*	*27163.500000*	*17.000000*	*31.500000*	*120.000000*
max	*70.000000*	*225178.000000*	*1850.000000*	*266.000000*	*219.000000*

Tabla 2. Tabla descriptiva de los valores de los estudiantes aprobados

Se puede observar la ausencia de la variable *aprobado* en las tablas. Este atributo es una variable nominal que en un principio no se puede expresar de forma numérica; así pues, se ha representado con valores binarios, asignándole un 1 para el caso afirmativo y un0 para el negativo. Existe también la posibilidad de extraer como variables el género, la edad, la localización y el lenguaje de los estudiantes, pero el valor para la gran mayoría de ellos es *null*.

Por otro lado, si calculamos las matrices de correlación entre los atributos, podemos observar la correlación que existe con la clase, es decir, con la variable clase *aprobado*.

	n_videos	*t_videos*	*n_mov_haciadelante*	*n_mov_haciaatras*
aprobado	0.716989	0.425424	0.207458	0.351697

Tabla 3. Correlación entre variables y la clase *aprobado*

	n_aciertos	*n_intentos*	*n_problemas*	*n_comentarios*
aprobado	**0.943520**	0.355386	**0.926379**	0.251459

Tabla 4. Correlación entre variables y la clase *aprobado*

	usuario_mac	*usuario_windows*	*usuario_linux*	*n_eventos*	*n_sesiones*
aprobado	-0.013628	0.090195	0.083082	0.748162	0.653066

Tabla 5. Correlación entre variables y la clase aprobado

En las tres tablas anteriores se puede ver que los atributos que tienen mayor correlación con la clase *aprobado* son *n_aciertos y n_problemas*. Esto es algo que se corresponde bastante con la intuición, por otra parte, puesto que para aprobar el estudiante debe contestar bien a los diferentes cuestionarios y ejercicios que existen en el curso. Por otra parte, se puede observar que la variable *n_videos* tiene una alta correlación con la variable *aprobado*. Se puede concluir, por tanto, que la mayoría de los estudiantes que aprueban ven la mayoría de los videos. Eso significa que, en este curso en concreto, los vídeos toman un papel importante en el aprendizaje de los estudiantes y ayuda a que asimilen mejor los conocimientos de cara a aprobar los exámenes. Otras dos variables con una correlación similar son *n_eventos y n_sesiones*, lo que relaciona directamente la actividad que tiene el estudiante en el curso con su nota final en el mismo. En el Anexo I se encuentran las distribuciones de todas las variables.

4.2 Clasificación supervisada

Para este TFM se ha realizado una comparativa de diversos algoritmos de aprendizaje supervisado para compararlos con las técnicas de *Deep Learning*, y así poder comprobar si nos ofrece una ventaja frente al enfoque tradicional.

4.2.1 Näive Bayes

Uno de los algoritmos escogidos para este estudio es *Näive Bayes* (clasificado bayesiano ingenuo).La base de este algoritmo es el teorema de Bayes, que sirve para calcular la

probabilidad de que ocurra un evento A habiendo ocurrido previamente otro evento B. Dicho teorema se expresa de la siguiente forma:

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)}$$

Donde:

- $p(A|B)$ es la probabilidad de que se dé la característica A habiéndose dado la característica B.
- $P(A)$ es la probabilidad a priori de que se produzca la característica A.
- $p(B|A)$ es la probabilidad de que se produzca la característica B habiéndose dado la característica A.

Este algoritmo recibe el apelativo de ingenuo porque asume que todas las características son independientes entre sí, y que la ausencia o presencia de una determinada característica no está relacionada con la ausencia o presencia del resto de características. Por ejemplo, si estamos construyendo un clasificador de animales, el hecho de que el animal a clasificar tenga escamas contribuye a la probabilidad de que sea un pez, independientemente de que también se dé el hecho de que tenga patas. Aunque esto parezca a primera vista una desventaja, supone uno de los puntos fuertes del algoritmo, ya que nos permite analizar cada dimensión del problema por separado en lugar de tener una compleja estructura n-dimensional.

Para este clasificador, se ha asumido que los atributos siguen distribuciones gaussianas, por lo cual la probabilidad de que un atributo x (que sigue una distribución normal v) pertenezca a la clase c es:

$$P(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}$$

Siendo:

- μ_c la media del atributo x asociado a la clase c.
- σ_c^2 la varianza del atributo x asociado a la clase c.

Para cada clase, el clasificador calculará la probabilidad de que los atributos dados pertenezcan a ella, y determinará que pertenecen a la clase que nos proporcione una probabilidad mayor.

4.2.1.1 Resultados

En este apartado se van a mostrar los resultados de la aplicación del algoritmo *Näive Bayes* para la clasificación de estudiantes en aprobados y suspensos. En la siguiente figura podemos observar la matriz de confusión de este algoritmo.

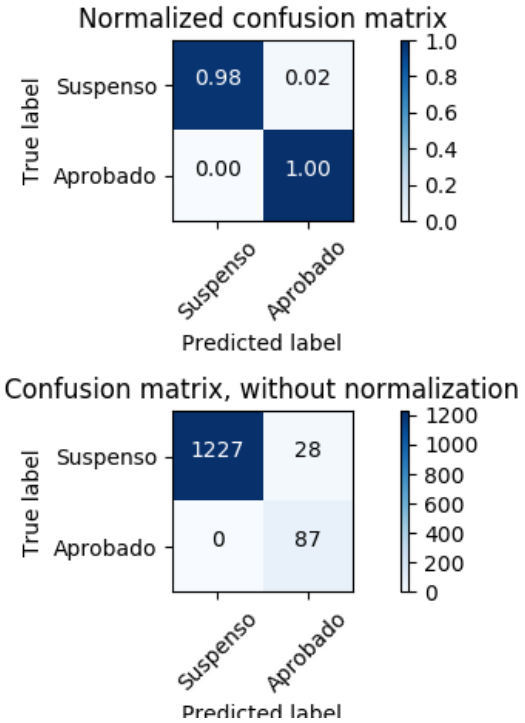

Figura 3. Matriz de confusión de Näive Bayes

Observando la matriz de confusión se puede concluir que clasifica correctamente a casi la totalidad de los suspensos y a la totalidad de los aprobados. Además, calculando la precisión y la exhaustividad de la clase suspenso obtenemos 1 y 0.9777 respectivamente. Esto quiere decir que clasifica todos los aprobados bien y que se nos escapa el 0.0223 de los suspensos, que son clasificados como aprobados. De igual manera calculamos la precisión y la exhaustividad de la clase aprobado y obtenemos 0.7565 y 1 respectivamente. La precisión baja, puesto que se clasifican como aprobados un 0.0223 de los suspensos, como se ha comentado anteriormente.

4.2.2 Árboles de decisión

Los árboles de decisión nos proporcionan un conjunto de reglas con las que podemos determinar a qué clase pertenece un elemento atendiendo a sus atributos. Estos árboles están compuestos de dos tipos de nodos:

- **Nodos de decisión**: Son nodos internos del árbol, y están asociados a uno de los atributos. De cada uno de ellos parten dos o más ramas, que representan los posibles valores que pueden tomar el atributo al que está asociado dicho nodo.
- **Nodos de respuesta**: Son las hojas del árbol, y nos indican a qué clase pertenece el elemento.

Existen diversos algoritmos para construir árboles de decisión, entre los que destacan ID3, C4.5, C5 y CART (*Classification And Regression Trees*). Es este último el escogido para la construcción de nuestro árbol. El algoritmo CART consta de cuatro fases:

- **Construcción**: A partir de un nodo raíz que contiene todos los elementos usados para entrenar, se busca la variable más adecuada para dividir el conjunto en dos nodos hijos. Para decidir qué variable se debe utilizar, se usa una medida de pureza. La variable que obtiene una mayor pureza es la seleccionada para dividir el conjunto en dos. Se debe buscar una función de partición que haga que la pureza de los nodos hijos sea máxima.

  Durante la fase de construcción se le asigna una etiqueta a cada nodo que indica a qué clase pertenece. Este proceso se realiza mediante una función de asignación que tiene en cuenta la probabilidad a priori de pertenencia a cada clase, la pureza de la partición y la proporción final de casos que aparecen en los nodos hojas.
- **Parada del proceso de crecimiento del árbol**: La fase de construcción continúa hasta que se da uno de los siguientes supuestos:
  - Se ha alcanzado la profundidad máxima fijada para el árbol.
  - Todos los nodos hoja tienen la misma probabilidad de pertenecer a una clase u otra (por lo que no se puede determinar un criterio de máxima pureza).
  - Cada nodo hoja sólo tiene un elemento del conjunto de entrenamiento que cumple todas sus condiciones, por lo que no se puede seguir ramificando más.
- **Podado**: En esta fase se simplifica el árbol creado durante la fase de crecimiento. Se utiliza un método de podado en el que se eliminan los nodos que apenas influyen en la precisión del árbol. Este método pondera la precisión del árbol frente a su complejidad y tiempo de procesamiento para determinar qué nodos es mejor eliminar. Los árboles más simples son los que nos dan una mayor capacidad de generalización.

- **Selección:** De todas las posibles podas del árbol, se seleccionará aquella que nos dé el menor error posible en la predicción durante la fase de entrenamiento del árbol y que también minimice el error durante la fase de test.

  Para asegurar el mínimo error a la hora de predecir elementos que no están en el conjunto de entrenamiento, se usa la técnica de *crossvalidation*. Con dicha técnica, se dividirá el conjunto de datos de entrenamiento en N partes. El árbol siempre se entrenará con N-1 partes, dejando la restante para verificar la capacidad de predicción del mismo. Esta operación se repetirá N veces, dejando como test cada vez un conjunto diferente de los N posibles.

Como medida de la pureza, se usan dos métodos diferentes: la impureza de Gini y la ganancia de la información.

- **Impureza de Gini**: Nos permite medir la frecuencia con la que un elemento cualquiera del conjunto sería etiquetado incorrectamente si dicho etiquetado se hiciera de una manera aleatoria.
- **Ganancia de la información**: Nos permite calcular la diferencia en la cantidad de información que se tenía antes de producirse una observación y la cantidad de información que se tiene después de producirse dicha observación. Este método está muy ligado al concepto de entropía.

4.2.2.1 AdaBoost

Normalmente los árboles de decisión se usan conjuntamente con el algoritmo AdaBoost para mejorar los resultados de la predicción. AdaBoost es un meta-algoritmo de aprendizaje que usa combina de forma ponderada la salida de otros algoritmos de aprendizaje. La teoría que está detrás de este algoritmo es que se puede conseguir un clasificador robusto a partir de varios clasificadores débiles.

En este caso, se han usado múltiples árboles de decisión generados a partir de los mismos datos de entrenamiento, pero cada uno con diferencias en sus nodos. A partir de todas estas predicciones, el algoritmo AdaBoost calcula el resultado más probable.

4.2.2.2 Resultados

En este apartado se van a mostrar los resultados de las diferentes versiones del algoritmo basado en árboles de decisión. En la siguiente figura podemos observar las matrices de confusión de este algoritmo.

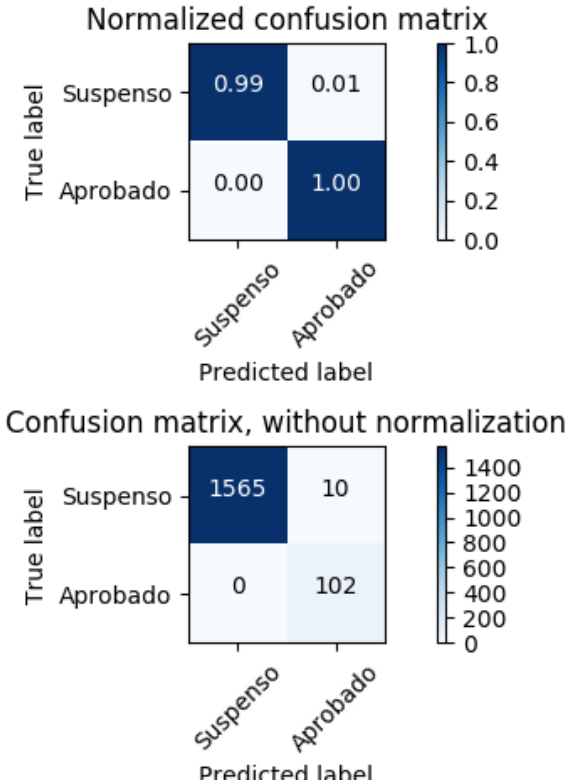

Figura 4. Matriz de confusión de Árbol de decisión (entropía)

Al observar la matriz de confusión, se puede apreciar que clasifica correctamente a casi la totalidad de los suspensos y a la totalidad de los aprobados. Además, calculando la precisión y la exhaustividad de la clase suspenso obtenemos 1 y 0.9936 respectivamente. Esto quiere decir que clasifica todos los aprobados bien y que se nos escapa el 0.0063 de los suspensos, que quedan clasificados como aprobados. De igual manera calculamos la precisión y la exhaustividad de la clase aprobado y obtenemos 0.9107 y 1 respectivamente. La precisión baja, puesto que se clasifican como aprobados un 0.0063 de los suspensos. Sí se observa mejoría respecto al algoritmo de *Näive Bayes*.

4.2.3 Vectores de Máquinas de Soporte (SVM)

Es un algoritmo de clasificación capaz de entrenarse a partir de una serie de elementos para posteriormente predecir a cuál de las posibles clases pertenecen nuevos elementos.

Los elementos que se proporcionan al algoritmo tanto para su entrenamiento como para la predicción vienen definidos por una serie valores que representan mediciones realizadas sobre diferentes características del mismo (por ejemplo, su peso, altura, anchura, color, etc).Al definir cada elemento usando n atributos, realmente estamos representando puntos en un espacio n-dimensional.

El algoritmo SVM es capaz de crear un hiperplano en este espacio n-dimensional que sirve de frontera entre ambas clases, de forma que todos los elementos que queden a un lado del hiperplano pertenecerán a una clase y los que queden al otro lado del hiperplano pertenecerán a la otra clase.

Este hiperplano se calcula de forma que mantenga la máxima distancia posible con los puntos que se encuentran más cerca del mismo. Sin embargo, no siempre es posible la creación de un hiperplano que separe perfectamente en dos clases, ya que estas pueden tener superficie curva, por ejemplo. Por este motivo, se permite cierta flexibilidad a la hora de determinar el hiperplano y permitir algunos errores durante la fase de entrenamiento.

En los casos en los que las clases no se pueden separar un hiperplano, existe una técnica llamada "función kernel" que introduce dimensiones extra al espacio n-dimensional, y proyecta los puntos sobre estas nuevas dimensiones. De esta forma, los puntos que antes no eran separables ahora sí lo son en las dimensiones añadidas.

Matemáticamente, el hiperplano se define como:

$$W^T x_i + b = 0$$

Siendo:

- W el vector ortogonal al hiperplano.
- b el coeficiente de intersección.

Aquellos elementos que den un resultado positivo serán considerados de una clase y los que den un resultado negativo, de la otra. Por tanto, la función de clasificación será:

$$f(x) = sgn(W^T x + b)$$

4.2.3.1 Resultados

En este apartado se van a mostrar los resultados de la utilización del algoritmo SVM con los datos disponibles. En la siguiente figura podemos observar la matriz de confusión.

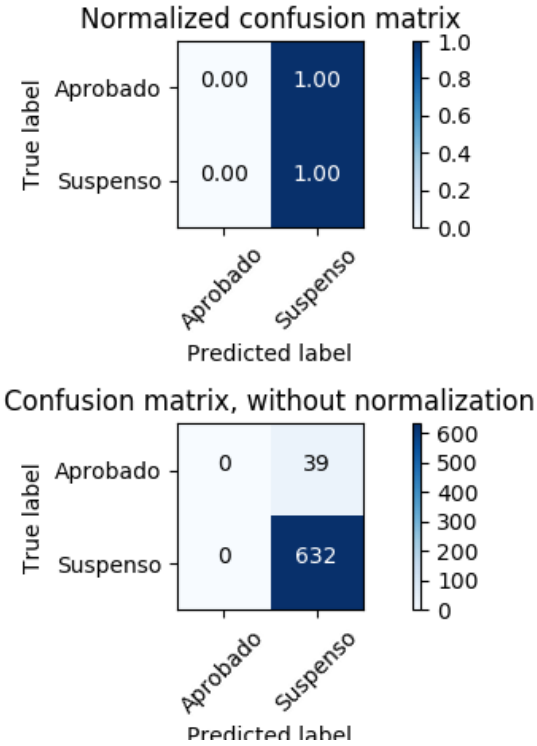

Figura 5. Matriz de confusión de SVM

Al observar la matriz de confusión, lo primero que apreciamos es que este algoritmo no clasifica ningún patrón como aprobado. Esto puede deberse a que un 94.0650% de los patrones que tenemos son de la clase *suspenso*. Además, calculando la precisión y la exhaustividad de la clase suspenso obtenemos 0.9419 y 1 respectivamente. Esto quiere decir que clasifica todos los suspensos bien y que considera directamente los aprobados como suspensos (no clasifica nada como aprobado). En este algoritmo no ha bajado demasiado la precisión, pero esto se debe a que existe un alto porcentaje de suspensos, y no clasifica ningún patrón como aprobado.

4.2.4 Long Short-Term Memory

Como algoritmo de *Deep Learning*, se han escogido las redes neuronales recurrentes (RNN). En este tipo de redes, la predicción se hace no solo en base a los datos de entrenamiento, sino que usa la salida de la propia red neuronal en etapas anteriores como entrada en nuevas fases de entrenamiento.

Como podemos ver en el siguiente diagrama, la red A recibe una colección de datos x con los que genera una predicción h, y además se retroalimenta con su propia información.

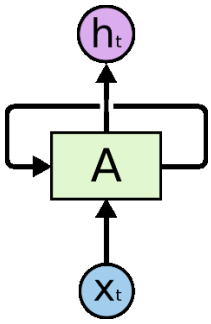

Figura 6. Retroalimentación de la *Recurrent Neural Network*

Si nos adentramos en el bucle de entrenamiento, el diagrama quedaría como se muestra a continuación:

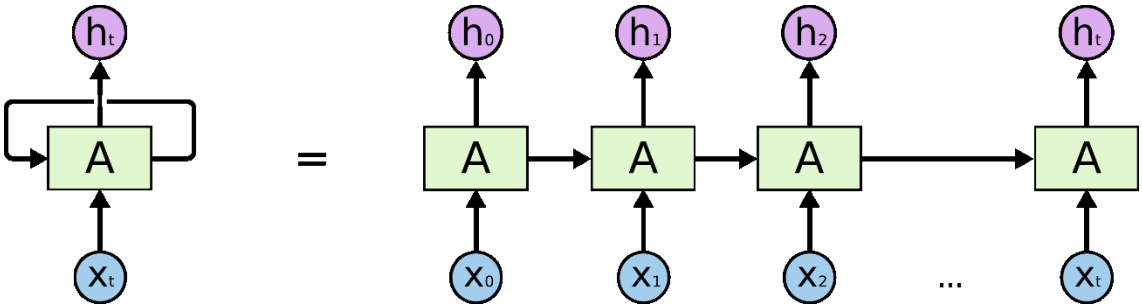

Figura 7. Bucle de entrenamiento de la Recurrent Neural Network

Este tipo de redes permiten tener una especie de "memoria", donde los resultados previos siguen presentes durante el entrenamiento.

Entre los múltiples algoritmos que pertenecen a la familia de las redes neuronales recurrentes, se ha optado por el algoritmo LSTM (Long Short Term Memory) [26]. Este algoritmo se caracteriza por ser capaz de establecer dependencias entre elementos a pesar de que dichos elementos estén separados por varias fases de entrenamiento (dependencias a largo plazo).

4.2.4.1 Resultados

En este apartado se van a mostrar los resultados obtenidos mediante el algoritmo SVM. En la siguiente figura podemos observar la matriz de confusión.

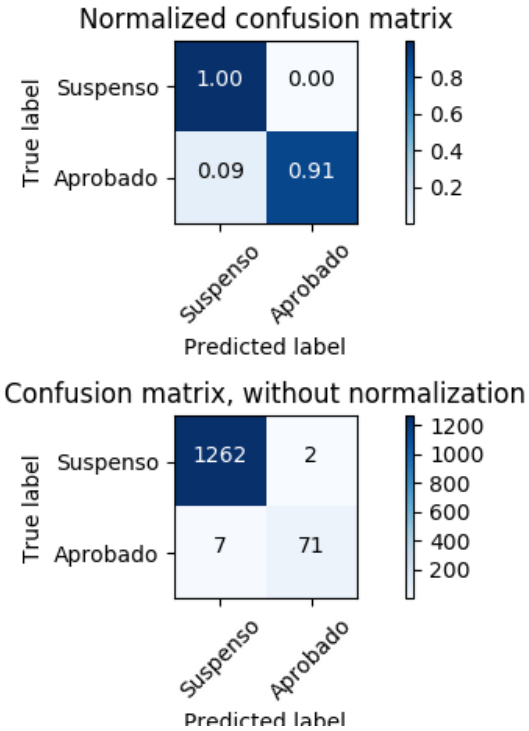

Figura 8. Matriz de confusión de LSTM

Al observar la matriz de confusión, vemos que la mayoría de los suspensos los clasifica como suspensos, pero sigue clasificando mal unos pocos. De igual forma observamos que la mayoría de los aprobados están bien clasificados, pero sigue clasificando mal algunos. Además, calculando la precisión y la exhaustividad de la clase suspenso obtenemos 0.9949 y 0.9984 respectivamente. Si comparamos estos resultados con los obtenidos mediante el árbol de decisión, vemos que este algoritmo no aporta un resultado mejor a los anteriores.

4.2.5 Comparación de los resultados

A continuación, se va a proceder a comparar la tasa de acierto de los cuatro algoritmos para poder determinar cuál es mejor.

	Naive Bayes	*Decision Tree Entropy*	*Decision Tree Gini*	*SVM*	*LSTM*
Tasa de acierto	97.47 %	99.40 %	99.64 %	93.59 %	94.41 %

Tabla 6. Comparativa de Tasas de Aciertos

En la tabla se puede observar la tasa de aciertos de cada algoritmo de aprendizaje. Se puede apreciar que el algoritmo con mayor precisión es el **árbol de decisión.** Uno de los objetivos de este TFM era comprobar si los algoritmos de *Deep Learning* aportaban alguna mejora a los análisis del ámbito de *Learning Analytics*. Según los resultados finales, se puede concluir que en este caso el algoritmo *Long Short-Term Memory*, algoritmo de *Deep Learning*, **no aporta una mejora clara.** Además, este tipo de algoritmos no permite ver las decisiones que se toman (no se conoce la explicación de por qué se clasifica de una manera u otra). Esto nos conduce a pensar que los algoritmos de Deep Learning no aportan mejoría en este ámbito que nos ocupa.

Sin embargo, es necesario tener en cuenta que los algoritmos de *Deep Learning* pueden dar lugar a mejores resultados cuando la cantidad de variables es muy grande. Quedaría abierta, por tanto, la posibilidad de tratar de extraer más variables de los *logs* y experimentar con un mayor número de variables que las utilizadas en este trabajo, para comprobar si, en ese caso, este tipo de algoritmos podrían conducir a la obtención de mejores resultados.

Finalmente, como el objetivo principal de este TFM es predecir situaciones de riesgo y también proporcionar *feedback* concreto que permita realizar un seguimiento a los estudiantes. Los algoritmos de *Deep Learning* no son capaces de ofrecer este tipo de información. No obstante, los árboles de decisión sí permiten obtener información sobre por qué se predice que un estudiante va a aprobar o suspender. Por esta razón, la predicción basada en un análisis de los datos por semanas que se va a presentar a continuación se ha realizado con árboles de decisión.

4.3 Análisis por semanas

Como se ha comentado anteriormente, de cara a ofrecer retroalimentación semanalmente, se ha llevado a cabo la extracción de 42 modelos. Cada modelo corresponde a la progresión que se ha tenido en cada semana, haciendo un total de 42 semanas. Tras entrenar los datos con un árbol de decisión, usando el criterio de selección de entropía, se ha obtenido la siguiente representación de los árboles. Debido al tamaño de estos, en esta sección se mostrarán parcialmente para comentar los resultados y en el anexo II se mostrarán en tamaño completo. Además, se comentará el resultado de las 10 primeras semanas para observar la evolución en los criterios de decisión. En la representación de los árboles se hace referencia

a las variables utilizadas para entrenar con formato X_i. Así pues, para un mejor seguimiento de los resultados, se muestra la siguiente tabla con el índice de cada variable.

0	n_videos	**5**	**n_intentos**
1	t_videos	**6**	n_problemas
2	n_mov_haciadelante	**7**	n_comentarios
3	n_mov_haciaatras	**8**	n_eventos
4	n_aciertos	**9**	n_sesiones

Esto es, X_6 corresponde al número de problemas realizados. Una vez mostrado el índice de cada variable se procede a explicar los resultados obtenidos. La figura 10 muestra el árbol generado para la primera semana del curso.

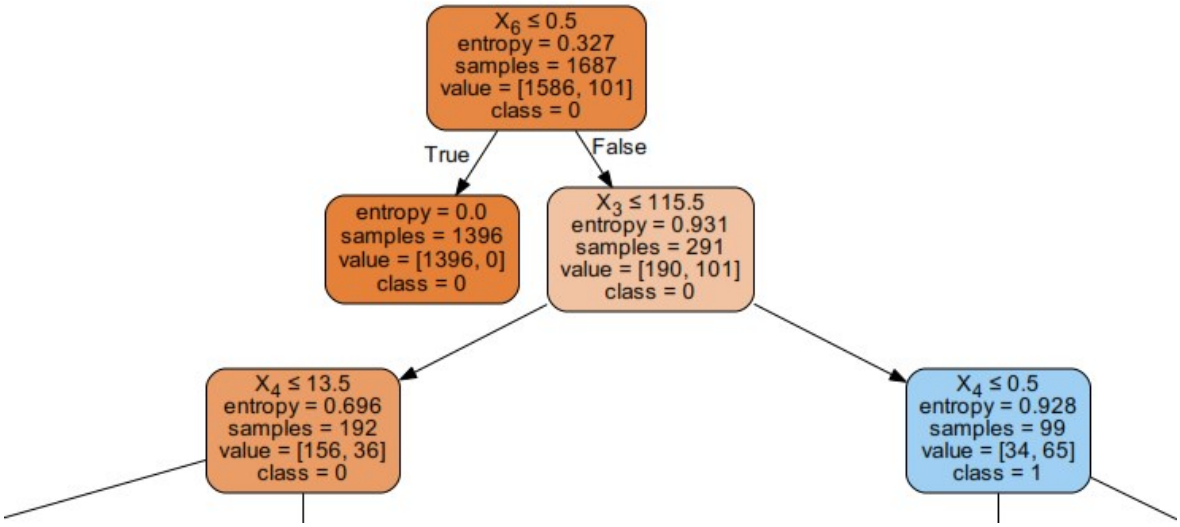

Figura 9. Árbol de decisión de la semana 1

La primera decisión para clasificar, en este caso, se toma en función del número de problemas realizados. Tal y como se puede observar en el primer nodo del árbol, se considera que, si en la primera semana el estudiante no ha realizado ni un solo problema, lo más probable es que suspenda. Si ha realizado algún ejercicio, entonces el siguiente parámetro influyente es el número de movimientos hacia atrás realizados (X_3), lo cual podría denotar interés en enterarse bien de los contenidos en esta primera semana de clase. En cualquier caso, durante esta primera semana de clase, se podría intervenir para motivar a los estudiantes a que se impliquen en la realización de ejercicios (para lo cual necesitarán estudiar primero). Más adelante, en las siguientes semanas, se verá que la variable *n_aciertos*

(X_4) empieza a ser más determinante, y se podrá observar el progreso del estudiante esperado para lograr aprobar.

En la semana 3 (ver figura 11) ya observamos que el criterio principal para clasificar es el número de aciertos del estudiante. Se determina que si el estudiante no ha resuelto ejercicios correctamente un mínimo de 15 veces en esas tres semanas de curso lo más probable es que suspenda. Además, considera que si, aun teniendo más de 15 aciertos, tiene menos de 56, puede seguir en riesgo de suspender y empieza a fijarse en el número de veces que va hacia atrás durante la visualización de los vídeos. Esta última variable puede denotar el interés en el estudiante por no perderse nada de los vídeos mostrados hasta el momento.

Por tanto, en este momento también es posible ofrecer intervenir, ofreciendo *feedback* (al profesor o al estudiante) para advertir del progreso del estudiante e incluso aconsejarle o motivarle para que siga avanzando y no se estanque.

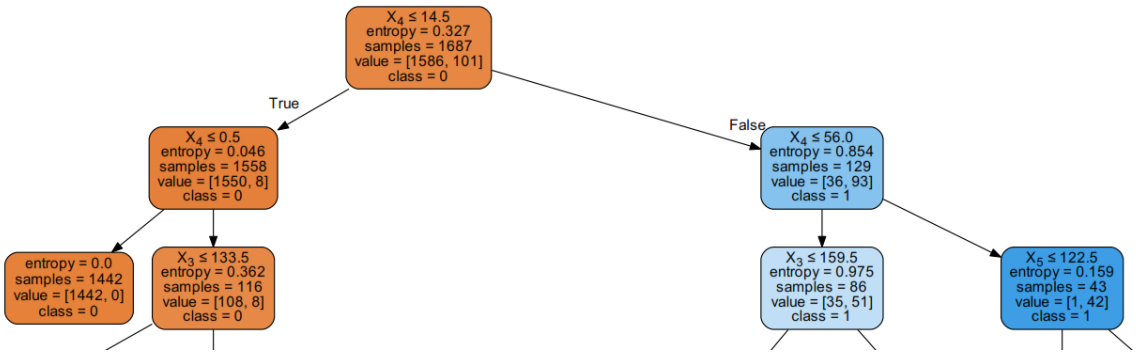

Figura 10. Árbol de decisión de la semana 3

En el árbol de decisión de la semana 5 (figura 12) se puede observar que las variables relevantes para la clasificación no varían apenas, pero vemos que el valor asociado al criterio de clasificación sí varía. La cantidad de aciertos que debe tener ahora en total es mayor que en las semanas anteriores (mayor que 34). Esto demuestra que, al menos en las primeras semanas del curso, parece existir una progresión en la primera variable determinante. Y de la misma forma que en la semana 3, sigue advirtiendo en la segunda condición de que si no tiene más de 56 aciertos puede seguir con riesgo de suspender.

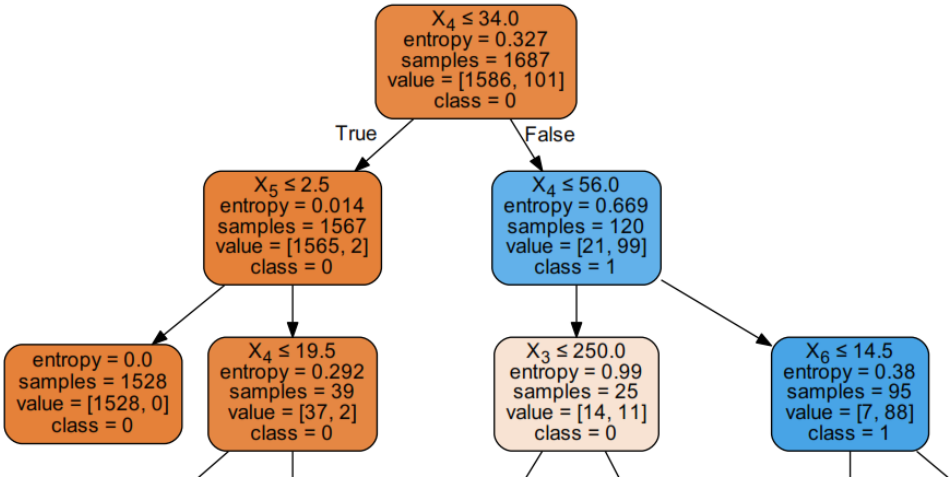

Figura 11. Árbol de decisión de la semana 5

En la figura 13, correspondiente a la semana 8, se puede observar que el número de aciertos necesario para que no se prediga riesgo de suspenso ha aumentado considerablemente: es necesario tener un mínimo de 51 aciertos para tener posibilidades de aprobar. En este caso y para los datos analizados, este sería el valor límite bajo el cual sería recomendable realizar una intervención.

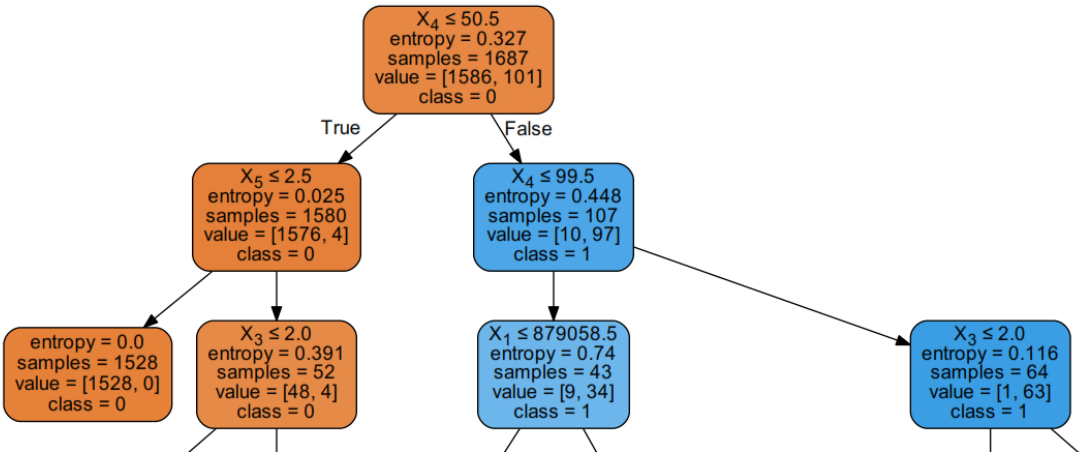

Figura 12. Árbol de decisión de la semana 8

Llega un momento, mientras los modelos van evolucionando para contemplar semanas subsecuentes, en el que el árbol se estabiliza y no muestra casi cambio en las primeras ramas, como se puede ver en la semana 21 (figura 14). Esto se debe a que los nuevos datos no aportan información significativa para que el árbol cambie de atributos determinantes, y se

siguen considerando los mismos para clasificar en una clase o en otra. Sí se puede apreciar en la rama de la clase 0 (*suspender*), una pequeña variación en la clasificación de unos pocos estudiantes suspensos, (ahora 1526 son clasificados como suspensos, mientras que en el árbol anterior eran 1528, indicando esto que para 2 estudiantes se continúan mirando otros atributos antes de clasificarles definitivamente).

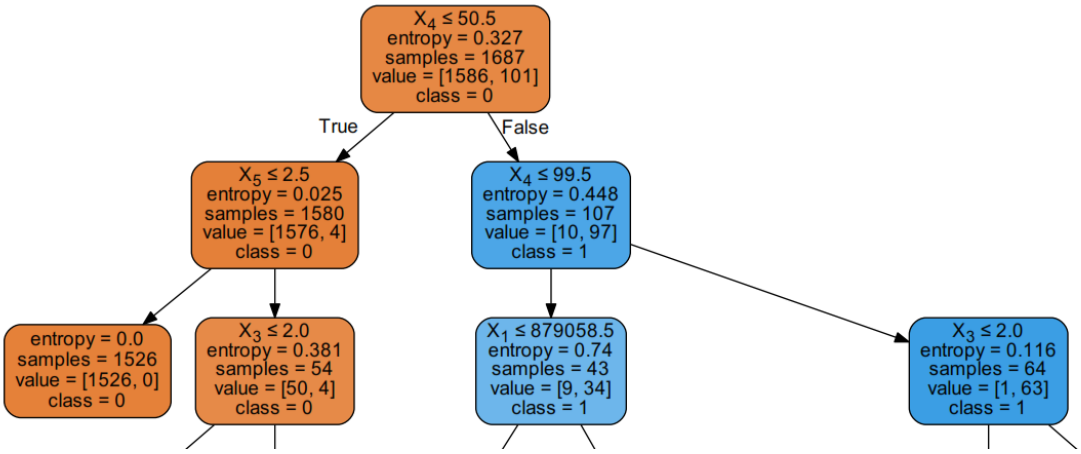

Figura 13. Árbol de decisión de la semana 21

Como se ha comentado antes, en el anexo II se muestran los árboles completos y se puede ver de forma más detallada cómo va variando la forma que ajusta las predicciones conforme van avanzando las semanas.

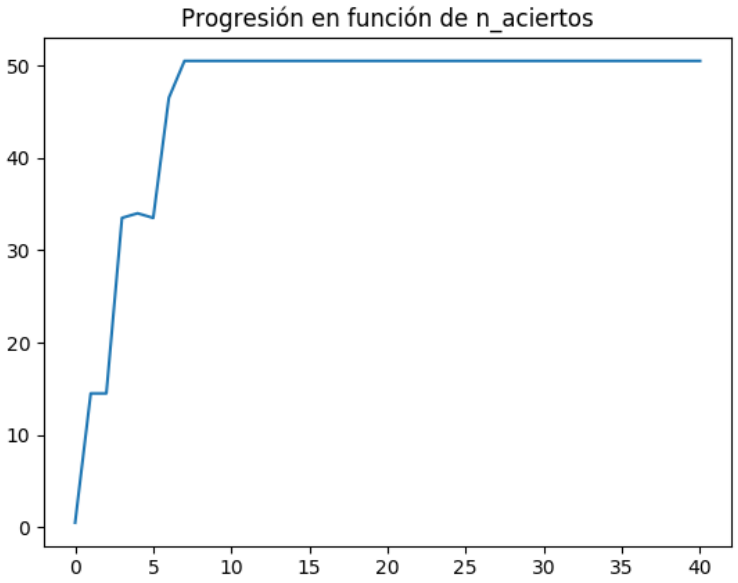

Figura 14. Criterio de clasificación basado en el número de aciertos

En la figura anterior se observa el valor mínimo del número de aciertos que un estudiante debe tener para no ser clasificado como suspenso a lo largo de las distintas semanas del curso (la progresión que se ha comentado anteriormente). Se puede observar que ese valor se estabiliza en una semana determinada. Esto se debe a que los datos de los estudiantes no aportan información significativa para cambiar de criterio, por lo que este criterio se mantiene estable a partir de ese momento.

Esto es solo un ejemplo de la información que se puede extraer de los árboles de decisión sobre el progreso esperado de los estudiantes en las distintas semanas del curso. Gracias a este análisis, se le puede proporcionar información relevante al profesor o al propio estudiante, motivándole o incluso avisándole del riesgo que corre, para intentar que remonte y termine el curso de forma exitosa.

5 Conclusiones y trabajo futuro

5.1 Conclusiones

En este trabajo se ha explorado la utilización de técnicas de aprendizaje automático y *deep learning* para el análisis de los datos sobre las interacciones de los estudiantes de un curso MOOC, con el objetivo de ser capaces de predecir situaciones de riesgo como el suspenso o el abandono. Tras el trabajo realizado, se pueden extraer dos conclusiones principalmente. Por un lado, se ha observado que el aprendizaje con un algoritmo de *Deep Learning* avanzado (*Long Short-Term Memory*) no aporta una mejora clara en comparación con el resto de algoritmos convencionales que se han utilizado. Por tanto, la utilización de este tipo de algoritmos no parece justificar el hecho de que, en caso de utilizarlos, se desconocerían los motivos por los que los estudiantes son clasificados de un modo u otro, dado que funcionan como una "caja negra".

Sin embargo, se quiere dejar una puerta abierta a la posibilidad de extraer un mayor número de variables de los eventos almacenados en los *logs* de los cursos. Los algoritmos de *Deep Learning* han demostrado su buen funcionamiento con grandes números de variables, por lo que quizá en ese caso los resultados podrían mejorar.

Por otro lado, se ha realizado un análisis de los datos por semanas, generando 42 modelos que permiten predecir el riesgo de suspenso semana a semana, utilizando árboles de decisión. La generación de estos árboles ha permitido observar cómo se va generando una especie de "progresión ideal" que debería tener el estudiante en el curso para aprobar. Los resultados mostrados en este TFM son solo un ejemplo concreto aplicado a un MOOC. Sería interesante aplicar este análisis a otros MOOCs de la misma plataforma y ver sus resultados.

Durante el proceso de extracción de las variables hemos observado que la estructura que ahora mismo existe para registrar los eventos en los *logs* no favorece el análisis de los mismos. Sería ideal que esta estructura estuviera diseñada para facilitar la extracción de información relevante en el ámbito de la analítica del aprendizaje, ya que mediante esta analítica se puede no solo ayudar a los estudiantes, sino ofrecer sugerencias de mejora a los profesores, etc.

Se ha observado que este MOOC en concreto apenas tiene restricciones. Esto significa que el estudiante puede realizar el curso y aprobar sin necesidad de completar todo el temario ni

estudiar todo el material disponible. Esta situación, por una parte flexible para los estudiantes, dificulta su seguimiento ya que algunos de ellos no acceden a los contenidos de forma ordenada. Por ejemplo, se observó que un estudiante entró una vez al curso, realizó todos los ejercicios (sin visualizar vídeos) y aprobó en esa misma sesión (y no era el profesor). Datos como estos pueden distorsionar los resultados del análisis del aprendizaje. Por otro lado, justamente la variedad de perfiles y de formas de interactuar con los cursos constituyen un reto para la investigación en este ámbito.

Finalmente, se ha observado en la información disponible sobre los estudiantes que hay bastantes de personas que no tienen la necesidad de obtener el certificado ni, por lo tanto, de aprobar el curso. Observando los comentarios en los foros, muchas de estas personas tienen interés en aprender, pero no en examinarse. Actualmente la forma de medir si han adquirido el conocimiento es comprobando si lo han aprobado o no, esto es, comprobando cuál es su nota final. De cara a analizar el aprendizaje de las personas en un sentido más amplio habría que investigar o inventar nuevas formas de comprobarlo. De momento, en este trabajo nos centramos en predecir y tratar de evitar el riesgo de suspenso, aunque sería interesante estudiar otros aspectos pedagógicos y sociales en el contexto del aprendizaje a través de cursos MOOC.

5.2 Trabajo Futuro

Como trabajo futuro, se han planteado varias cosas ya durante esta memoria. Lo primero y en relación con el objetivo principal de este TFM, sería explorar la utilización de algoritmos de *Deep Learning* con una cantidad mayor de variables para ver si existe la posibilidad de que, en este ámbito, se mejoren los resultados obtenidos mediante técnicas más tradicionales.

En segundo lugar, sería interesante aplicar el análisis por semanas que se ha realizado en este MOOC a otros MOOCs de la plataforma edX. De esta forma, se podría analizar la viabilidad y efectividad de la aplicación de este método en otros cursos y con otros estudiantes, analizando también si es posible proporcionar retroalimentación al profesor o al estudiante y cómo se podría llevar a cabo.

Por otro lado, existe una interacción social en los foros de estos cursos que podría aportar mucha información, no solo sobre la relación entre los estudiantes o incluso entre profesores y estudiantes, sino también sobre los temas sobre los que intercambian mensajes. Esto permitiría detectar qué temas o conceptos generan más dudas y podría incluso explicar unos

malos resultados en ejercicios relacionados con estos temas. Los contenidos de los comentarios de los estudiantes en los foros pueden constituir una fuente de información que ayuden a explicar el comportamiento y los resultados de los estudiantes..

Un componente importante de los MOOCs y que tiene bastante relación con la nota final de un estudiante, al menos en el MOOC analizado en este TFM, son los vídeos. Es necesario prestar especial atención a este componente porque puede influir en el aprendizaje. Un posible análisis futuro sería el comparar los resultados de MOOCs con videos de poca duración y vídeos de larga duración. Es posible que esta característica también influya en el aprendizaje.

Los MOOCs que se ofertan en esta plataforma tienen temáticas y ámbitos muy variados. Otro trabajo que sería interesante realizar es la comparación de los análisis entre MOOCs de diferentes ámbitos como por ejemplo entre ciencias puras y ciencias sociales. Esta comparación puede sacar a la luz diferencias en el comportamiento y aprendizaje en diferentes ámbitos de la educación, si lo hubiera.

Finalmente, una última reflexión: son muchos los tipos de análisis que se pueden realizar a partir de la información sobre las interacciones de los estudiantes con los MOOCs en el ámbito de *Learning Analytics*, y este tipo de analíticas del aprendizaje pueden suponer un gran avance en el contexto de la enseñanza-aprendizaje a distancia.

6 Bibliografía y referencias

[1] C. J. Bonk, M. M. Lee, T. C. Reeves, y T. H. Reynolds (2015), *MOOCs and Open Education Around the World*. Routledge,.

[2] R. S. Baker y P. S. Inventado, «Educational Data Mining and Learning Analytics», en *Learning Analytics*, Springer, New York, NY, 2014, pp. 61-75.

[3] Y. LeCun, Y. Bengio, y G. Hinton (2015). «Deep learning», Nature, vol. 521, n.o 7553, pp. 436-444.

[4] Deng, L., & Yu, D. (2014). Deep learning: methods and applications. Foundations and Trends® in Signal Processing, 7(3–4), 197-387.

[5] «Course | Quijote501x | edX». [En línea]. Disponible en: https://courses.edx.org/courses/course-v1:UAMx+Quijote501x+3T2017/course/. [Accedido: 11-sep-2017].

[6] «edX», *edX*. [En línea]. Disponible en: https://www.edx.org/. [Accedido: 11-sep-2017].

[7] Charles LANG, George SIEMENS, Alyssa WISE, and Dragan GAŠEVIĆ. (2017). Handbook of Learning Analytics – First edition. : Society for Learning Analytics Research. pp 1-356

[8] Ferguson, R., & Shum, S. B. (2012, April). Social learning analytics: five approaches. In Proceedings of the 2nd international conference on learning analytics and knowledge. pp. 23-33. ACM

[9] Bannert, M., Reimann, P., & Sonnenberg, C. (2014). Process mining techniques for analysing patterns and strategies in students' self-regulated learning. Metacognition and learning, 9(2), 161-185.

[10] Daems, O., Erkens, M., Malzahn, N., & Hoppe, H. U. (2014). Using content analysis and domain ontologies to check learners' understanding of science concepts. journal of computers in education, 1(2-3), 113-131..

[11] Kloos, C. D., Alario-Hoyos, C., Fernández-Panadero, C., Estévez-Ayres, I., Muñoz-Merino, P. J., Cobos, R., ... & Chicaiza, J. (2016, September). eMadrid project: MOOCs and learning analytics. In Computers in Education (SIIE), 2016 International Symposium on. pp. 1-5. IEEE..

[12] «Miríada X», *Miríada X*. [En línea]. Disponible en: https://miriadax.net/home. [Accedido: 11-sep-2017].

[13] Srilekshmi, M., Sindhumol, S., Chatterjee, S., & Bijlani, K. (2016, December). Learning Analytics to Identify Students At-risk in MOOCs. In Technology for Education (T4E), 2016 IEEE Eighth International Conference on. pp. 194-199. IEEE.

[14] L. P. V. BRAGA, L. I. O. VALENCIA, y S. S. R. CARVAJAL, *Introducción a la Minería de Datos*. Editora E-papers.

[15] «INTEF - educaLAB». [En línea]. Disponible en: http://educalab.es/intef. [Accedido: 11-sep-2017].

[16] «Udacity - Free Online Classes & Nanodegrees». [En línea]. Disponible en: /. [Accedido: 11-sep-2017].

[17] «Cursos online: aprende de todo y a tu propio ritmo | Udemy». [En línea]. Disponible en: https://www.udemy.com/. [Accedido: 11-sep-2017].

[18] «Coursera | Online Courses From Top Universities. Join for Free», *Coursera*. [En línea]. Disponible en: https://es.coursera.org/. [Accedido: 11-sep-2017].

[19] *Pattern Recognition and Machine Learning | Christopher Bishop | Springer*. .

[20] «Artificial Neuronal Network». [En línea]. Disponible en: http://www.iitbhu.ac.in/faculty/min/rajesh-rai/NMEICT-Slope/lecture/c14/l1.html. [Accedido: 11-sep-2017].

[21] «Welcome to Python.org», *Python.org*. [En línea]. Disponible en: https://www.python.org/. [Accedido: 11-sep-2017].

[22] «TensorFlow», *TensorFlow*. [En línea]. Disponible en: https://www.tensorflow.org/. [Accedido: 11-sep-2017].

[23] «scikit-learn: machine learning in Python — scikit-learn 0.19.0 documentation». [En línea]. Disponible en: http://scikit-learn.org/stable/. [Accedido: 11-sep-2017].

[24] «Sublime Text: The text editor you'll fall in love with». [En línea]. Disponible en: https://www.sublimetext.com/. [Accedido: 11-sep-2017].

[25] «PyCharm: Python IDE for Professional Developers by JetBrains», *JetBrains*. [En línea]. Disponible en: https://www.jetbrains.com/pycharm/. [Accedido: 11-sep-2017].

[26] «Microsoft Word 2016, Software de procesamiento de texto y documentos». [En línea]. Disponible en: https://products.office.com/es-es/word. [Accedido: 11-sep-2017].

[27] «Zotero | Home». [En línea]. Disponible en: https://www.zotero.org/. [Accedido: 11-sep-2017].

[28] «Understanding LSTM Networks -- colah's blog». [En línea]. Disponible en: http://colah.github.io/posts/2015-08-Understanding-LSTMs/. [Accedido: 11-sep-2017].

Anexo I. Distribuciones de las variables

A continuación, se muestran las distribuciones de las distintas variables utilizadas en este TFM.

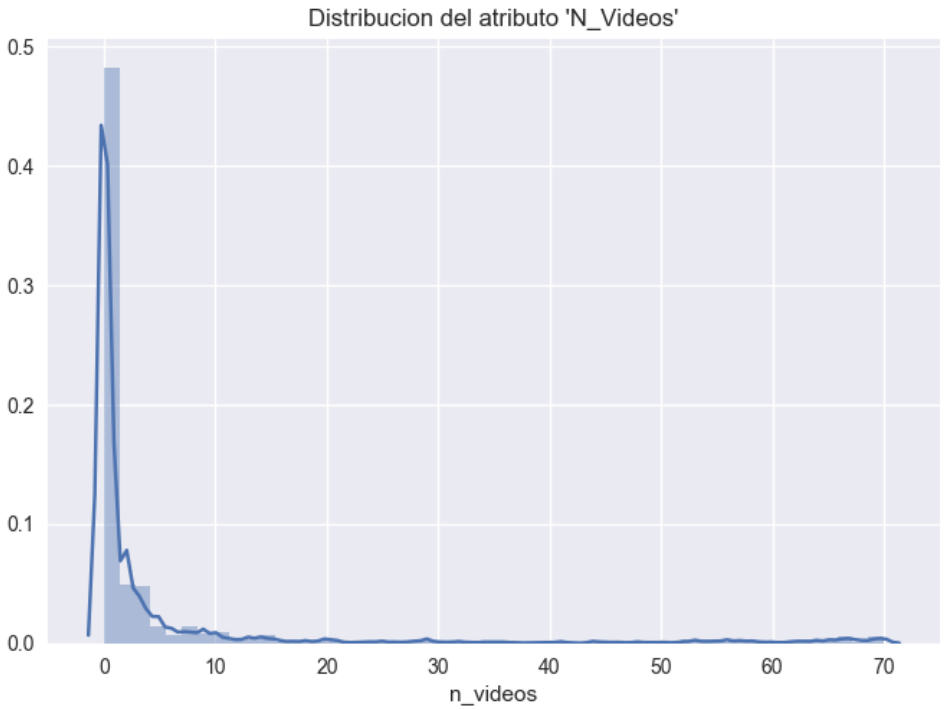

Figura 15. Distribución de la variable n_videos

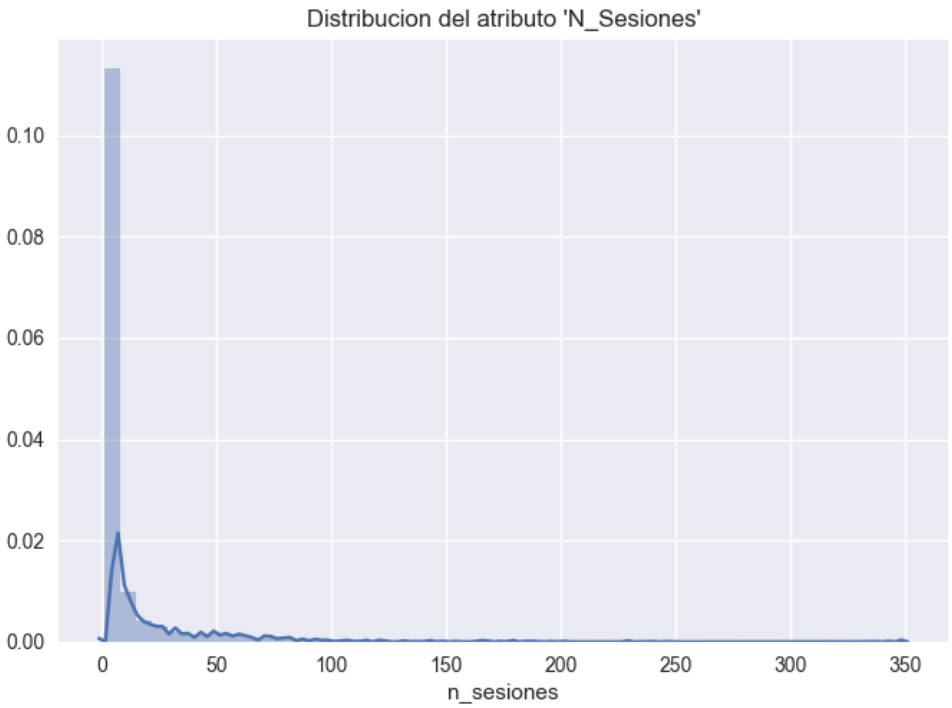

Figura 16. Distribución de la variable n_sesiones

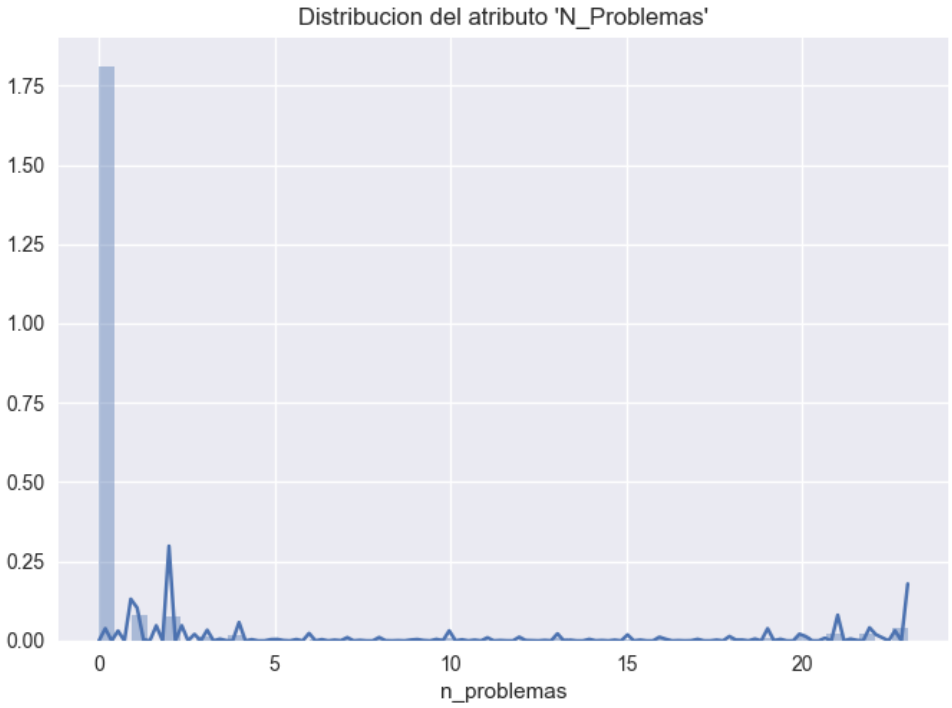

Figura 17. Distribución de la variable n_problemas

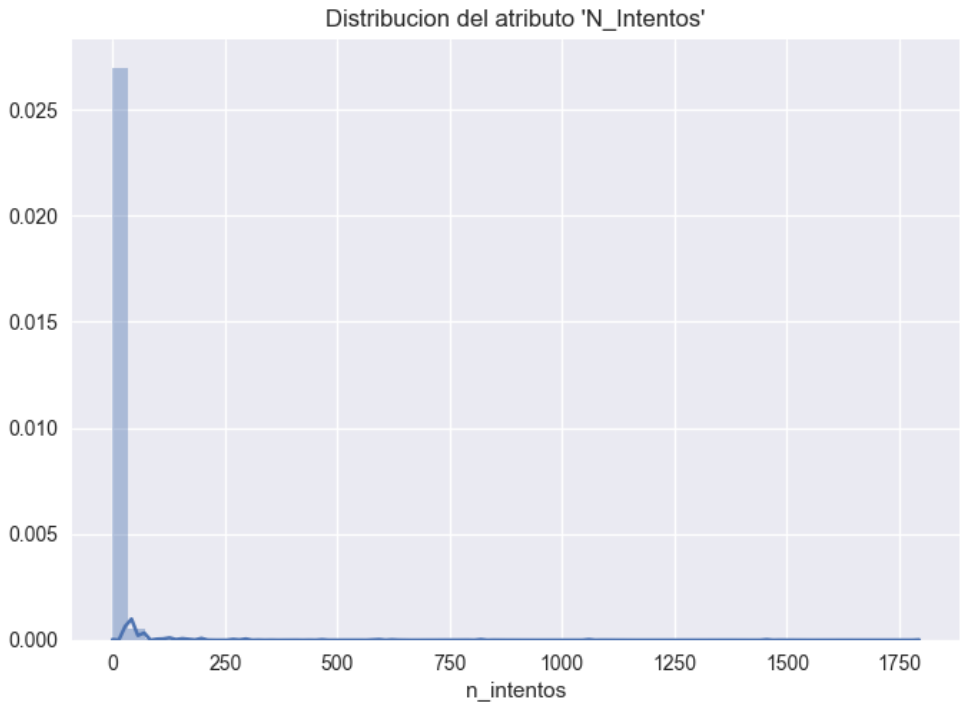

Figura 18. Distribución de la variable n_intentos

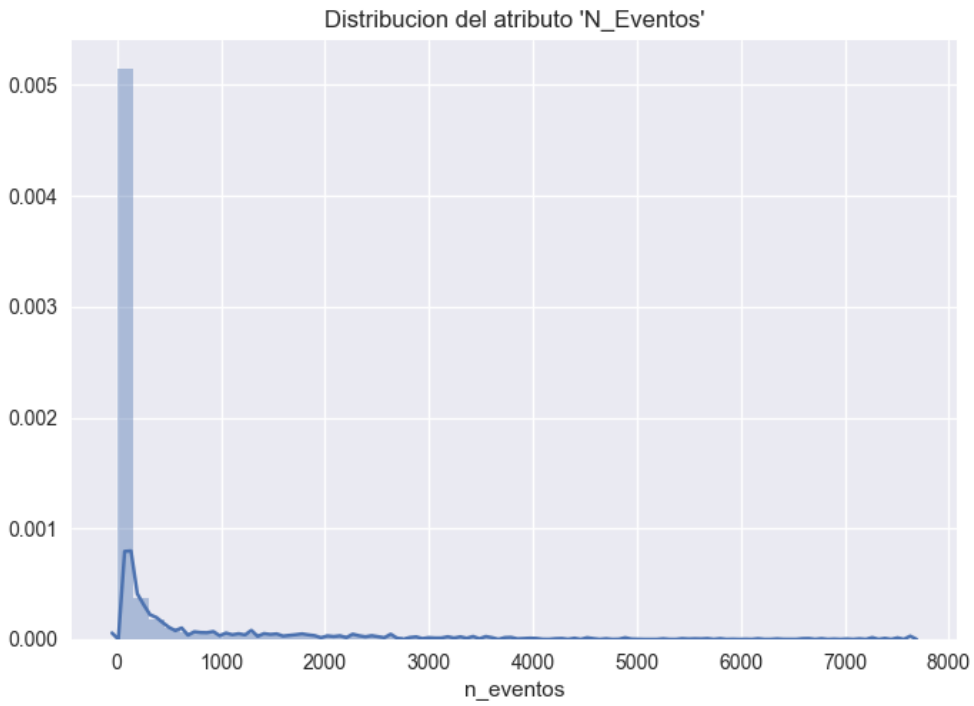

Figura 19. Distribución de la variable n_eventos

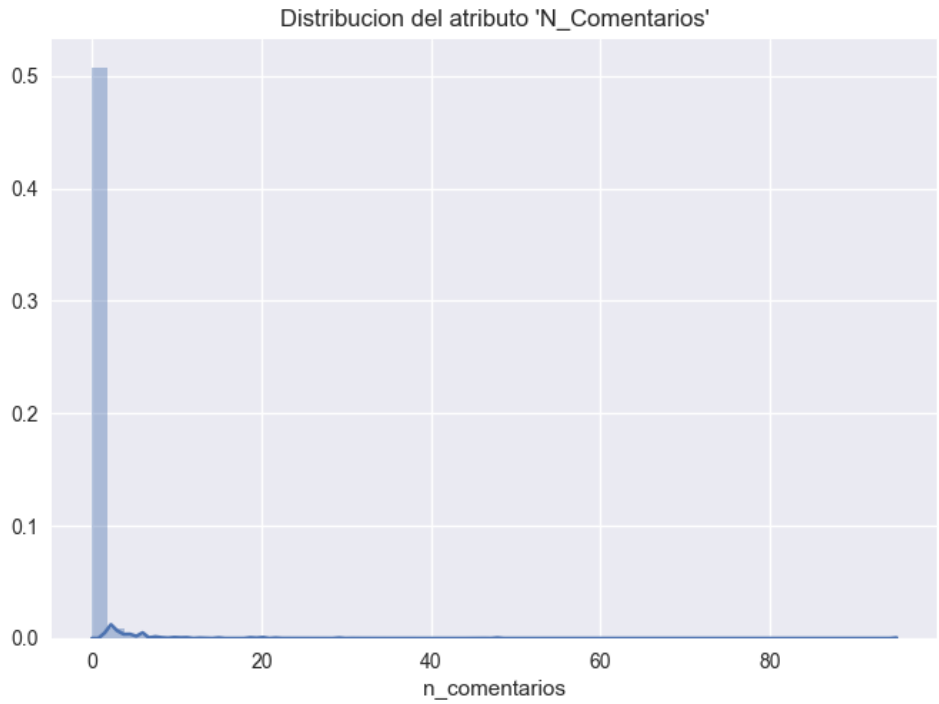

Figura 20. Distribución de la variable n_comentarios

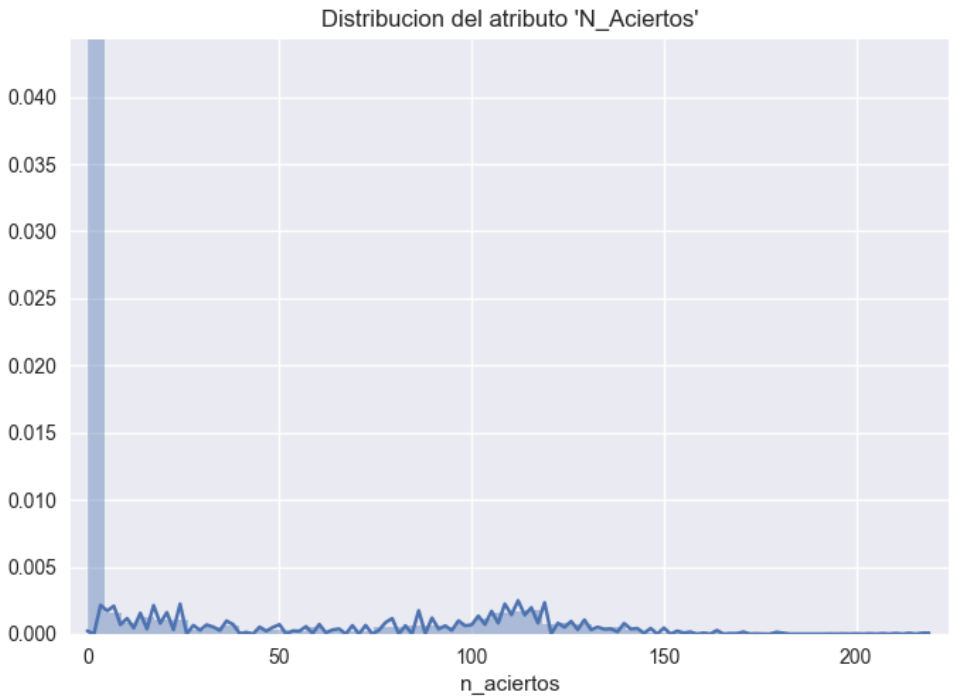

Figura 21. Distribución de la variable n_aciertos

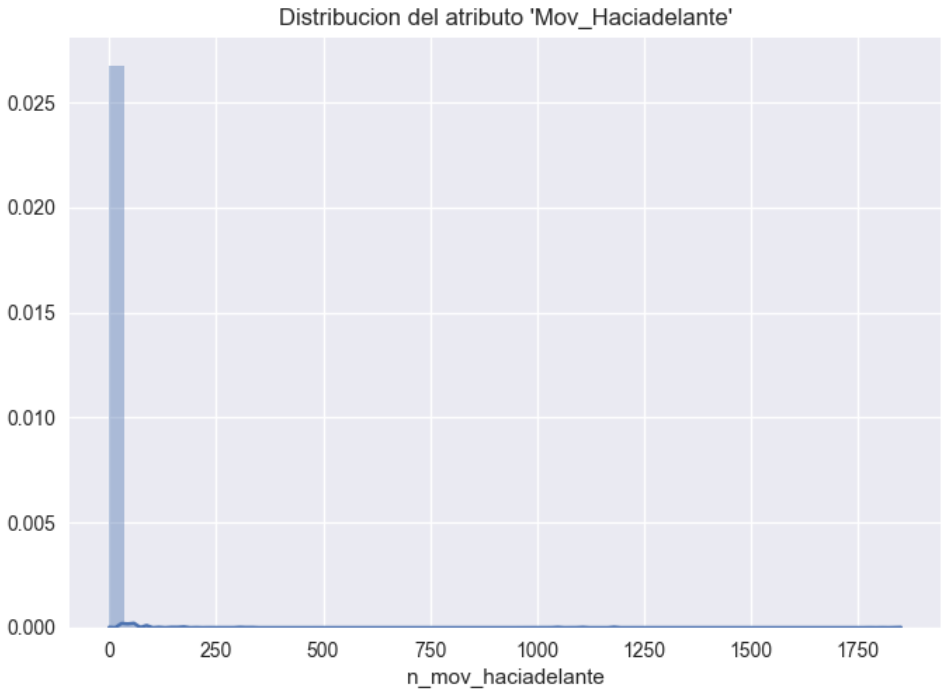

Figura 22. Distribución de la variable n_mov_haciadelante

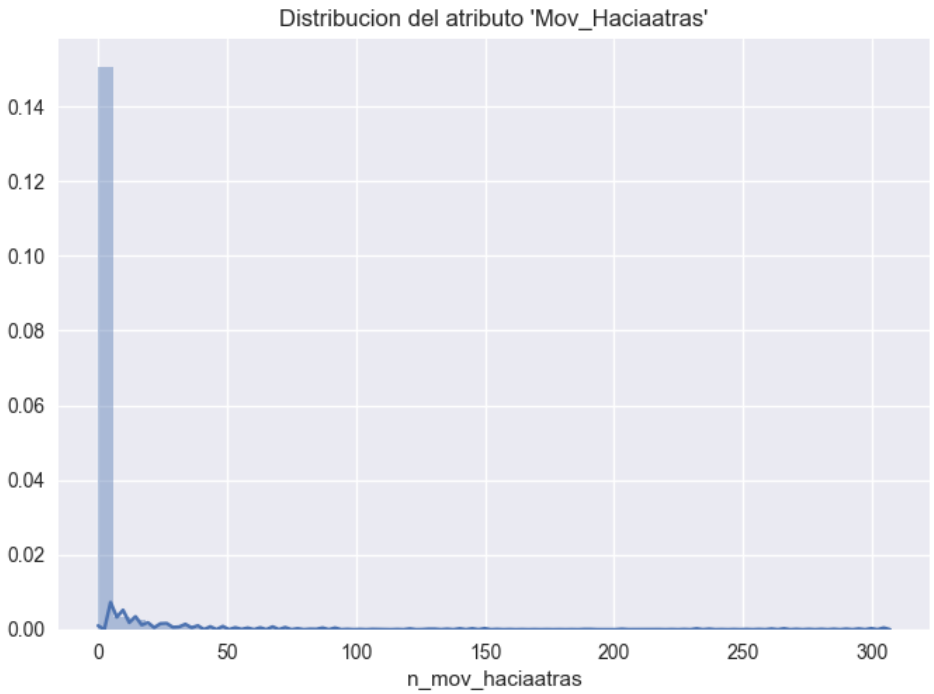

Figura 23. Distribución de la variable n_mov_haciaatras

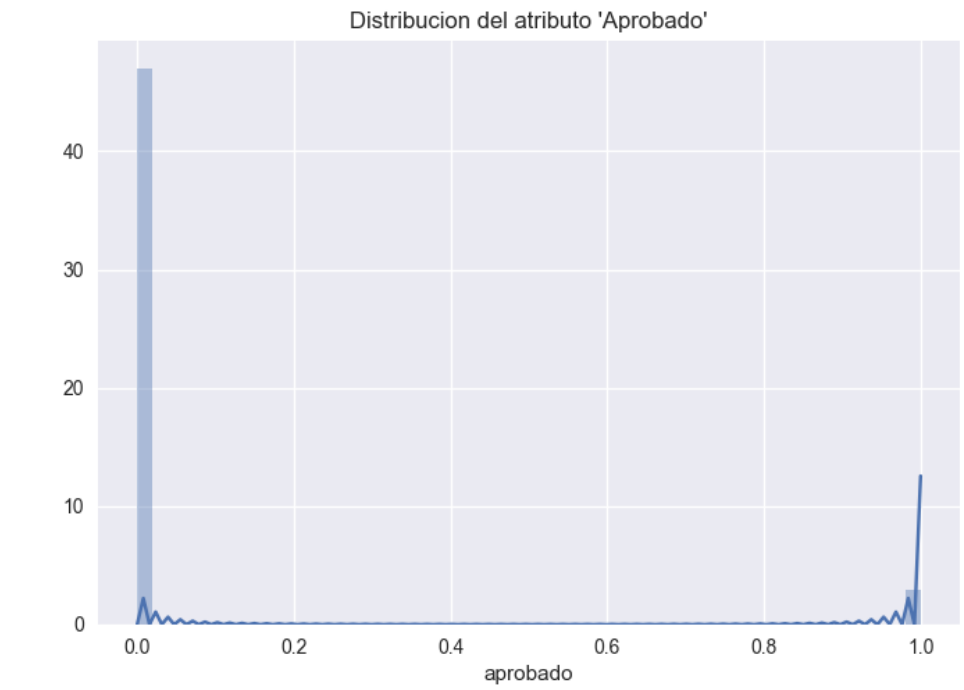

Figura 24. Distribución de la variable aprobado

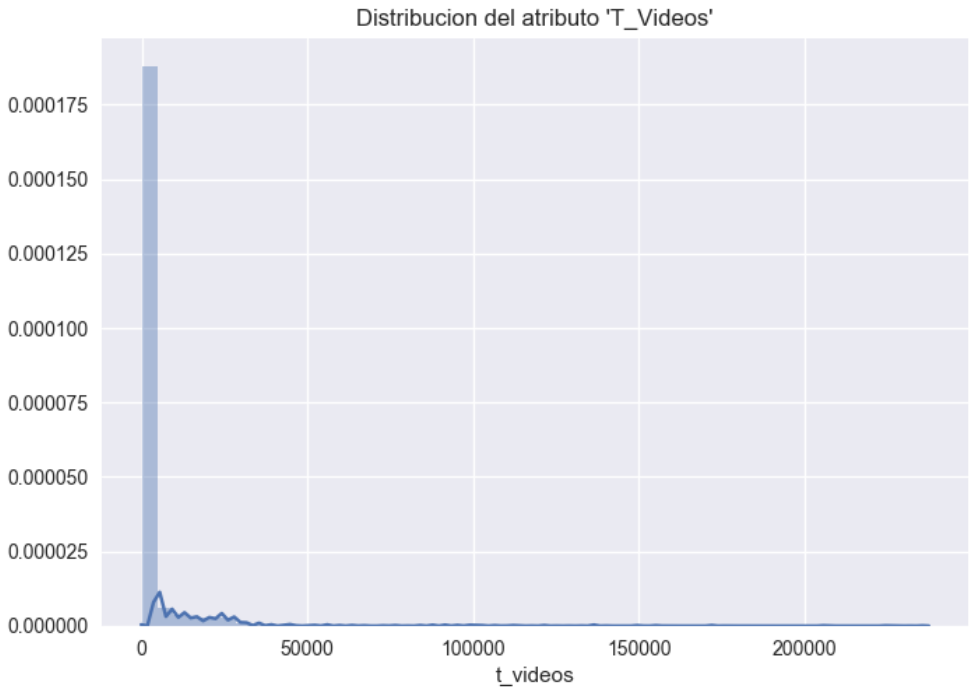

Figura 25. Distribución de la variable t_videos

Anexo II. Árboles de decisión

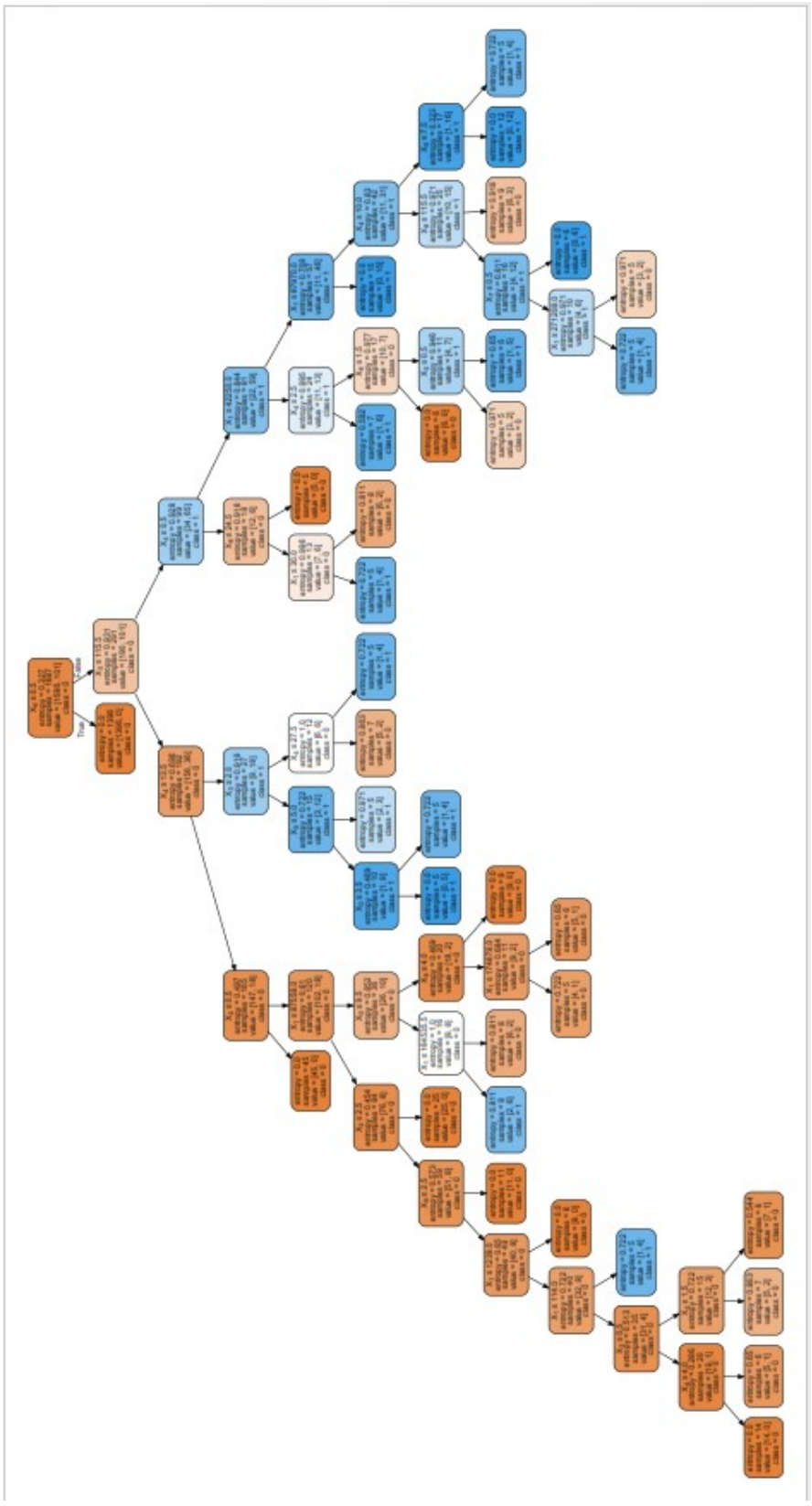

Figura 26. Árbol de decisión semana 1

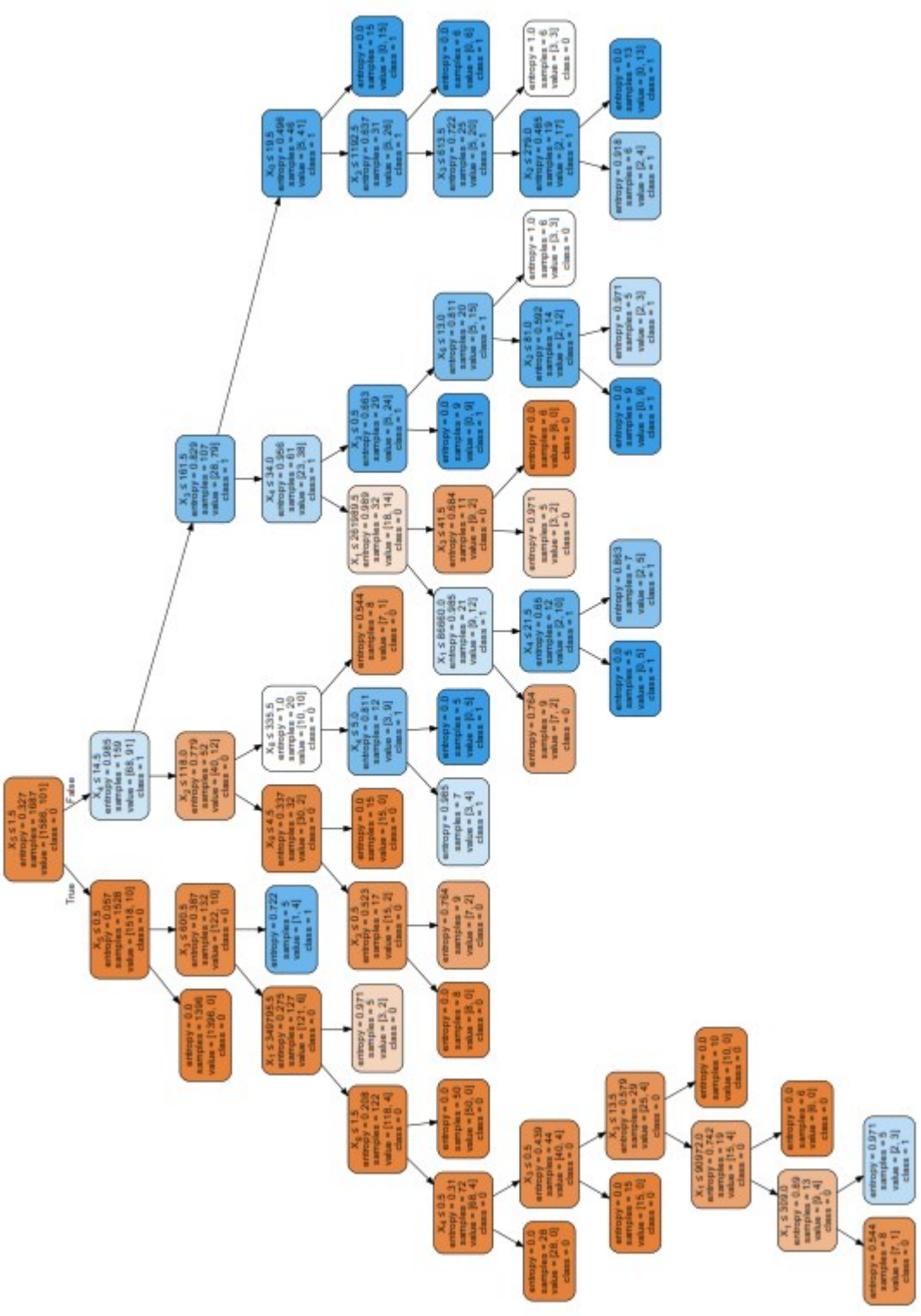

Figura 27. Árbol de decisión de la semana 2

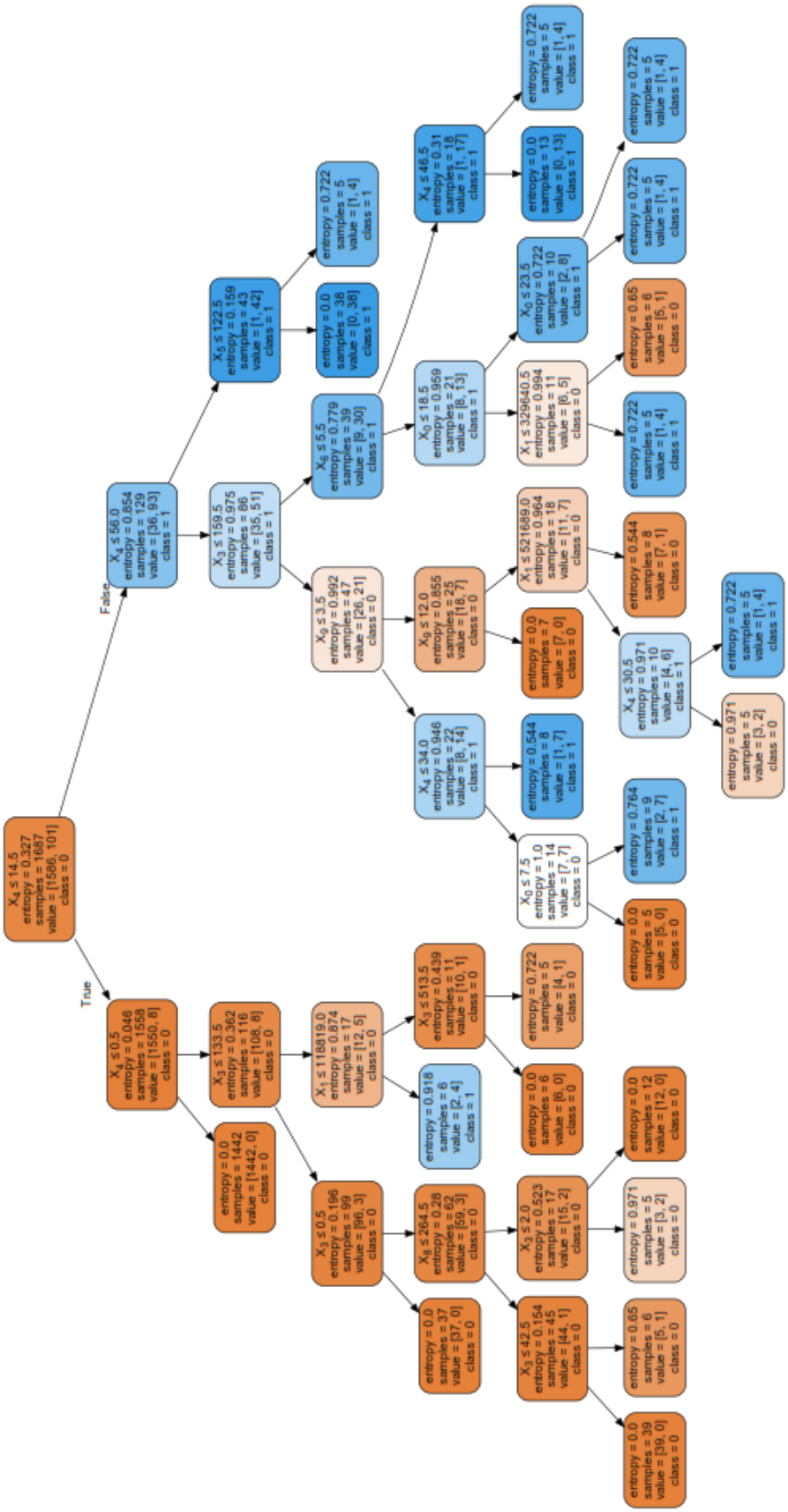

Figura 28. Árbol de decisión de la semana 3

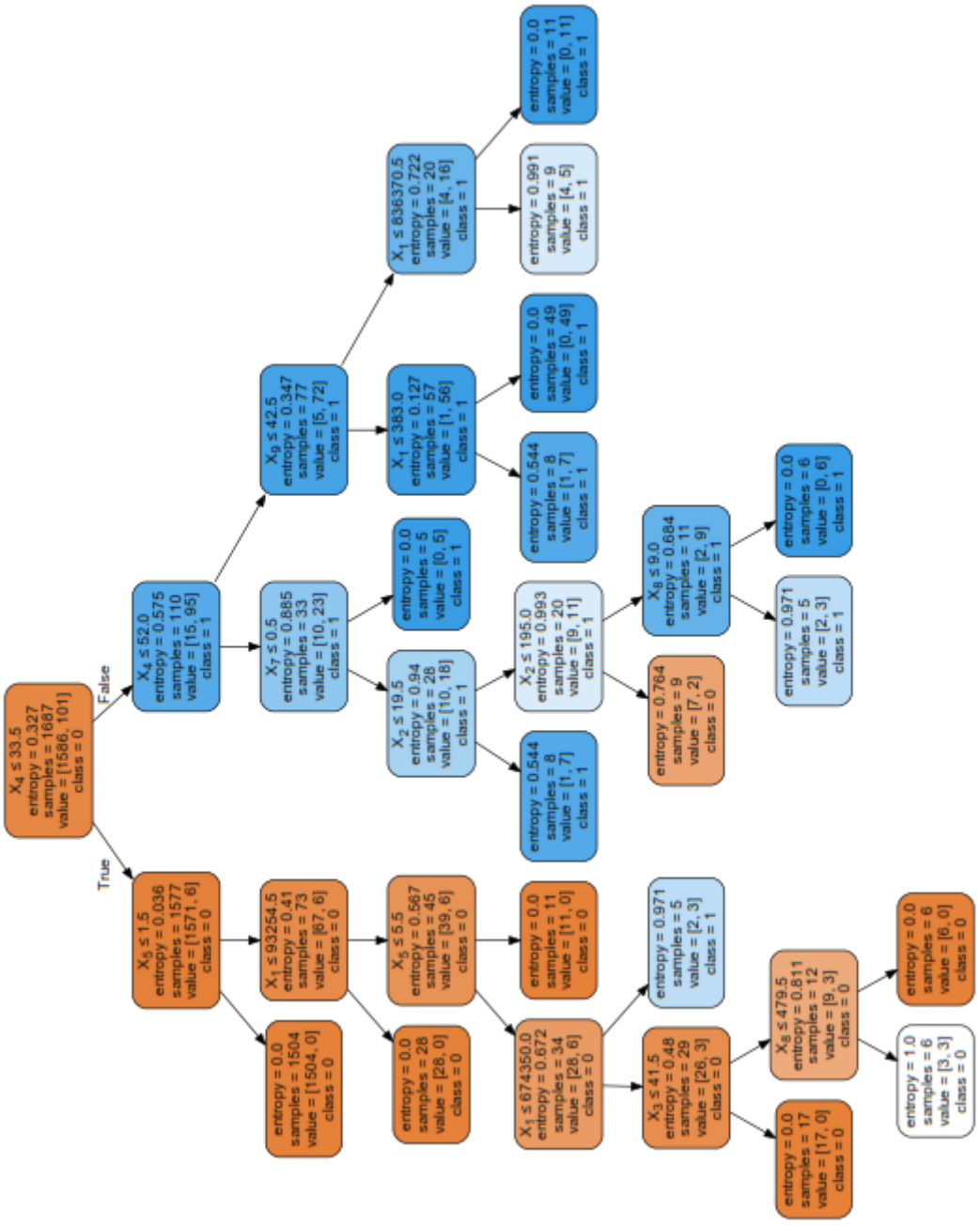

Figura 29. Árbol de decisión de la semana 4

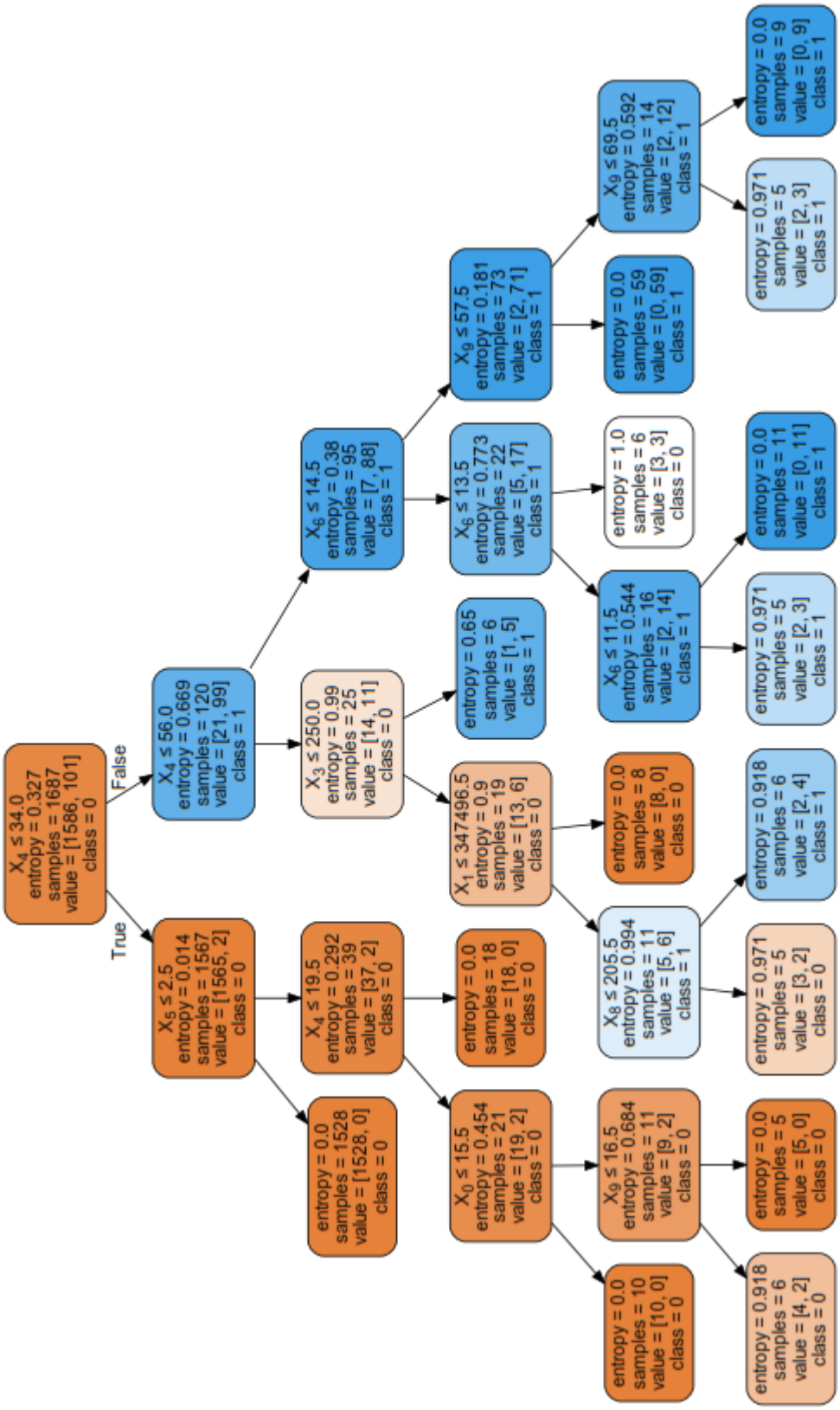

Figura 30. Árbol de decisión semana 5

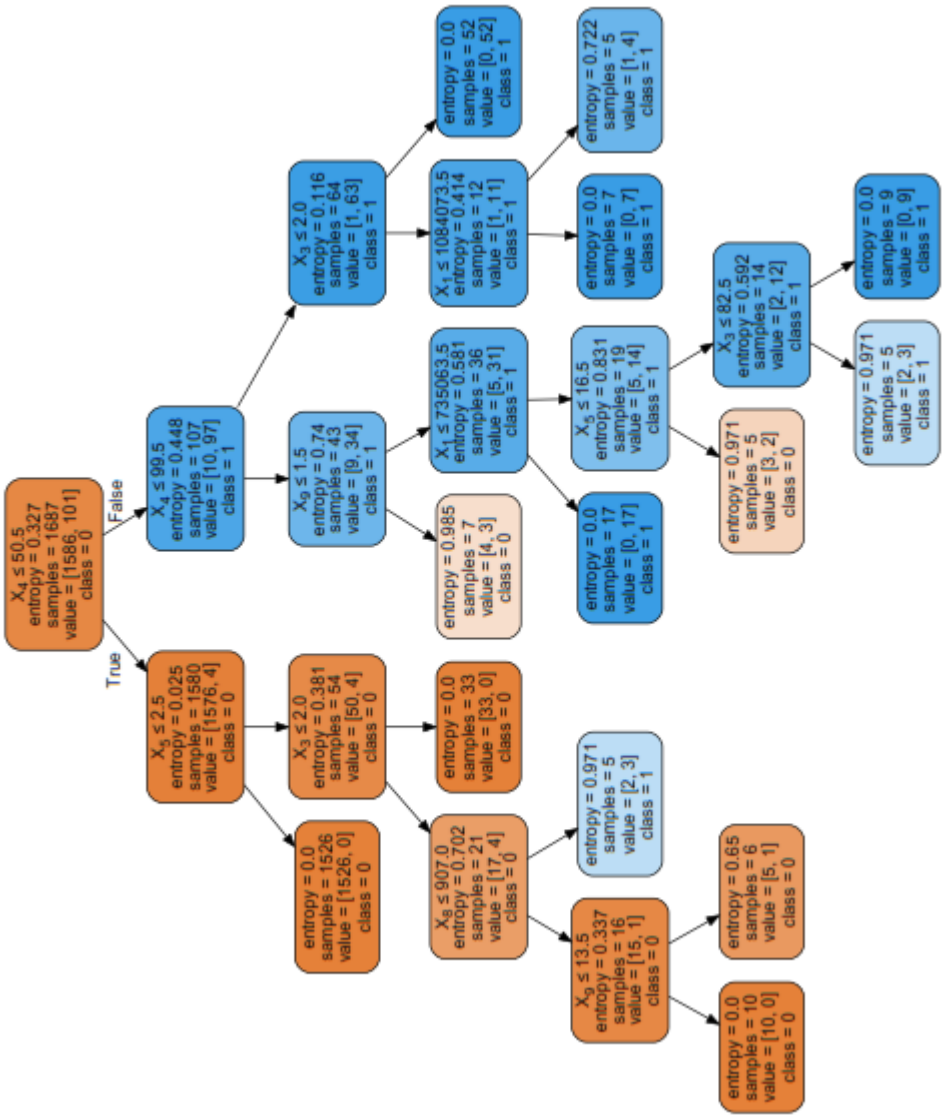

Figura 31. Árbol de decisión semana 10

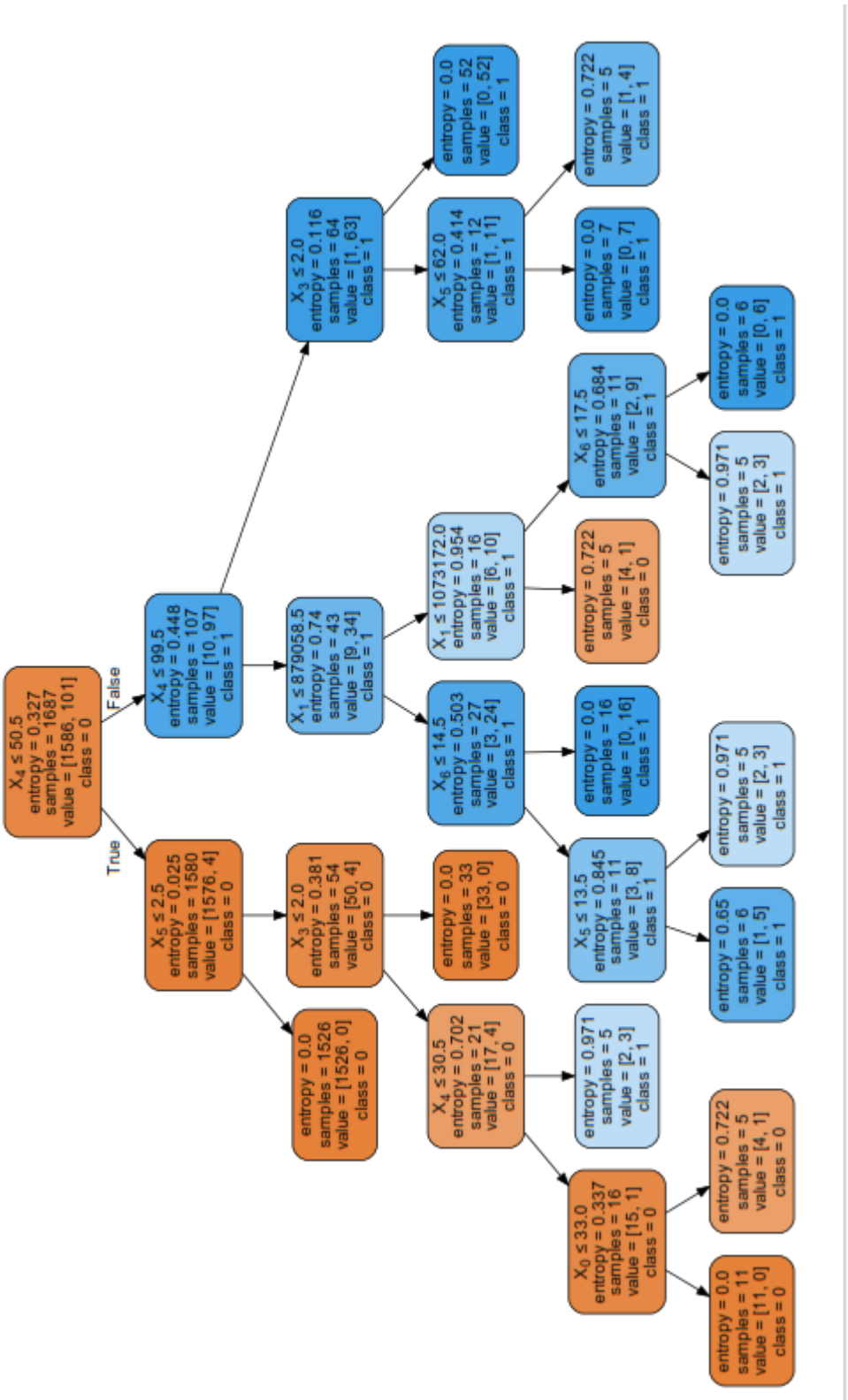

Figura 32. Árbol de decisión semana 15

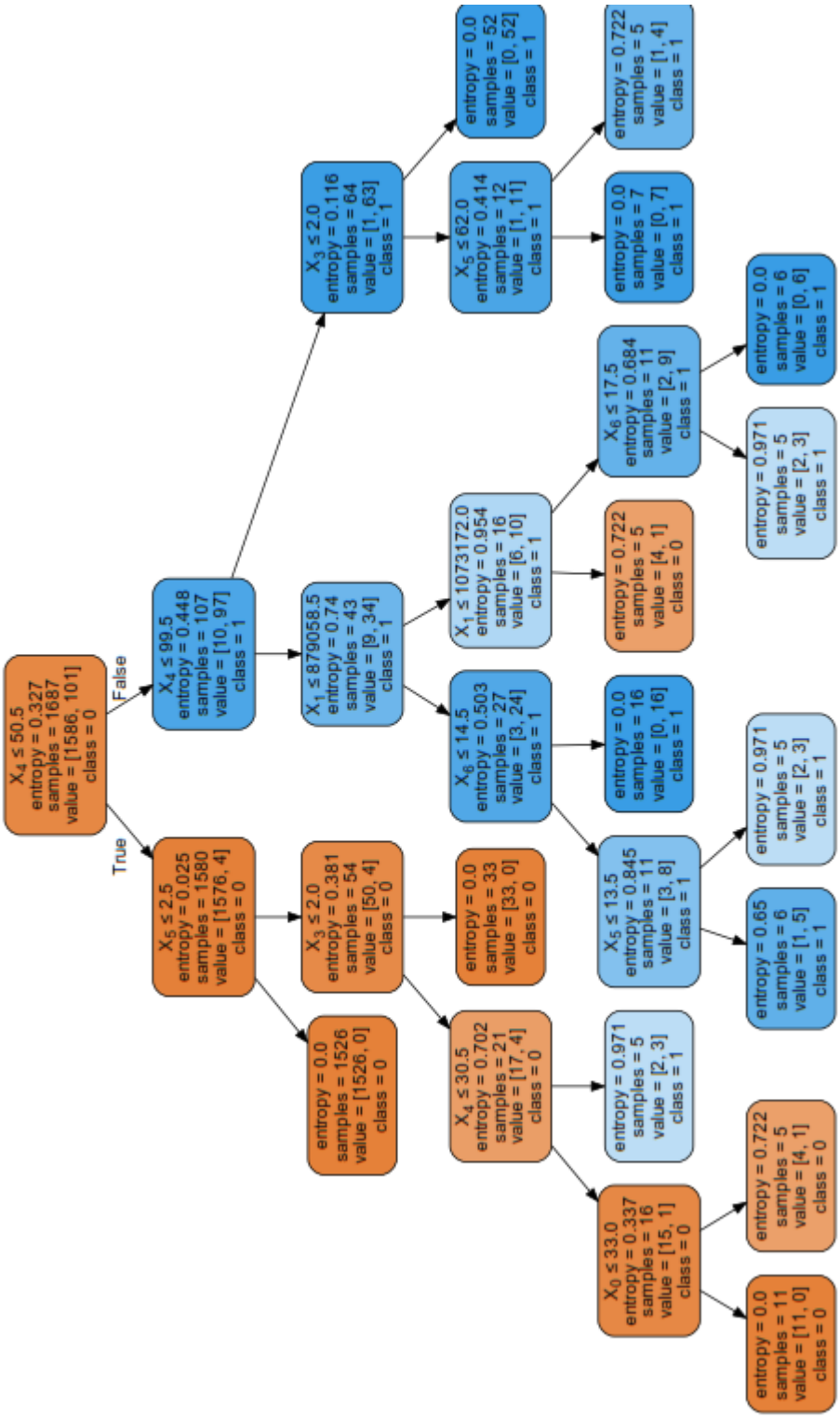

Figura 33. Árbol de decisión semana 20

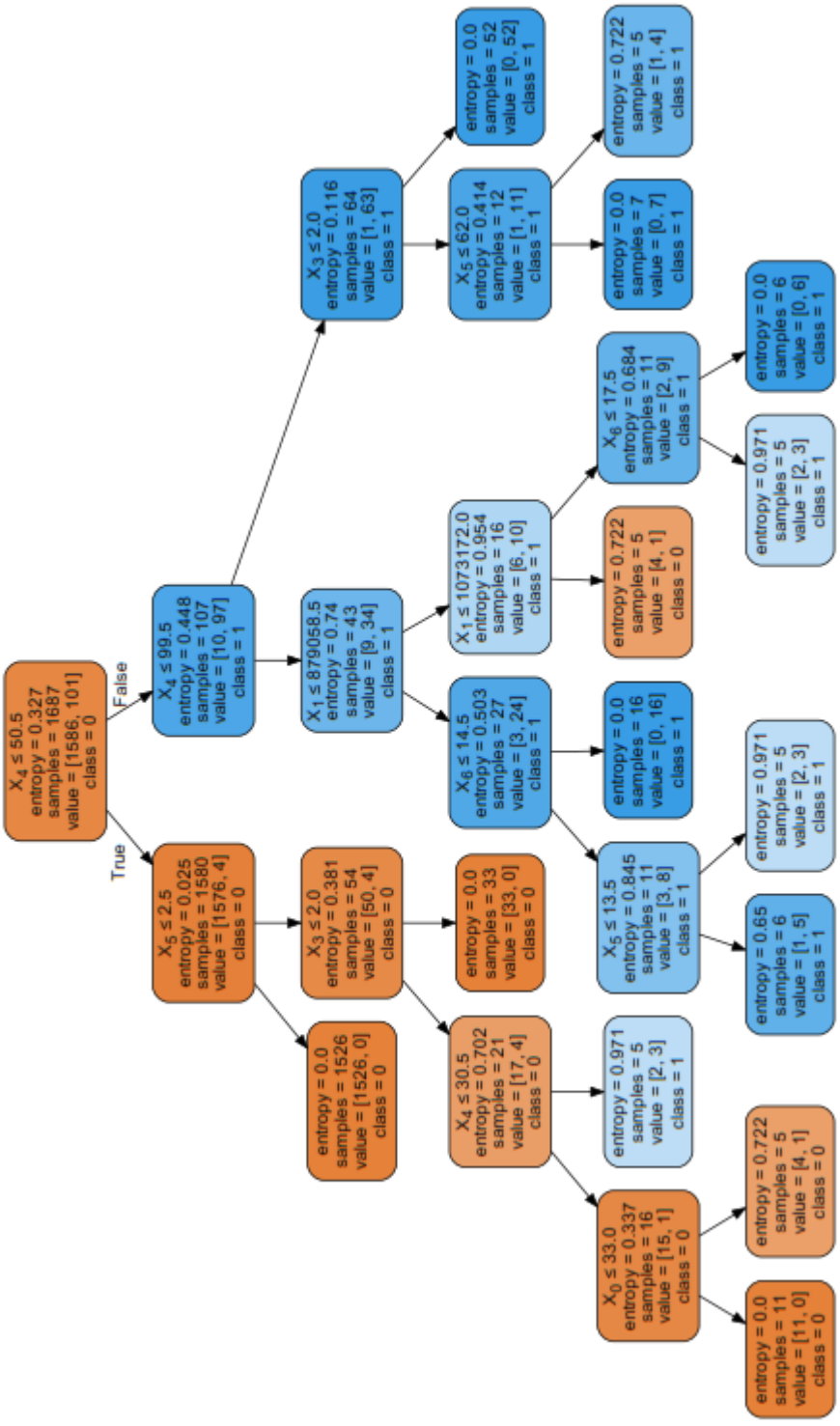

Figura 34. Árbol de decisión semana 30

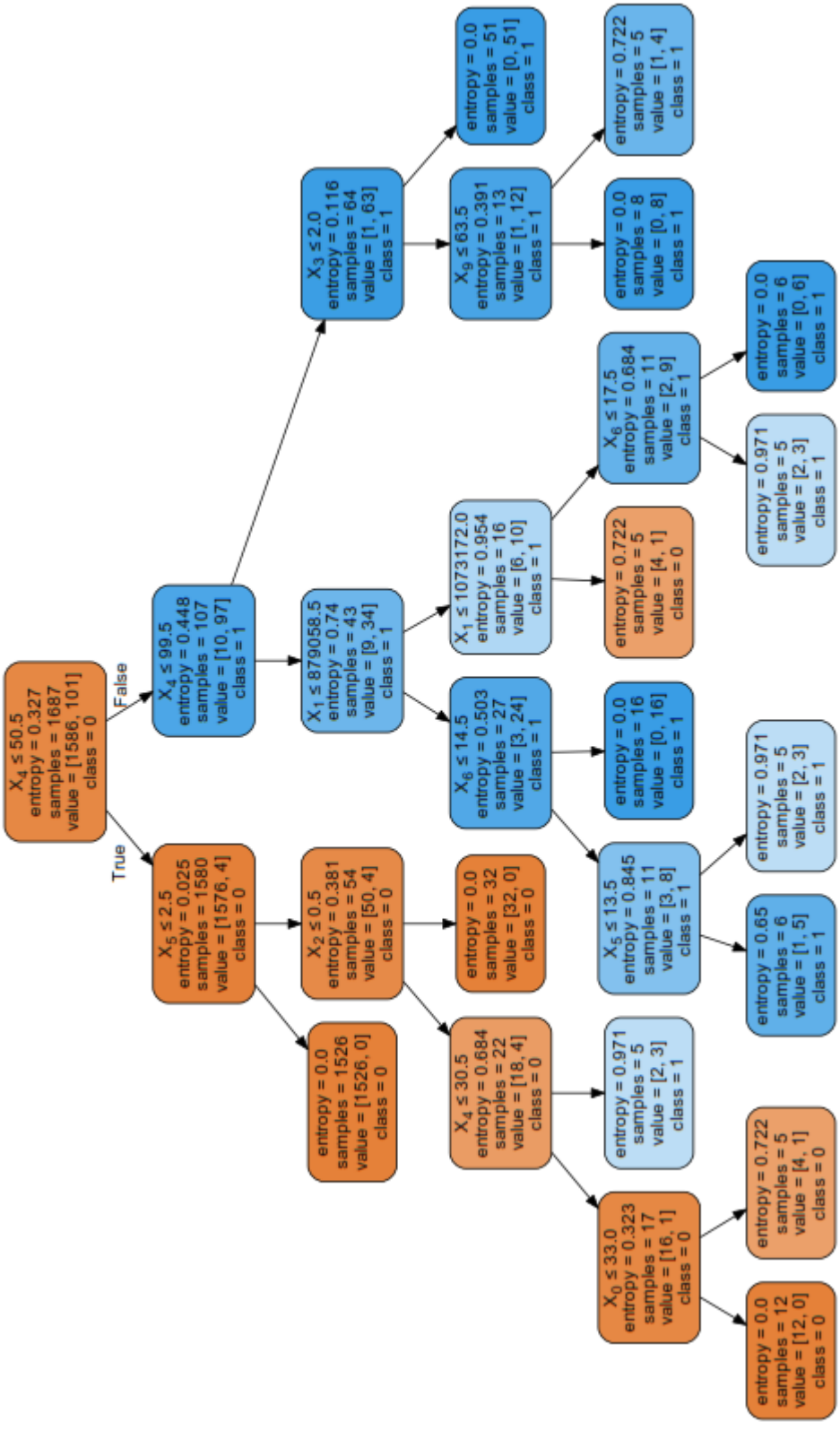

Figura 35. Árbol de decisión semana 42