Vol. 19, n.° 4, pp. 37-52, 2020
Revista UIS Ingenierías
Página de la revista: revistas.uis.edu.co/index.php/revistauisingenierias

Predicción de riesgo crediticio en Colombia usando técnicas de inteligencia artificial

Credit risk prediction in Colombia using artificial intelligence techniques

Diego Borrero-Tigreros [1], Oscar Bedoya-Leiva [2]

[1] Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Colombia. Orcid: 0000-0002-2934-3977. Correo electrónico: diego.borrero@correounivalle.edu.co
[2] Grupo de Univalle en Inteligencia Artificial, Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Colombia. Orcid: 0000-0002-9378-7337. Correo electrónico: oscar.bedoya@correounivalle.edu.co

Recibido: 12 marzo, 2020. Aceptado: 16 junio, 2020. Versión final: 6 julio, 2020.

Resumen

En este artículo se proponen modelos para la predicción de riesgo crediticio en Colombia utilizando diferentes técnicas de inteligencia artificial. Estos modelos se pueden usar como apoyo por el área de gestión de riesgo en los bancos y tienen como objetivo identificar clientes que podrían incurrir en un estado de mora generando un posible riesgo de crédito para las entidades financieras. En particular, se proponen modelos basados en tres técnicas de aprendizaje supervisado (redes neuronales, árboles de decisión y máquinas de soporte vectorial) para predecir el próximo pago de la cuota de un cliente a partir de datos básicos de la operación, del cliente y de pagos de cuotas anteriores registradas. De acuerdo con los resultados obtenidos, los árboles de decisión resultan ser más exactos que las otras técnicas utilizadas para la predicción de riesgo crediticio con un área bajo la curva ROC de 88.29%. Los modelos propuestos alcanzan exactitudes similares y en algunos casos superan las exactitudes reportadas en algunos trabajos del estado del arte.

Palabras clave: árboles de decisión; entidades financieras; estado de mora; inteligencia artificial; modelos; predicción, riesgo crediticio; riesgo de no pago; redes neuronales; toma de decisiones.

Abstract

In this paper, new models for credit risk prediction in Colombia are proposed by using different artificial intelligence techniques. These models can be used to support the risk management area in banks, and they aim to identify clients that could be in default, generating a possible credit risk for financial institutions. Three techniques are used to obtain the models (neuronal networks, decision trees, and support vector machines) that predict the next payment of a client's fee based on basic data from the client and previous recorded installment payments. Decision trees turns out to be more accurate than the other techniques that have been used when predicting credit risk with a ROC area of 88.29%. The proposed models reach accuracies that are like some other papers in the state of the art and in some cases, they overcome models in other studies.

Keywords: artificial intelligence; credit risk; decision making; decision trees; default; financial institutions; models; loan default; neural networks, prediction.

ISSN impreso: 1657 - 4583. ISSN en línea: 2145 – 8456, **CC BY-ND 4.0**
Como citar: D. Borrero, O. Bedoya, "Predicción de riesgo crediticio en Colombia usando técnicas de inteligencia artificial," *Rev. UIS Ing.*, vol. 19, no. 4, pp. 37-52, 2020, doi: https://doi.org/10.18273/revuin.v19n4-2020004

1. Introducción

Las compañías de financiamiento comercial son aquellas entidades que tienen como función principal captar recursos a término con el objetivo de realizar operaciones activas de crédito para facilitar la comercialización de bienes y servicios.

Para las compañías que ofrecen servicios de crédito financiero es muy importante tener una liquidez financiera que le permita cumplir con todas las obligaciones adquiridas y tener un flujo de efectivo constante para continuar prestando el servicio. Además, este tipo de entidades también tienen como propósito disminuir el riesgo de crédito por ser ésta una característica propia de la actividad que desarrollan.

Por lo tanto, es importante realizar procesos y análisis que permitan disminuir el riesgo de liquidez y crédito ya que son las principales incertidumbres a las cuales se enfrentan. Poder detectar clientes con insolvencia crediticia antes de que incumplan con sus obligaciones contractuales es un desafío para estas entidades, que acaban realizando procesos de gestión de cartera para los deudores morosos y en algunos casos procesos jurídicos para recuperar el capital y los intereses causados por la operación de crédito.

Uno de los principales objetivos de los establecimientos bancarios, es otorgar préstamos a los clientes, el desafío es establecer a quién suministrar un crédito, sin asumir un alto riesgo económico. Otro problema que enfrentan las entidades bancarias es la predicción de fraude crediticio donde se intenta identificar patrones en las transacciones bancarias para distinguir entre una actividad fraudulenta y una que no lo es.

Se han usado diversas técnicas de inteligencia artificial sobre conjuntos de datos generados a partir de entidades financieras [1-18]. Los modelos propuestos se centran en dos enfoques relacionados con el área del riesgo crediticio. El primer enfoque, conocido como riesgo de no pago, se centra en predecir si un cliente que ya tiene un crédito podría llegar a incurrir en una situación de no pago de una de sus cuotas. En el segundo enfoque, conocido como riesgo del crédito, se proponen modelos que sirven de apoyo a las entidades financieras para decidir si un cliente debería, o no, recibir el monto solicitado. En ambos enfoques se utilizan datos históricos del comportamiento de los clientes y se entrenan modelos a través de técnicas de inteligencia artificial.

En [1] se aborda el problema de riesgo de no pago. Se utilizan algoritmos de clasificación supervisada y un total de 23 atributos para representar a cada cliente incluyendo variables tales como el monto otorgado, género, nivel de escolaridad, estado civil, edad, historial de pago de los últimos seis meses y estado de cuenta de los últimos seis meses. Los modelos propuestos permiten decidir si un cliente incumplirá, o no, con el pago de la siguiente cuota y alcanzan una exactitud que va del 76% al 80%.

En [7] se utilizan redes neuronales artificiales y árboles de decisión para la predicción de riesgo de no pago. En este estudio, cada cliente se representa por medio de 13 atributos que incluye características como el monto, el uso dado al dinero prestado, la tasa de interés, tipo de vivienda, situación laboral, balance crediticio, entre otra información. La técnica de redes neuronales alcanza una exactitud del 81.2%.

En [12] se aborda el problema del riesgo del crédito. Se emplean como técnicas los árboles de decisión y las redes bayesianas. Cada cliente se represa por medio de siete atributos que incluye información sobre el historial crediticio, género, monto, edad, tipo de vivienda, y si tiene actualmente trabajo, o no, entre otros datos. Los modelos propuestos alcanzan una exactitud que va del 73% al 78%.

En [15] se construye una red neuronal multicapa tipo feed-forward que con base en 11 atributos tales como la edad, tipo de cuenta, ingreso, nacionalidad, tipo de vivienda, experiencia laboral, balance crediticio y el monto del préstamo solicitado, es capaz de determinar si la entidad financiera debería hacer el préstamo, o no. La red neuronal alcanza una exactitud del 95% comparando las salidas de dicha red con las decisiones de un experto en un conjunto de prueba.

Actualmente, existe la necesidad, por parte de las entidades financieras en Colombia y el mundo, de tener un mecanismo que permita predecir qué clientes pueden incurrir en morosidad. En este artículo se aborda específicamente el problema del riesgo de no pago. Este problema es de vital importancia para las entidades financieras ya que se deriva en la pérdida de dinero causada por la insolvencia crediticia de sus clientes. El problema se evidencia en un reporte dado por la Superintendencia Financiera según el cual en agosto de 2016 el valor de las deudas en mora con la banca sumó 13,2 billones de pesos. Además, se conoce que este saldo crece a un ritmo mayor que el de la colocación de nuevos créditos.

En particular, los créditos vencidos crecieron siete veces más que los préstamos nuevos [19]. Para esto, en este artículo se proponen modelos de predicción, usando técnicas de aprendizaje supervisado, que permiten conocer qué clientes podrían incurrir en morosidad y así aplicar estrategias persuasivas de pago para disminuir el riesgo de crédito. A pesar de que existen trabajos enfocados hacia la predicción de riesgo crediticio, no hay evidencias de este tipo de trabajos en Colombia, entendiendo las particularidades que pueden llegar a tener los clientes de cada país, lo que hace necesario crear modelos propios que representen su comportamiento.

2. Metodología

En esta sección se presenta la metodología usada para obtener los modelos propuestos para la predicción de riesgo crediticio. Los modelos se generaron utilizando tres tipos algoritmos de clasificación supervisada (redes neuronales, árboles de decisión y máquinas de soporte vectorial) y tienen como objetivo predecir si un cliente va a incumplir, o no, con el pago de la cuota de un crédito. Para esto, se utilizan datos básicos del cliente, sus operaciones y los pagos previos generados.

2.1. Selección de datos

Para la selección de los datos se cuenta con una base de datos relacional anónima la cual hace parte del ciclo de vida de los créditos registrados en una entidad financiera en Colombia. Para el problema de gestión de riesgo crediticio es necesario obtener los datos básicos de los clientes con información de la operación crediticia y la información detallada de los pagos realizados en un periodo de tiempo. La base de datos utilizada en este estudio cuenta con las tablas Clientes, Operaciones y DetIngresosClientes con las cuales se procede a realizar el diseño del modelo estrella en el entorno integrado SSMS (SQL Server Management Studio). Este modelo estrella permite pre-procesar los datos, integrar diferentes fuentes de información y facilitar la aplicación de los algoritmos de aprendizaje supervisado que se usan para obtener los modelos.

La Figura 1 muestra el modelo estrella generado a partir del análisis del proceso de pagos crediticios. En el diseño del modelo estrella se establecieron la tabla de hechos y de dimensiones. La tabla de hechos, llamada Ingresos, contiene información detallada sobre los pagos de las cuotas de cada cliente tal como el valor pagado de la cuota, los días de mora, el saldo capital de la cuota cancelada y fechas de cancelación.

Las tablas dimensión están compuestas por la tabla DimClientes que contiene información básica de los clientes, la tabla DimOperación que tiene datos de las operaciones de crédito y la tabla DimFecha que contiene fechas con un nivel de granularidad de días.

El proceso de transformación, extracción y carga de datos se realizó a través de procedimientos almacenados que extraen información válida de la base de datos y transforman datos cualitativos nominales categorizándolos numéricamente para la carga de datos al modelo estrella.

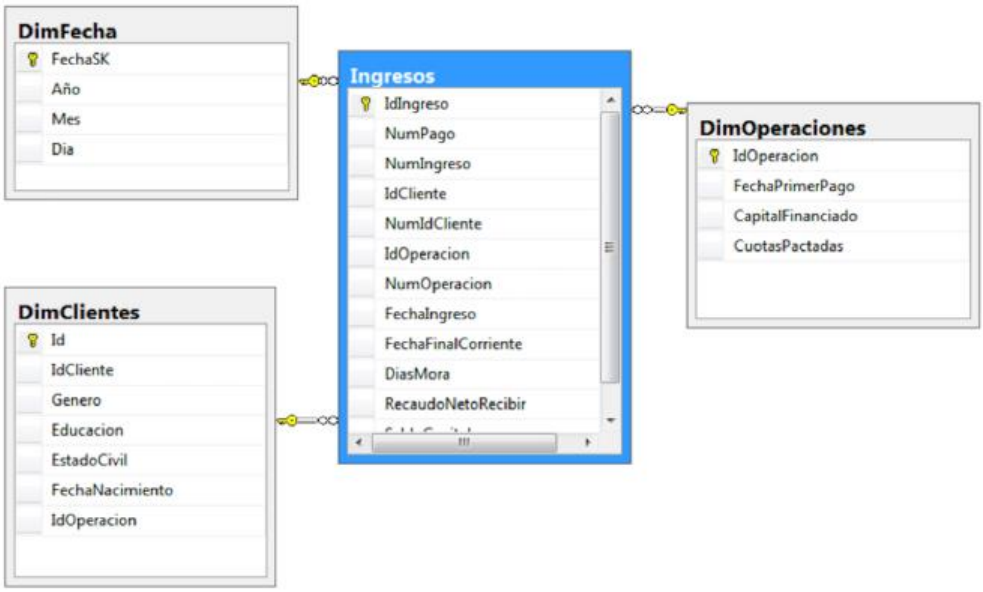

Figura 1. Modelo estrella para la selección de datos. Fuente: elaboración propia.

Tabla 1. Definición de los atributos utilizados

Atributo	Descripción	Tipo de dato
X1	Capital financiado en pesos colombianos	Numérico
X2	Género (Masculino = 1, Femenino = 2)	Nominal
X3	Educación (Primaria = 1, Secundaria = 2, Universitaria = 3, Maestría = 4)	Nominal
X4	Estado Civil (Casado = 1, Soltero = 2, Otro = 3)	Nominal
X5	Edad (Años)	Numérico
X6 a X11	Historial de los últimos 6 meses de pago clasificados por estado de amortización (pago antes de vencimiento = -2, no pago = -1, pago cumplido de la cuota = 0, retraso de pago por un mes = 1, retraso de pago por dos meses = 2,... retraso de pago por ocho meses = 8, retraso de pago por nueve meses = 9 etc.)	Nominal
X12 a X17	Importe de estado de cuenta en pesos colombianos de los últimos 6 meses	Numérico
X18 a X23	Monto pagado de los últimos 6 meses	Numérico
X24	Cuotas pactadas	Numérico
Etiqueta de clase	Etiqueta a predecir. Corresponde al incumplimiento de la cuota siguiente a los últimos 6 meses de pago (cuota no incumplida = 0, cuota incumplida = 1)	Nominal

Fuente: elaboración propia.

Los conjuntos de entrenamiento y prueba contienen un total de 561 registros almacenados en un archivo CSV. Cada registro corresponde a un cliente del banco el cual se representa por medio de 25 atributos que incluye datos básicos, sus operaciones y los pagos generados. Estos atributos se muestran en la Tabla 1. El atributo X1 (capital financiado) indica el valor en pesos colombianos del crédito aprobado por la entidad financiera. Como parte de los datos básicos se tiene el género (X2), el nivel educativo (X3), el estado civil (X4) y la edad (X5). Los estados de amortización de los últimos seis pagos registrados indican si cada pago se generó antes de vencimiento, en fecha de vencimiento o si se pagaron con mora (X6 a X11). El importe de estado de cuenta indica el saldo en capital después de cada pago (X12 a X17). Los montos pagados por el cliente en los últimos seis meses (X18 a X23). Se incluye también las cuotas pactadas del crédito (X24). La etiqueta de clase que se quiere predecir corresponde al pago después de los últimos seis meses y se obtiene de los ingresos del cliente categorizando el pago del mes número siete. Si hay un recaudo se asigna “0”, lo cual significa que el cliente no incumplió el pago de la cuota y si no hay un recaudo se asigna “1”, lo cual significa que el cliente incumplió el pago de la cuota.

2.2. Construcción de los modelos de predicción

2.2.1. Modelos de predicción usando redes neuronales

Para la construcción del modelo usando redes neuronales se utilizó una red perceptrón multicapa (MLP) y algunas reglas adhoc para el diseño de la topología. Según [20] uno de los principales problemas al construir una red MLP es el diseño de la topología ya que de esto depende la capacidad de generalización del modelo. De acuerdo con [21], las redes con una sola capa oculta resultan suficientes para resolver problemas arbitrariamente complejos siempre que el número de nodos ocultos sea al menos tres veces el número de nodos de entrada. Por otra parte, en [22] se aplica una extensión del teorema de Kolmogorov para demostrar que una red con una capa oculta compuesta por 2N+1 neuronas, donde N es el número de neuronas de entrada, y con funciones de transferencias continuas no lineales y crecientes resulta óptimo para calcular cualquier función continua de N variables de entrada.

Para determinar el número de neuronas ocultas de la red MLP propuesta en este artículo se utilizaron dos reglas adhoc que según [20] han demostrado un buen comportamiento en diversas aplicaciones. La primera regla adhoc utilizada se conoce como la pirámide geométrica y se basa en la suposición de que el número de neuronas de la capa oculta debe ser inferior al número de neuronas en la capa de entrada y mayor que el número de neuronas en la capa de salida. Además, el número de neuronas en cada capa sigue una progresión geométrica, tal que para una red de una sola capa oculta el número de neuronas en ella debe ser un aproximado de $\sqrt{N} * M$, donde N es el número de neuronas en la capa de entrada y M el número de neuronas en la capa de salida. La segunda regla adhoc utilizada se conoce como Capa oculta – Capa de entrada y consiste en relacionar el número de neuronas de la capa oculta con el número de neuronas en la capa de entrada. Por lo regular suele

aplicarse la regla 2x1 de tal forma que el número de neuronas ocultas no puede ser superior al doble del número de variables de entrada.

Para la construcción de la red perceptrón multicapa se utilizó la librería neuralnet del lenguaje de programación R. Se probaron diferentes topologías de red variando el número de capas ocultas y el número de neuronas siguiendo las dos estrategias adhoc mencionadas anteriormente. Igualmente se evaluaron diferentes valores del parámetro threshold, el cual especifica el umbral para las derivadas parciales de la función de error.

Los datos usados para la construcción de los modelos tienen una proporción de 28% de casos positivos, estos son clientes que incumplieron el pago de la cuota, y 72% de casos negativos, es decir, clientes que no incumplieron el pago de la cuota. Este desbalance se presenta debido a que son menos los clientes que incumplen los pagos que aquellos que cumplen con el pago de sus cuotas.

Utilizando las reglas adhoc se generaron un total de 998 redes MLP obteniendo topologías de redes con rangos neuronas ocultas desde la raíz cuadrada del número de neuronas de entrada ($\sqrt{N} * M$) hasta el doble del número de neuronas de entrada más uno (2N+1), con una y dos capas ocultas y modificando el hiperparámetro threshold en rangos de 30% de error hasta 10%.

Para cada red obtenida se calcularon las métricas que se presentan en la sección 3 y después de un análisis de resultados se seleccionaron las diez redes neuronales con mayores valores de sensibilidad, exactitud y precisión. La topología de red seleccionada para el problema de la predicción de riesgo crediticio se presenta en la Figura 2. Esta red cuenta con 24 neuronas en la capa de entrada que corresponden a los atributos presentados en la Tabla 1 y dos capas ocultas. La primera capa oculta tiene 34 neuronas y la segunda 25. La capa de salida tiene una sola neurona. Esta neurona toma el valor "1" cuando el cliente incumple con el pago de la cuota pactada y "0" cuando el cliente no incumple.

2.2.2. Modelos de predicción usando árboles de decisión

En esta sección se presentan tres modelos de predicción de riesgo crediticio que se basan en los algoritmos de árboles de decisión C4.5, Random Forest y C5.0.

2.2.2.1. Árboles de decisión aplicando el algoritmo C4.5

Para la construcción del modelo de árboles de decisión aplicando el algoritmo C4.5 se utilizó la librería rpart del lenguaje R. Dentro de la experimentación se incluyó el hiperparámetro minsplit que es el número mínimo de observaciones que deben existir en un nodo para que se produzca o se intente generar una división.

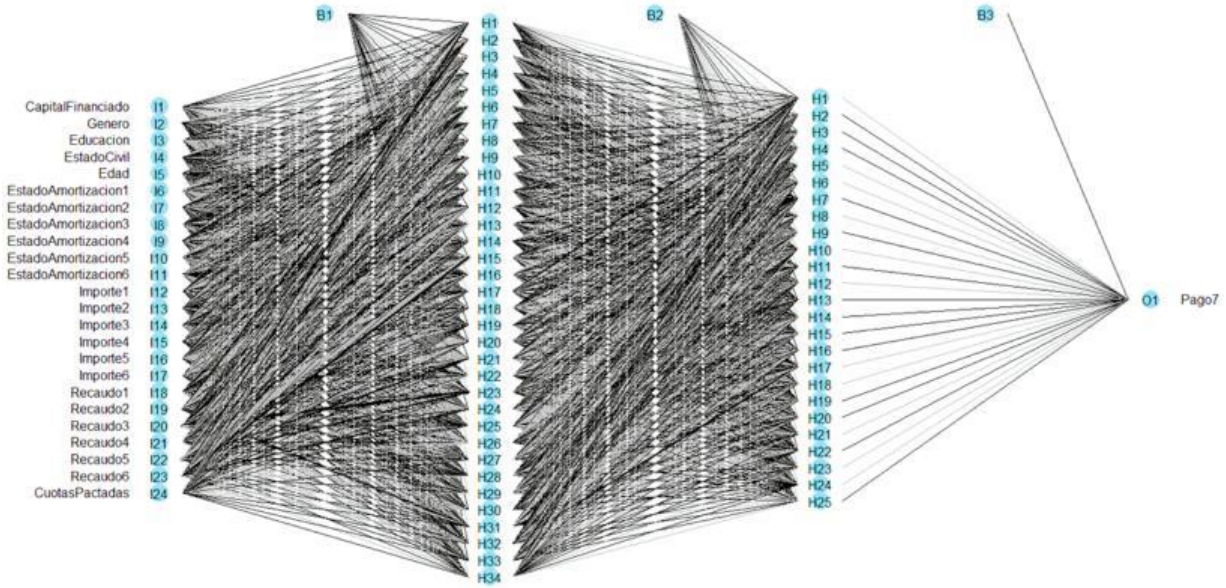

Figura 1. Modelo de red neuronal propuesta. Fuente: elaboración propia.

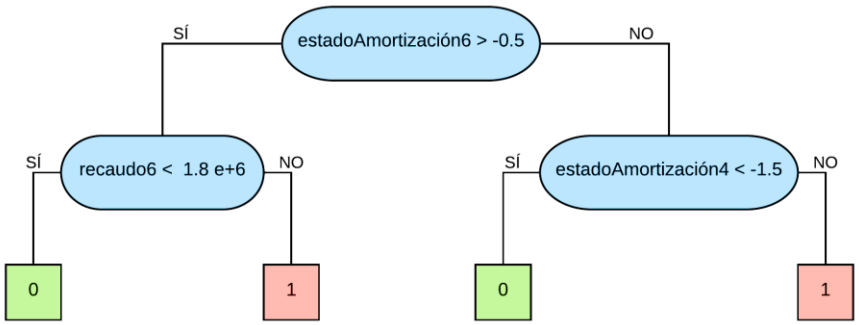

Figura 3. Modelo de árbol C4.5 propuesto. Fuente: elaboración propia.

El valor por defecto de este hiperparámetro es 20 y se experimentó desde esa cantidad hasta el número máximo de observaciones en los datos de entrenamiento que son 504 en esta investigación. El árbol de decisión generado se muestra en la Figura 3 y consta de diferentes pruebas sobre tres de las variables independientes que se presentaron en la Tabla 1 (EstadoAmortización6, Recaudo6, y EstadoAmortización4).

La clasificación de un nuevo cliente se realiza sometiendo sus datos a la prueba que hay en cada nodo del árbol y siguiendo la rama que cumpla con las condiciones generadas hasta llegar a un nodo hoja. Cada hoja del árbol representa la variable dependiente que se quiere predecir.

2.2.2.2. Árboles de decisión aplicando el algoritmo Random Forest

Para la construcción del modelo de árboles de decisión aplicando el algoritmo Random forest se utilizó la librería random forest del lenguaje R. El algoritmo Random Forest permite hacer la predicción empleando, no solamente un árbol como en el caso del algoritmo C4.5, sino un conjunto de ellos de tal forma que cada árbol se utilice como un experto y al integrarlos se tengan más elementos para decidir sobre un nuevo cliente. Este algoritmo tiene tres hiperparámetros por establecer, estos son, ntree, mtry, y nodesize.

Para la generación del modelo de predicción de riesgo crediticio aplicando el algoritmo Random Forest se realizó inicialmente una optimización del hiperparámetro ntree el cual indica el número de árboles a generar. Para establecer el valor de este hiperparámetro se probaron diferentes valores que van desde 1000 árboles hasta 2500 con el objetivo de evitar consumir recursos computacionales innecesarios. En la Tabla 2 se presentan los resultados obtenidos para diferentes valores del parámetro ntree. El valor más acertado fue 2500 con una precisión media de 75.21% y un coeficiente kappa de 31.48% que muestra una fuerza de concordancia mediana.

Tabla 2. Exactitudes para diferentes valores del parámetro ntree

ntree	Precisión media	Coeficiente kappa
1000	0.7488	0.3079
1500	0.7494	0.3113
2000	0.7514	0.3158
2500	0.7521	0.3148

Fuente: elaboración propia.

El hiperparámetro mtry indica el número de variables tomadas de forma pseudoaleatoria para ser candidatos en cada división. Para ajustar este hiperparámetro se define una función que calcula un arreglo de la tasa de error de los casos que no son considerados para entrenar el árbol al cual se le llama out-of-bag-error (error OOB). El hiperparámetro mtry va desde 1 hasta la cantidad máxima de las variables independientes que son 24 en esta investigación. La Figura 4 muestra la evolución del error OOB para los diferentes valores de mtry. Se puede observar que cuando se consideran 11 variables se obtiene el menor error OOB.

El algoritmo Random Forest tiene además el hiperparámetro nodesize que indica el tamaño mínimo de nodos terminales. Para ajustar este hiperparámetro se define una función que calcula un arreglo de error OOB por cada árbol generado en un rango que va desde 1 hasta 20 tal como se muestra en la Figura 5. Se puede observar que el nodesize que alcanza un menor error OOB es 14.

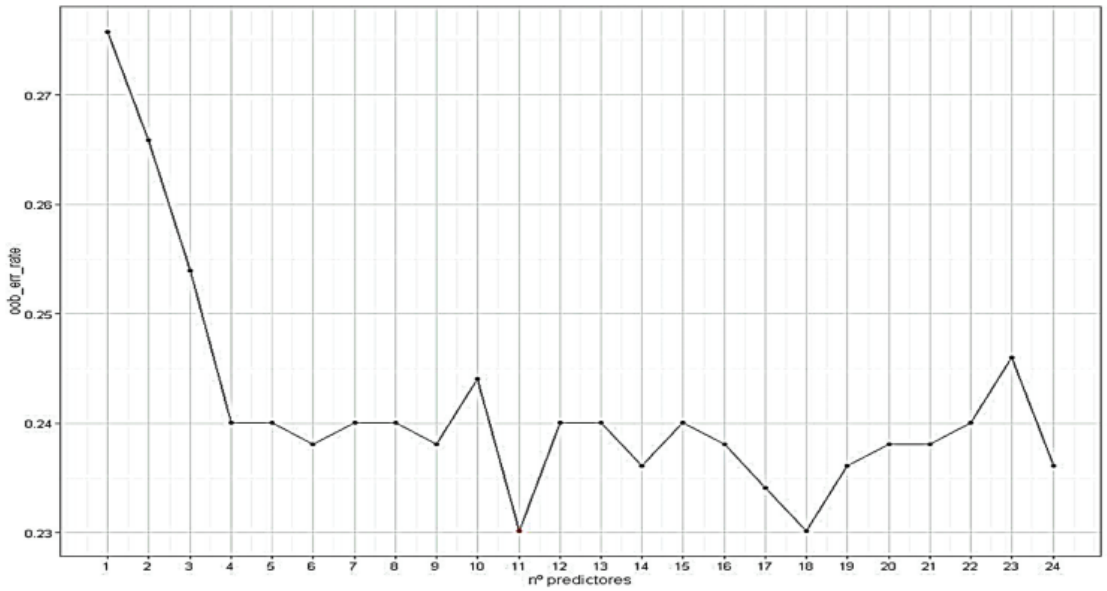

Figura 4. Análisis del error OOB y el parámetro mtry. Fuente: elaboración propia.

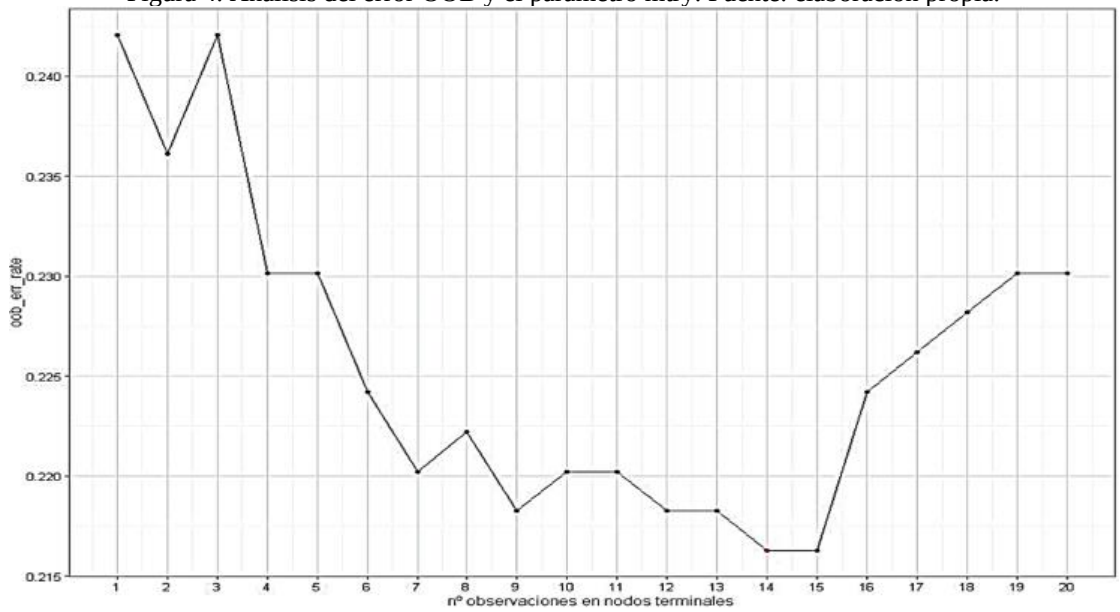

Figura 5. Análisis del error OOB y el parámetro nodesize. Fuente: elaboración propia.

Con base en el ajuste de hiperparámetros, el modelo para para la predicción de riesgo crediticio empleando el algoritmo Random Forest utiliza los valores 2500, 11 y 14, para los parámetros ntree, mtry, y nodesize, respectivamente. Este modelo no se puede presentar gráficamente, pero en la siguiente sección se detallan los resultados obtenidos.

2.2.2.3. Árboles de decisión aplicando el algoritmo C5.0

Para la construcción del modelo de predicción utilizando el algoritmo C5.0 se hace uso de la librería C50 en lenguaje R. Este es uno de los algoritmos de árboles de decisión más ampliamente usados en diversas investigaciones debido a que se considera como una mejora al algoritmo C4.5. El algoritmo C5.0 tiene el hiperparámetro costs con el cual se puede definir el peso de los errores para enfatizar ciertas clases sobre otras. Para el problema que se aborda en esta investigación, realizar una predicción que clasifique a una persona como un cliente que no va a incumplir y termine incumpliendo es más costoso para una entidad financiera que una persona que se clasifique como un cliente que incumplirá y que cumpla con el pago. Teniendo en cuenta lo anterior se define una matriz de costos de 2x2 ya que

se manejan dos etiquetas de clase, los clientes que pagan y los que no pagan. La matriz de costos se presenta en la Tabla 3.

El árbol de decisión generado se muestra en la Figura 6 y consta de 12 de las variables presentadas en la Tabla 1 (EstadoAmortización6, Recaudo6, recaudo5, Edad, Educación, Recaudo4, EstadoAmortización5, EstadoAmortización4, Importe, EstadoAmortización2, Recaudo1, CuotasPactadas). La clasificación de un nuevo cliente se realiza sometiendo sus datos a las pruebas que hay en cada nodo siguiendo la rama del árbol que cumpla con las condiciones generadas hasta llegar a un nodo hoja. Las hojas del árbol representan la variable dependiente o etiqueta de clase.

Tabla 3. Matriz de costos definida como hiperparámetro

	Valor Real	
Valor Predicho	**No**	**Si**
No	0	4
Si	1	0

Fuente: elaboración propia.

2.2.3. Modelos de predicción usando máquinas de soporte vectorial

Para la construcción del modelo de predicción empleando la técnica de máquinas de soporte vectorial se utilizó la librería e1071 del lenguaje R. Para encontrar la mejor configuración de los hiperparámetros se emplea la función tune de la librería e1071 que realiza un ajuste de los hiperparámetros cost, gamma y la función kernel utilizando una búsqueda de cuadrícula. Los hiperparámetros permiten variar el margen de separación entre observaciones. El parámetro cost es el peso que se le da a cada observación a la hora de clasificar e incide en el error de la función de regresión. Si este valor es muy grande, mayor es el peso de una observación. De lo contrario, un cost demasiado pequeño clasificaría erróneamente un número muy elevado de observaciones. El gamma es otro hiperparámetro que permite suavizar la sobrestimación e influye en la distancia entre las observaciones que separan los subespacios del modelo. Un menor valor gamma implica mayor distancia entre las observaciones y por ende la estimación es más conservadora.

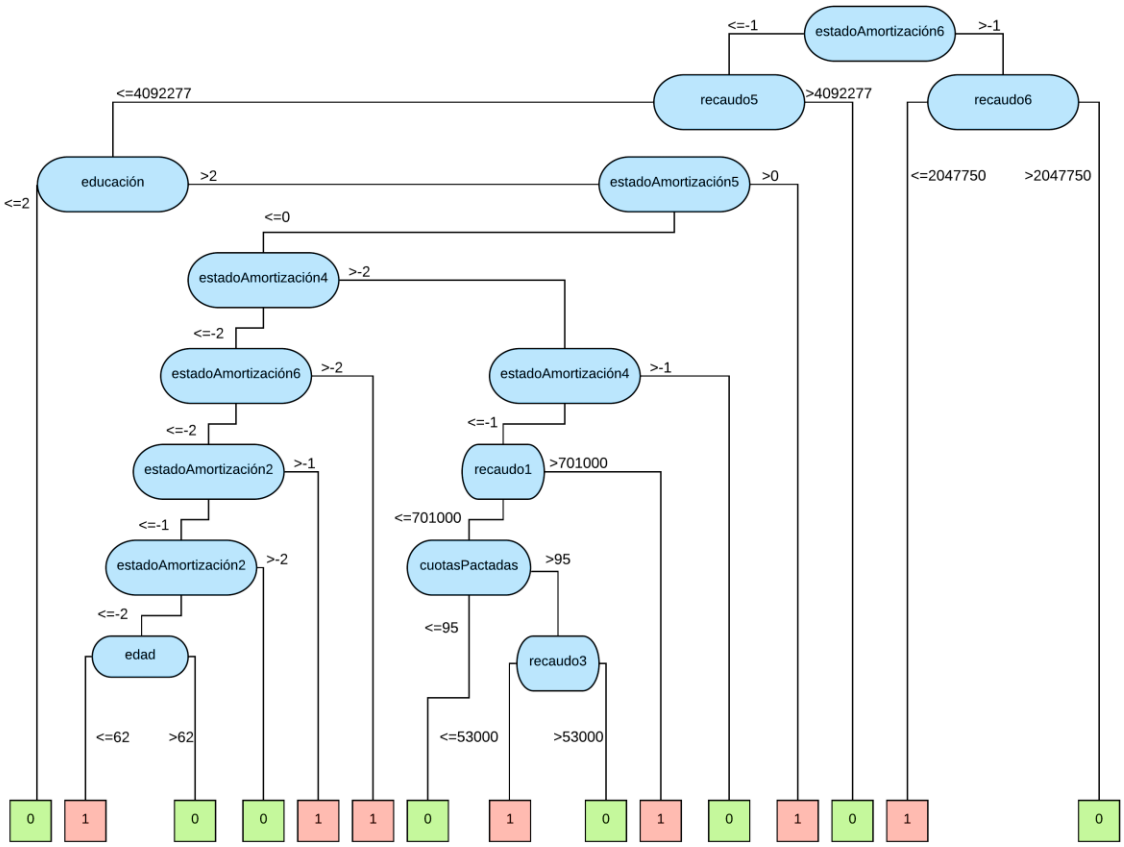

Figura 6. Modelo de árbol C5.0 propuesto. Fuente: elaboración propia.

Sin embargo, un valor de gamma muy elevado genera predicciones menos suavizadas y esto hace que el modelo se sobrestime. Utilizando la función tune se logró encontrar que el kernel que mejor se ajusta para este problema es el sigmoide con valores cost y gamma de 8 y 4, respectivamente.

3. Resultados

En esta sección se presentan los criterios de evaluación usados durante la experimentación. Además, se muestran los valores obtenidos por los modelos propuestos para la predicción de riesgo crediticio. En las pruebas realizadas se utilizó validación cruzada con k=5. Esto garantiza que cada instancia del conjunto de datos sea consideraba como parte de las pruebas.

3.1. Criterios de evaluación

El criterio de comparación que se usó para los modelos predictivos de riesgo crediticio es el análisis de la curva ROC calculando el área bajo la curva (AUC) ya que es un indicador de la capacidad predictiva de cada modelo. Los parámetros calculados en cada prueba son los verdaderos positivos (VP) que corresponden a los casos en los que el modelo clasifica a un cliente con la etiqueta "1" (incumple) y realmente incumple, Falsos positivos (FP) que son los casos en los que el modelo clasifica un cliente con la etiqueta "1" (incumple) y realmente no incumple, verdaderos negativos (VN) que son los casos en los que el modelo clasifica un cliente con la etiqueta "0" (no incumple) y realmente no incumple y falsos negativos (FN) que son los casos en los que el modelo clasifica un cliente con la etiqueta "0" (no incumple) y realmente incumple.

Con estos parámetros se calcula la sensibilidad (Sn=VP/(VP+FN)), esto es, la probabilidad de que el modelo clasifique correctamente un cliente que incumplirá el pago de su cuota, la especificidad (Sp=VN/(VN+FP)) que es la probabilidad de que el modelo clasifique correctamente un cliente que no incumplirá el pago de su cuota, la exactitud ((VP+VN)/(VP+VN+FN+FP)) que mide la fracción de predicciones correctas, la precisión (VP/(VP+FP)) que calcula la fracción de los verdaderos positivos entre los casos que se prevén positivos, la tasa de error ((FP+FN)/(VP+VN+FN+FP)) que es el promedio de las clasificaciones incorrectas del modelo, y el valor de predicción negativo (VNP=VN/(FN+FP), que corresponde al porcentaje de clasificaciones correctas de clientes que no incumplirán el pago de las cuotas.

Por su parte, la curva ROC representa el valor 1-especificidad frente a la sensibilidad para cada punto de corte en la escala de resultados de los modelos. Cada resultado de predicción representa un punto en el espacio ROC. El mejor método posible de predicción se situaría en la esquina superior izquierda del espacio ROC. El área bajo la curva ROC refleja qué tan bueno es el modelo para discriminar clientes que puedan caer en un estado moratorio al no pagar la próxima cuota pactada. La curva ROC y el parámetro AUC se calcula para cada modelo con la librería ROCR del lenguaje R.

3.2. Pruebas y resultados usando redes neuronales

En la Tabla 4 se muestran los resultados obtenidos de las 10 mejores redes MLP generadas por cada threshold (0.1, 0.2, 0.3) sobre el conjunto de prueba. La tabla está ordenada de forma descendente por sensibilidad y exactitud. El primer lugar lo alcanza el modelo de red neuronal de la Figura 2 cuya exactitud es de 63.62% y una tasa de error de 36.38%. La sensibilidad, una de las métricas más importante en este problema, es de 71.83%.

Tabla 4. Resultados obtenidos con las redes neuronales

Capa 1	Capa 2	Threshold	Sensibilidad	Especificidad	Exactitud	Tasa de error	Precisión	NPV
34	25	0.3	0.7183	0.6834	0.6362	0.3638	0.4358	0.8715
13	0	0.1	0.7083	0.6187	0.6417	0.3512	0.3908	0.8600
25	8	0.1	0.7083	0.6330	0.6524	0.3663	0.4000	0.8627
45	43	0.1	0.7083	0.7050	0.7058	0.4107	0.4533	0.8750
41	30	0.2	0.6875	0.6402	0.6524	0.3592	0.3975	0.8557
33	8	0.1	0.6875	0.6834	0.6844	0.3636	0.4285	0.8636
43	35	0.2	0.6875	0.6618	0.6684	0.3668	0.4125	0.8598
30	21	0.2	0.6875	0.6474	0.6577	0.3687	0.4024	0.8571
37	32	0.2	0.6875	0.6546	0.6631	0.3769	0.40.74	0.8584
29	22	0.3	0.6875	0.6618	0.6684	0.3810	0.4125	0.8598

Fuente: elaboración propia.

Por su parte, la precisión, especificidad y valor predicción negativo es de 43.58%, 68.34%, y 87.15%, respectivamente.

La Figura 7 muestra la curva ROC del modelo de red neuronal con su respectiva área bajo la curva. El análisis muestra la representación gráfica de la sensibilidad frente a la especificidad. El área bajo la curva se calcula para evaluar la bondad de la red neuronal. El modelo de red neuronal de la Figura 2, cuyos resultados se muestran en la Tabla 4, alcanzó un área bajo la curva de 0.6991 el cual es considerado un resultado de test aceptable ya que existe un 69.91% de probabilidad de que la predicción realizada a un cliente que incumplirá el pago de la cuota sea más correcta que el de un cliente escogido al azar que no incumplirá el pago.

3.3. Pruebas y resultados usando árboles de decisión

3.3.1. Pruebas con el algoritmo C4.5

En la Tabla 5 se muestran los resultados obtenidos por el modelo de árboles de decisión usando el algoritmo C4.5. De acuerdo con los valores presentados en la tabla, se selecciona el árbol con un minsplit de 78, ya que a pesar de tener una sensibilidad menor que el modelo con un minsplit de 150, se disminuye el riesgo de sobre ajuste. La exactitud alcanzada por el modelo de árboles de decisión es de 82.45% y la sensibilidad de 64.28%. Al comparar este resultado con la especificidad, que es de 88.37%, se puede conocer que el modelo detecta mejor los casos de clientes que no incumplen.

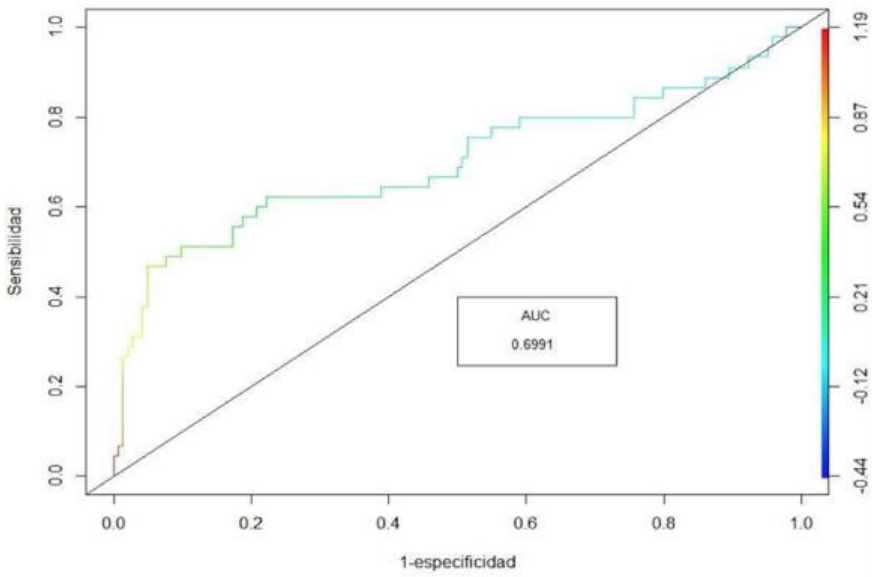

Figura 7. Análisis ROC del modelo de redes neuronales. Fuente: elaboración propia.

Tabla 5. Resultados obtenidos con el algoritmo C4.5.

Min split	Sn	Sp	Exact.	Tasa de error	Prec.	NPV
20	0.3571	0.9534	0.8070	0.1929	0.7142	0.8200
23	0.4285	0.9302	0.8070	0.1929	0.6666	0.8333
29	0.3571	0.9767	0.8245	0.1754	0.8333	0.8235
38	0.4285	0.9534	0.8245	0.1754	0.7500	0.8367
41	0.5714	0.9302	0.8421	0.1578	0.7272	0.8695
43	0.5000	0.9302	0.8245	0.1754	0.7000	0.8510
47	0.5000	0.9069	0.8070	0.1929	0.6363	0.8478
60	0.5000	0.9534	0.8421	0.1578	0.7777	0.8541
78	0.6428	0.8837	0.8245	0.1754	0.6428	0.8837
86	0.6428	0.8604	0.8070	0.1929	0.6000	0.8809
150	0.7142	0.8139	0.7894	0.2105	0.5555	0.8974

Fuente: elaboración propia.

Esto se debe principalmente al desbalance significativo que hay entre el número de casos de clientes que incumplen los pagos de sus cuotas y aquellos que no incumplen. Por su parte, la precisión alcanzada por el modelo es de 64.28% y el valor de predicción negativo de un 88.37%.

La Figura 8 muestra la curva ROC del modelo de árboles de decisión utilizando el algoritmo C4.5 y su respectivo parámetro AUC. El área bajo la curva de 78.24% supera el resultado obtenido por las redes neuronales que fue de 69.91%.

3.3.2. Pruebas con el algoritmo Random Forest

Al utilizar los hiperparámetros ntree, mtry y nodesize de 2500, 11 y 14, respectivamente, tal como se explicó en la sección 2.2.2.2., el algoritmo Random Forest alcanza una exactitud de 84.21%. Además, se obtiene una sensibilidad, especificidad, precisión, y valor de prediccion negativa de 57.14%, 93.02%, 72,72%, y 86.95%, respectivamente. La Figura 9 muestra la curva ROC del modelo de árboles de decisión utilizando el algoritmo Random Forest. El área bajo la curva es de 88.29%.

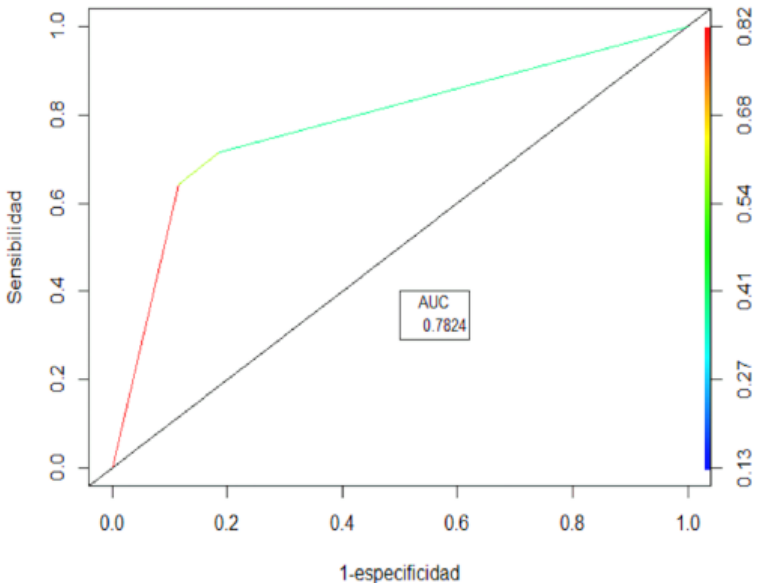

Figura 8. Análisis ROC del modelo de árbol C4.5. Fuente: elaboración propia.

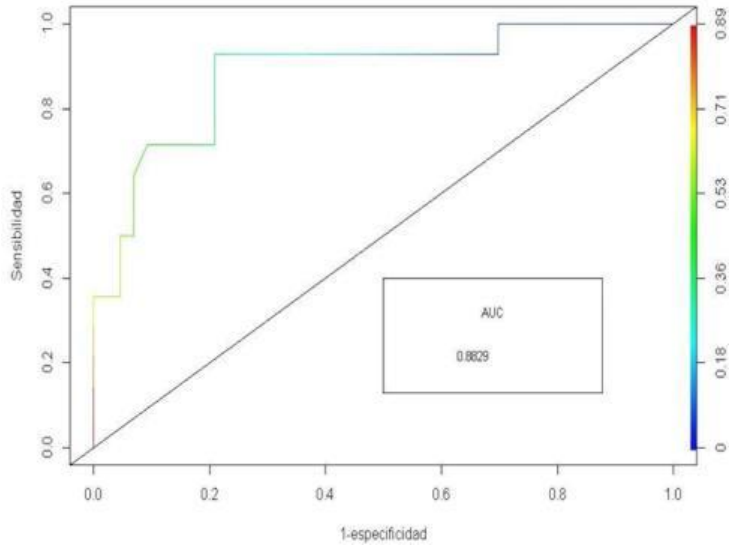

Figura 9. Análisis ROC del modelo de árbol Random Forest. Fuente: elaboración propia.

Este resultado supera significativamente el área bajo la curva de las redes neuronales y el algoritmo de árboles decisión C4.5.

3.3.3. Pruebas con el algoritmo C5.0

En la Tabla 6 se muestran los resultados obtenidos por el modelo de árboles de decisión utilizando el algoritmo C5.0 y variando el costo que penaliza las predicciones cuando el modelo clasifica un cliente con la etiqueta "0" (no incumple) y realmente el cliente incumple. La exactitud alcanzada por el modelo es de 63.15% y la tasa de error es de 8.77%. La sensibilidad alcanzó un 64.28%. Este resultado es superado por el modelo de redes neuronales, similar al algoritmo C4.5 y supera al algoritmo Random Forest. La especificidad, precisión, y precisión negativa fue de 46.51%, 28.12% y 80%, respectivamente.

El área bajo la curva obtenida por el modelo de árboles de decisión utilizando el algoritmo C5.0 es de 80.56%. Este resultado supera los valores obtenidos por el modelo de redes neuronales y el árbol de decisión con el algoritmo C4.5. La Figura 10 muestra la curva ROC del modelo de árboles de decisión utilizando el algoritmo C5.0 con su parámetro AUC.

3.4. Pruebas y resultados usando Máquinas de soporte vectorial

La Tabla 7 muestra los resultados por cada parámetro calculado empleando la función tune de la librería e1071.

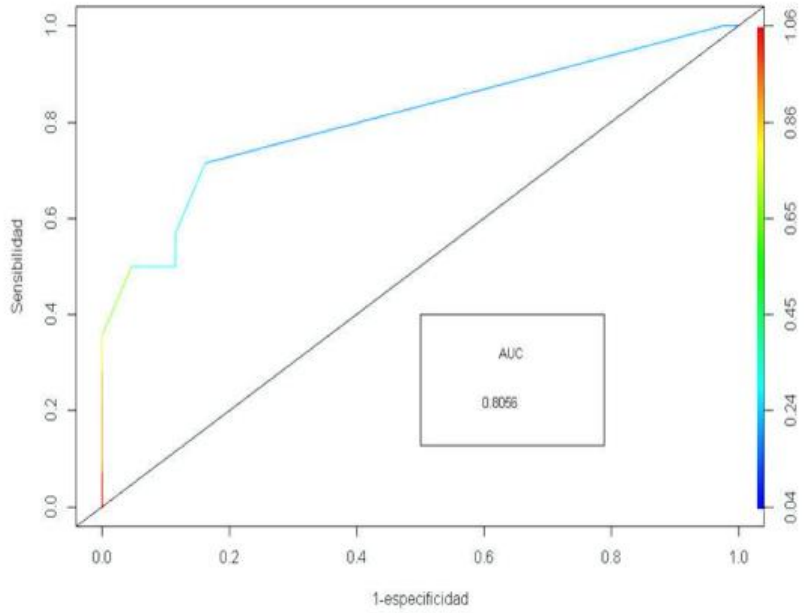

Figura 10. Análisis ROC del modelo de árbol C5.0. Fuente: elaboración propia.

Tabla 6. Resultados obtenidos con el algoritmo C5.0.

Costs	Exact.	Sn	Sp	Precisión	OOB
1	0.8421	0.3571	1.0000	1.0000	0.1578
2	0.7543	0.5000	0.8372	0.5000	0.1228
3	0.7368	0.5000	0.8139	0.4666	0.1228
4	0.6315	0.6428	0.4651	0.2812	0.0877
5	0.5614	0.7142	0.4883	0.3125	0.0701
6	0.5438	0.8571	0.4651	0.3428	0.0350
7	0.4385	0.8571	0.3023	0.2857	0.0350
8	0.4035	0.8571	0.2558	0.2727	0.0350
9	0.3508	0.8571	0.1860	0.2553	0.0350
10	0.3508	0.8571	0.1860	0.2553	0.0350

Fuente: elaboración propia.

Tabla 7. Resultados obtenidos con el algoritmo de Máquinas de soporte vectorial

cost	gamma	kernel	Sn	Sp	Exact.
1.000	0.042	lineal	0.1428	1.0000	0.7894
0.125	0.125	polinomial	0.2857	0.9534	0.7894
1.000	0.125	radial	0.2142	1.0000	0.8070
8.000	4.000	sigmoide	0.4444	0.8461	0.7192

Fuente: elaboración propia.

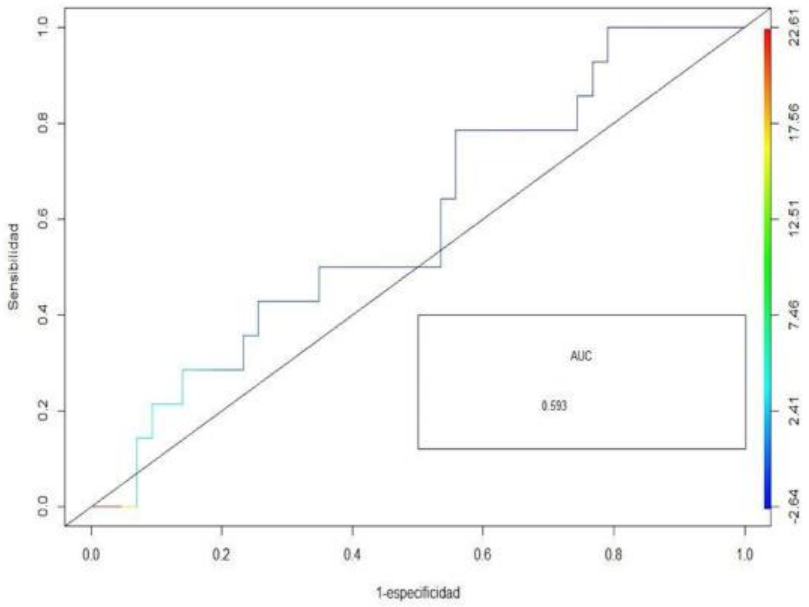

Figura 11. Análisis ROC del modelo usando Máquinas de soporte vectorial. Fuente: elaboración propia.

De acuerdo con los valores de la Tabla 7, se selecciona el modelo que utiliza los parámetros cost y gamma de 8 y 4, respectivamente, y la función kernel sigmoide. La exactitud del modelo es 71.92%. Al comparar la sensibilidad que fue de 44.44% con la especificidad que alcanzó el 84.61%, se puede determinar que el modelo detecta mejor los casos en los que el cliente no incumplirá el pago de la cuota. Este resultado supera el modelo de árboles de decisión con el algoritmo C4.5 y es superado por Random Forest, C5.0 y las redes neuronales.

El área bajo la curva obtenida por el modelo de máquinas de soporte vectorial es de 59.3%. Este resultado es el más bajo en toda la experimentación e indica que el modelo no tiene una buena capacidad discriminatoria de clientes que incumplen el pago de su cuota y de los que no incumplen. La Figura 11 muestra la curva ROC del modelo de máquinas de soporte vectorial con su parámetro AUC.

3.5. Análisis de resultados

El consolidado de las pruebas realizadas se muestra en la Tabla 8. El modelo que se obtiene con el algoritmo de árboles de decisión Random Forest resulta ser más acertado que los demás modelos. Este algoritmo obtuvo los valores más altos de especificidad (93.02%), exactitud (84.21), precisión (72.72%), y área bajo la curva (88.29%). Por su parte, la mayor sensibilidad la obtuvo el modelo de redes neuronales con un 71.83%.

Los valores obtenidos en esta investigación son comparables y en algunos casos superan la exactitud reportada en trabajos similares del estado del arte. Por ejemplo, en [1] se utiliza un total de 23 atributos y algoritmos de ensamble como AdaBoost, Voting y RandomTreesEmbedding, alcanzando una exactitud máxima de 81% y una especificidad de 37%.

Los modelos propuestos en este trabajo con algoritmos como Random Forest y C4.5 superan la exactitud del trabajo de [1] y además presentan una especificidad mucho más alta, lo cual resulta importante para este problema ya que no solo se logra identificar un alto porcentaje de los ciertos positivos y negativos, sino que también se logra mantener una alta especificidad, lo cual significa que los modelos pueden identificar un gran porcentaje de clientes que no incumplirán el pago.

En el caso del algoritmo Random Forest, alcanza una exactitud de 84.21% y una especificidad de 93.02%. Mientras que el modelo obtenido con el algoritmo C4.5 presenta una exactitud de 82.45% y una especificidad de 88.37%. Además, en [7] se reporta una exactitud del 81% usando redes neuronales y un total de 13 atributos. La exactitud de dos de los cinco modelos presentados en la Tabla 8 alcanzan una exactitud mayor a los resultados reportados en [7].

La exactitud mide la fracción de predicciones correctas y considera tanto a los verdaderos positivos (clientes que incumplen) como a los verdaderos negativos (clientes que no incumplen). Por esta razón, se convierte en una medida que tiene en cuenta ambas predicciones resaltando la capacidad de los modelos propuestos para no enfocarse en la clasificación de solamente uno de los dos tipos de predicciones, sino que logra mantener un balance entre las dos.

Para permitir que los cinco modelos de la Tabla 8 estuvieran disponibles para la gestión de riesgo crediticio se desarrolló un prototipo web. El prototipo se muestra en la Figura 12 y tiene, entre otras tareas, un módulo de gestión crediticia que cuenta con dos formularios llamados Gestión y Gestión masiva. El formulario Gestión permite, a partir de los 24 atributos descritos en la Tabla 1, predecir si un cliente incumplirá, o no, en su siguiente cuota, seleccionando y aplicando cualquiera de los cinco modelos propuestos en esta investigación y presentados en la Tabla 8.

Tabla 8. Consolidado de resultados

Algoritmo	Sn	Sp	Exact.	Precisión	AUC
Red neuronal	0.7183	0.6834	0.6362	0.4358	0.6991
C4.5	0.6428	0.8837	0.8245	0.6428	0.7824
Random Forest	0.5714	0.9302	0.8421	0.7272	0.8829
C5.0	0.6428	0.4651	0.6315	0.2812	0.8056
SVM	0.4444	0.8461	0.7192	0.5714	0.5930

Fuente: elaboración propia.

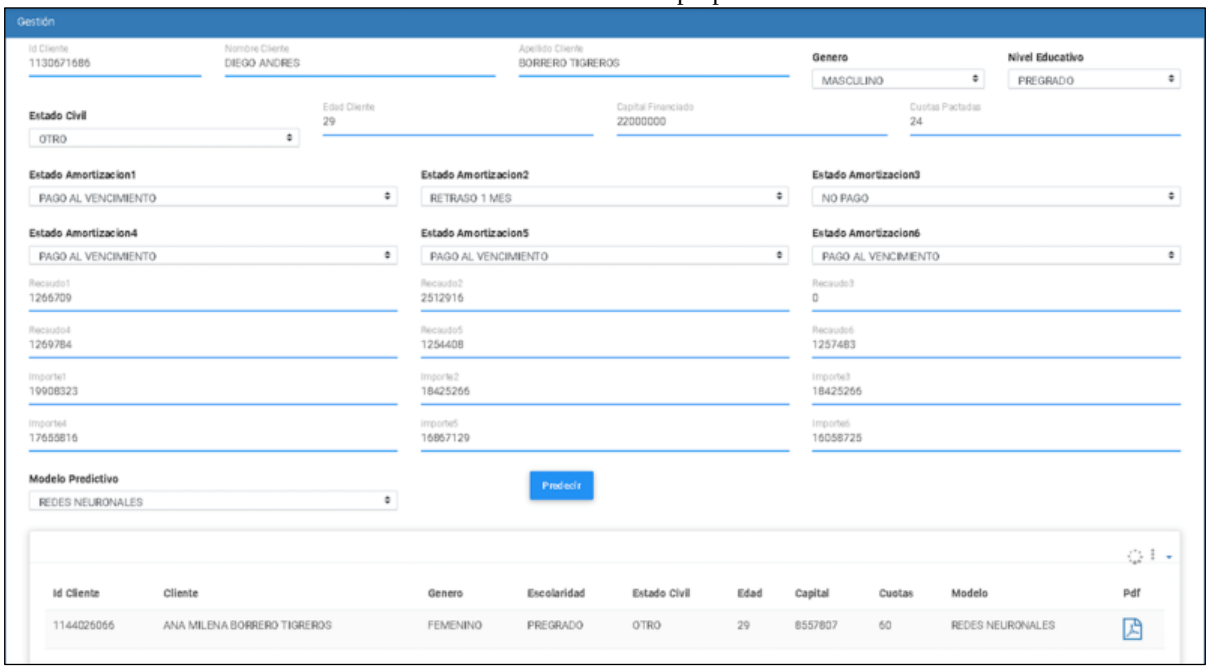

Figura 11. Análisis ROC del modelo usando Máquinas de soporte vectorial. Fuente: elaboración propia.

La predicción queda almacenada en una base de datos y puede ser consultada por parámetros como fecha de generación, id del cliente y modelo predictivo. El formulario Gestión masiva permite realizar predicciones a un conjunto masivo de clientes cuyos datos son cargados en una plantilla en formato CSV. Esta opción se utiliza para hacer la predicción de grandes cantidades de clientes.

4. Conclusiones

En este artículo se abordó el problema del riesgo crediticio en entidades financieras, específicamente lo relacionado con el análisis predictivo de deudores que pueden incurrir en mora. Este análisis es crucial para las entidades financieras quienes necesitan herramientas que les ayuden a tomar decisiones sobre sus clientes.

En esta investigación, cada cliente se representa con un conjunto de atributos que describen sus datos básicos, operaciones y pagos generados. Se proponen modelos que permiten predecir si un cliente incumplirá, o no, en el pago de su siguiente cuota. Los modelos se basan en algoritmos de inteligencia artificial. Específicamente se utilizan técnicas basadas en redes neuronales, árboles de decisión, y máquinas de soporte vectorial. Para evaluar cada modelo se calcularon medidas como la sensibilidad, especificidad, exactitud, tasa de error, precisión, precisión negativa, y área bajo la curva.

De acuerdo con los resultados obtenidos, los algoritmos Random Forest, C5.0, C4.5, y redes neuronales alcanzan áreas bajo la curva de 88.29%, 80.56%, 78.24%, y 69.91%, respectivamente. Por otra parte, el modelo obtenido con el algoritmo SVM no presentó el mismo comportamiento y su área bajo la curva es de 59.30%. Los modelos obtenidos alcanzan exactitudes similares y en algunos casos superan algunos de los trabajos del estado del arte.

Finalmente, se presenta un prototipo web que podrá ser usado por las personas encargadas de hacer el estudio de los clientes en las entidades financieras en Colombia para la toma de decisiones apoyados en los modelos propuestos en este artículo. El prototipo permite tener información predictiva sobre el posible comportamiento de los clientes y así facilitará la toma de medidas preventivas antes de que ocurran retrasos en los pagos. Este es un primer trabajo exploratorio con datos de entidades financieras en Colombia. Es importante destacar que se puede continuar con la exploración de otras técnicas que pudiesen mejorar aún más los resultados obtenidos. Por ejemplo, estrategias como redes neuronales profundas, redes Bayesianas, meta clasificadores, y sistemas difusos.

Referencias

[1] R. E. Turkson, E. Y. Baagyere y G. E. Wenya, "A machine learning approach for predicting bank credit worthiness", *en Third International Conference on Artificial Intelligence and Pattern Recognition (AIPR)*, Lodz, 2016, pp. 1-7, doi: 10.1109/ICAIPR.2016.7585216

[2] A. Gahlaut, Tushar, P. K. Singh, "Prediction analysis of risky credit using Data mining classification models", *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Delhi, 2017, pp. 1-7, doi: 10.1109/ICCCNT.2017.8203982

[3] S. V. Kulkarni, S. N. Dhage, "Advanced credit score calculation using social media and machine learning", *J. Intell. Fuzzy Syst.*, vol. 36, no. 3, pp. 2373-2380, 2019, doi: 10.3233/JIFS-169948.V

[4] H. R. Sanabila, W. Jatmiko, "Ensemble Learning on Large Scale Financial Imbalanced Data", *International Workshop on Big Data and Information Security (IWBIS)*, Jakarta, 2018, pp. 93-98, doi: 10.1109/IWBIS.2018.8471702

[5] J. Chaisuwan, N. Chumuang, "Intelligent Credit Service Risk Predicting System Based on Customer's Behavior By Using Machine Learning", *14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, Chiang Mai, Thailand, 2019, pp. 1-6, doi: 10.1109/iSAI-NLP48611.2019.9045452

[6] A. Al-qerem, G. Al-Naymat, M. Alhasan, "Loan Default Prediction Model Improvement through Comprehensive Preprocessing and Features Selection", *International Arab Conference on Information Technology (ACIT)*, Al Ain, United Arab Emirates, 2019, pp. 235-240, doi: 10.1109/ACIT47987.2019.8991084

[7] J. Bae, S. Lee, H. Seo, "Artificial Intelligence Techniques for Predicting Online Peer-to-Peer (P2P) Loan Default", *The Journal of Society for e-Business Studies*, vol. 23, no. 3, pp. 207-224, 2018.

[8] Y. Li, "Credit Risk Prediction Based on Machine Learning Methods", *14th International Conference on Computer Science & Education (ICCSE)*, Toronto, ON, Canada, 2019, pp. 1011-1013, doi: 10.1109/ICCSE.2019.8845444

[9] Z. Ereiz, "Predicting Default Loans Using Machine Learning (OptiML)", *27th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2019, pp. 1-4, doi: 10.1109/TELFOR48224.2019.8971110

[10] M. Malekipirbazari, V. Aksakalli, "Risk assessment in social lending via random forests", *Expert Systems with Applications*, vol. 42, no. 10, pp. 4621-4631, 2015, doi: 10.1016/j.eswa.2015.02.001

[11] A. Byanjankar, M. Heikkilä, J. Mezei, "Predicting Credit Risk in Peer-to-Peer Lending: A Neural Network Approach", *IEEE Symposium Series on Computational Intelligence*, Cape Town, 2015, pp. 719-725, doi: 10.1109/SSCI.2015.109

[12] J.H. Aboobyda, M.A. Tarig, "Developing Prediction Model Of Loan Risk In Banks Using Data Mining", *Machine Learning and Applications: An International Journal (MLAIJ)*, vol. 3, no. 1, pp. 1-9, 2016.

[13] J. Yan *et al.*, "Mining social lending motivations for loan project recommendations", *Expert Syst. Appl.*, vol. 111, pp. 100-106, 2018, doi: 10.1016/j.eswa.2017.11.010

[14] M. Sudhakar, C. Reddy, "Two Step Credit Risk Assessment Model For Retail Bank Loan Applications Using Decision Tree Data Mining Technique", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 5, no. 3, pp. 705-718, 2016.

[15] S. Eletter, S. Yaseen, G. Elrefae, "Neuro-based artificial intelligence model for loan decisions", *American Journal of Economics and Business Administration*, vol, 2, no 1, pp. 27-34, 2010.

[16] A. Blanco, R. Mejias, J. Lara, S. Rayo, "Credit scoring models for the microfinance industry using neural networks: evidence from Peru", *Expert Systems with Applications*, vol. 40, no. 1, pp. 356-364, 2013, doi: 10.1016/j.eswa.2012.07.051

[17] R. G. Lopes, R. N. Carvalho, M. Ladeira, R. S. Carvalho, "Predicting Recovery of Credit Operations on a Brazilian Bank", *15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Anaheim, CA, 2016, pp. 780-784, doi: 10.1109/ICMLA.2016.0139

[18] A. Khashman, "Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes", *Expert Syst. Appl.*, vol. 37, no. 9, pp. 6233-6239, 2010, doi: 10.1016/j.eswa.2010.02.101

[19] C. García. "Deudores bancarios empiezan a colgarse con sus obligaciones", 2016. [En línea]. Disponible en: http://www.eltiempo.com/economia/sectores/deudas-sin-pagar-crecen-en-colombia-42539

[20] R. Flórez, J. Fernández, *Las redes neuronales artificiales: Fundamentos teóricos y aplicaciones prácticas*. La Coruña, España: Editorial Netbiblo, 2008.

[21] R. Lippmann, "An introduction to computing with neural nets", en *IEEE ASSP Magazine*, vol. 4, no. 2, pp. 4-22, 1987, doi: 10.1109/MASSP.1987.1165576

[22] R. Hecht-Nielsen, *Neurocomputing*, Michigan, Estados Unidos: Addison-Wesley, 1990.