GRADO EN ECONOMÍA

TRABAJO FIN DE GRADO

De la Econometría clásica a los modelos de *Machine Learning*: un enfoque práctico de predicción en Economía

---

0018 Métodos Econométricos

24 de Mayo de 2020

Autor: Javier Carrasco Serrano

Profesor/es: Pedro Antonio Pérez Pascual

Contenido

1. Resumen / Abstract

Resumen

Este proyecto pretende abordar un problema de predicción de un evento binario, los impagos bancarios, mediante las técnicas econométricas vigentes en la industria: el modelo *logit* y las variables WOE. Para ello, se utilizarán datos de préstamos de la plataforma Lending Club, con el objetivo de mejorar, en términos de poder discriminante, el modelo que utilizan para evaluar el riesgo de sus operaciones. Por otro lado, y dada la popularidad que están ganando incluso en un sector tan regulatorio como el financiero, se van a construir modelos con técnicas de *machine learning* para ser comparados con el modelo de *scoring*.

Abstract

The aim of this working paper is to face a binary variable prediction problem, the defaults in banking loans, through the best practices in this industry: the logit model and WOE variables. For that purpose, loans information from Lending Club is used, trying to construct, in terms of discriminatory power, a model that improves the one used to assess the risk of their operations. Besides that, given their growing popularity, even considering the strong banking regulatory environment, additional models are built using machine learning techniques to be compared with the scoring model.

Keywords: scoring model, scorecard, logistic regression, logit model, machine learning.
JEL codes: C51, C52, C53.

2. Introducción

Objetivo

Lending Club[1] es una plataforma estadounidense de préstamos entre particulares, en la que unos usuarios solicitan préstamos, y otros (y/o la propia compañía) les conceden esos préstamos. La compañía maneja dos variables para diferenciar el riesgo de impago entre sus clientes: la primera de ellas, *grade*, consta de 7 categorías de riesgo; mientras que la variable *sub_grade* tiene 5 subgrupos para cada una de esas categorías. Así, un usuario que vaya a conceder un préstamo a otro usuario, tendrá como referencia una calificación (de entre 35) otorgada a esa operación en función de su riesgo crediticio. Además, Lending Club publica información relativa a las operaciones concedidas y denegadas, para que los prestamistas dispongan de más información de cara a tomar decisiones crediticias.

El objetivo de este proyecto es, a partir de la información que Lending Club publica de sus operaciones, generar una variable *score* que tenga más capacidad discriminante que la variable *sub_grade*. Es decir, construir un modelo que permita diferenciar el riesgo mejor que el modelo que aparentemente está usando la plataforma, o al menos el que está a disposición de sus usuarios. En general, a partir de un modelo que es capaz de ordenar mejor las operaciones en función del riesgo, es posible, por ejemplo, mantener la tasa de concesión de operaciones reduciendo la tasa de operaciones impagadas; mantener la tasa de impagos pero aumentando las operaciones concedidas; y mejorar la política de precios basados en riesgo.

Para ello, se va a construir un modelo típico de *scoring*, que consiste en una regresión logística (modelo *logit*) contra transformaciones WOE de algunas de las variables proporcionadas. A lo largo de este trabajo, se va a detallar cada paso desde la extracción de los datos hasta la generación del modelo final. Además, dada la popularidad que están ganando, se van a construir algunos ejemplos de modelos utilizando técnicas de *machine learning*, para ver si también es posible mejorar el modelo en vigor, e incluso a la regresión logística. Para el desarrollo del proyecto se hace uso del lenguaje R a través del software RStudio.

Cabe señalar que, dada la situación macroeconómica que se está viviendo, la demanda de créditos se podría ver significativamente afectada, así como el perfil de riesgo de los prestatarios, por lo que serían necesarios contrastes adicionales para garantizar que se puede implantar un modelo como el que se va a desarrollar. En este sentido, por ejemplo, la Autoridad Bancaria Europea, o EBA[2], ha indicado que:

> *"especially in difficult economic circumstances, it is particularly important to ensure that risk is identified and measured in a true and accurate manner. Institutions must therefore continue to adequately identify those situations where*

[1] Ver https://www.lendingclub.com/

[2] Para más detalle, ver EBA (2020).

short-term payment challenges may transpose into long-term financial difficulties and eventually lead to insolvency".

Revisión de la literatura

El origen del modelo *logit* se atribuye a Berkson[3] y la publicación del artículo *Application of the Logistic Function to Bio-Assay* en 1944, aunque la introducción de la función logística data del siglo anterior y se atribuye a trabajos de Verhulst y Quetelet (Cramer (2004)).

En cuanto al uso de este tipo de modelos en la industria bancaria, FICO[4], fundada en 1956 como Fair Isaac Corporation, fue la primera empresa, en 1958, en comercializar los modelos *credit scoring*. De hecho, a día de hoy el *score* de FICO es una variable que se utiliza para la concesión de créditos en muchos países, ya sea directamente como modelo o como variable formando parte de un modelo que contiene más información. La empresa estima que sobre el 75% de las concesiones de hipotecas utilizan directa o indirectamente su *score*.

Respecto a la metodología utilizada, dado que el principal software estadístico para la construcción e implementación de estos modelos es SAS, uno de los manuales más populares de la metodología construcción de los modelos de *scoring* proviene de dicha compañía: en 2005 Siddiqi publicó su *Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring*.

Por ejemplo, en Altman y Saunders (1998) y Thomas (2000) se pueden encontrar dos revisiones de la literatura respecto a la construcción y uso de modelos de *scoring* en la industria financiera.

Por último, cabe destacar la popularidad[5] que están adquiriendo las técnicas de *machine learning* en los problemas de predicción de eventos binarios, como es el caso de los impagos bancarios. Parte de esta popularidad se debe al aumento en la capacidad de almacenamiento y procesamiento en los datos. Además, competiciones como las planteadas por Kaggle[6], adquirida por Google en 2017, promueven la investigación de nuevos algoritmos en problemas de clasificación. Los dos manuales de Tibshirani sirven de referencia para este tipo de técnicas: *An Introduction to Statistical Learning,* y *The Elements of Statistical Learning*.

A nivel regulatorio, el uso de este tipo de técnicas en la concesión de préstamos supone un reto que se viene discutiendo en los últimos años y que se está introduciendo poco a poco en la industria, sino directamente en la construcción de los modelos, sí dando

[3] Para más detalle, ver Berkson (1944).

[4] Para más detalle, ver https://www.fico.com/about-us#history

[5] Ver, por ejemplo, Gambacorta, Huang, Qiu y Wang (2019).

[6] Ver https://www.kaggle.com/competitions

soporte a parte de los tradicionales modelos de *scoring*, por ejemplo, en la generación de nuevas variables, como señalaba recientemente el Banco de España[7]:

> "*Una opción relativamente extendida y que ofrece en la mayoría de los casos un razonable equilibrio entre beneficios y riesgos es utilizar las herramientas de inteligencia artificial de manera complementaria (no sustitutiva) a las técnicas tradicionales. Es decir, no se trata de aceptar sin más los resultados que proporcionan los algoritmos de un modo automático, sino de utilizarlos como parte de un proceso de reforzamiento y de validación de las decisiones*".

Conclusiones

El modelo de *scoring* construido mejora considerablemente en poder discriminante a la variable *sub_grade* que utiliza Lending Club y, sobre todo, que es la única referencia para inversores externos (posiblemente Lending Club disponga de un modelo más granular similar al que se construye en este trabajo, pero sólo hace públicas las calificaciones agregadas en esas 35 categorías). La mejora respecto al modelo en vigor es de un 40% relativo en la muestra *test* y un 36% relativo en la muestra OOT.

Además, aunque no se ha hecho una exploración intensiva en sus parametrizaciones, algunos de los modelos que utilizan técnicas de *machine learning* alcanzan niveles de discriminación similares e incluso mejores que la regresión logística, aunque es más complicado entender su funcionamiento y resultado. Si se pudieran explorar nuevas fuentes de datos, o tratar datos que han tenido que ser descartados por su formato (por ejemplo, variables de texto libre), se esperaría que el uso de estas técnicas diera incluso mejores resultados.

	Poder discriminante			Indicador temprano de poder discriminante
Modelo	**Train**	**Test**	**OOT**	**TTD**
Modelo vigente (variable WOE subgrade)	27,70%	27,38%	27,99%	32,22%
Regresión logística, 11 variables WOE	**39,11%**	**38,27%**	**38,06%**	**39,19%**
Árbol de regresión, variables crudas	39,06%	-	-	-
Árbol de regresión, variables WOE	39,13%	37,25%	36,88%	37,66%
Random forest, variables crudas	99,95%	-	-	-
Random forest, variables WOE	99,97%	35,55%	35,73%	36,27%
Random forest, variables WOE, máx 2.000 nodos	54,30%	25,84%	24,71%	24,32%
Random forest, variables WOE, mín 50 observs.	61,37%	30,59%	29,95%	30,11%
Árboles adaboost, variables crudas	42,13%	39,11%	33,92%	33,22%
Árboles adaboost, variables WOE	41,42%	39,03%	39,03%	39,43%
XGBoost, variables WOE	41,23%	39,22%	39,25%	39,73%

Tabla 1: Resultados de los principales modelos construidos y del modelo en vigor

[7] Para más detalle, ver Fernández, A. (2019).

3. Desarrollo

3.1 Extracción y carga de datos

Los datos con los que se ha trabajado fueron extraídos de la web de Lending Club (https://www.lendingclub.com/info/download-data.action) en Abril de 2019, aunque estaban actualizados a Diciembre de 2018. En el momento de la extracción, los datos estaban abiertos para quien quisiera acceder a ellos, pero actualmente es necesario registrarse en la plataforma. Los datos no se adjuntan debido a su tamaño (235Mb una vez comprimidos), pero serán puestos a disposición del lector si así se solicita.

Los datos, y por lo tanto los ficheros (en formato .csv), vienen agrupados de manera trimestral, recogiendo información sobre las operaciones concedidas en ese trimestre, y también información respecto al comportamiento de esas operaciones hasta un año después (por ejemplo, si hay un impago). Además, se dispone de un diccionario de datos que explica cada una de las 143 variables, que se adjunta en el Anexo 1. Los ficheros descargados contienen información desde 2016Q1 hasta 2018Q4, de manera trimestral.

Así, al haber sido descargados con información hasta 2018Q4, los datos de 2016 y 2017 contienen información completa de desempeño (estado del pago del préstamo), ya que lo miden hasta un año después, pero no así los de 2018. La volumetría de esas ventanas viene recogida en la siguiente tabla:

Ventana	Observaciones
2016Q1	133.887
2016Q2	97.854
2016Q3	99.120
2016Q4	103.546
2017Q1	96.779
2017Q2	105.451
2017Q3	122.701
2017Q4	118.648
2018Q1	107.864
2018Q2	130.772
2018Q3	128.194
2018Q4	128.412

Tabla 2: Operaciones aceptadas por ventana

Es posible también descargar información sobre las operaciones denegadas, aunque no tenía sentido incorporar esta información a los datos con los que se ha trabajado debido a que no almacenan la misma información que para las operaciones concedidas (9 frente a 143 variables), imposibilitando por tanto la generación de un modelo *dirty* para inferir un comportamiento en estas operaciones, e incluirlas[8] así a las muestras de desarrollo para construir el modelo definitivo. Este hecho también imposibilita ejercicios

[8] Por ejemplo, utilizando las metodologías *fuzzy* o *parcelling* descritas en Anderson, Haller y Siddiqi (2009).

interesantes como, una vez construido el modelo con el que se va a evaluar a la población, estudiar mover el punto de corte de admisión en función de la tasa de denegados y tasa de malos esperadas para cada punto, intentando así, si se dispone de alguna métrica de rentabilidad o beneficios, elegir a los clientes que se admiten maximizando dicha métrica. A nivel informativo, la volumetría de operaciones denegadas es la siguiente:

Ventana	Observaciones
2016Q1	1.096.204
2016Q2	996.561
2016Q3	1.272.619
2016Q4	1.404.490
2017Q1	1.379.756
2017Q2	1.665.309
2017Q3	2.007.021
2017Q4	2.020.487
2018Q1	1.875.134
2018Q2	2.441.981
2018Q3	2.585.245
2018Q4	2.594.422

Tabla 3: Operaciones denegadas por ventana

Es importante remarcar la fecha de extracción debido a que, si se hace una extracción actualizada a otro trimestre, no sólo se dispone de ficheros con nuevas operaciones, sino que algunas de las variables de los ficheros anteriores pueden cambiar su valor porque ahora se dispone de más tiempo para observar si una operación ha sido impagada o no.

Para poder cargar los datos de los ficheros .csv al proyecto de RStudio[9] ha sido necesario un tratamiento previo, que viene descrito en el código, y que incluye:

- Generar el id (identificador) de la operación.
- Eliminar la primera y las últimas dos filas de cada fichero (descripción de la ventana de datos y publicidad).
- Eliminar las dobles comillas ("…") que encerraban cada registro.
- Corregir saltos de fila erróneos (un registro se partía en dos registros) que ocurrían cuando el contenido de una variable de texto libre era demasiado largo y producía que, a partir de determinado registro, los valores de las variables estuvieran en las variables adyacentes, imposibilitando trabajar con buena parte de los datos.

3.2 Definiciones de target y muestras

Una vez se tienen los datos con los que se va a trabajar, es necesario definir la variable target o el evento que se quiere modelizar, y que será por tanto la variable dependiente del modelo que se construya.

[9] Referido a lo largo de este documento como "el proyecto" o "el código".

Por el tipo de modelos que se van a utilizar, es necesario que esta variable sea binaria y sólo pueda tomar dos valores, 0 y 1. Se marcarán con 1 los registros que presenten el evento, y se les conocerá como observaciones "malas" o directamente como "malos", mientras que los registros que no presenten el evento se conocerán como "buenos" y se marcarán con un 0.

La variable *loan_status* contiene información de impagos hasta un año después de la fecha en la que se admite la operación, y toma los siguientes posibles valores: current (operaciones que no han presentado impagos), in grace period (operaciones que han presentado impagos de 0 a 15 días), late 16 - 30 (operaciones que han presentado impagos de 16 a 30 días), late 31 - 120 (operaciones que han presentado impagos de 31 a 120 días), default (operaciones que han presentado impagos de más de 120 días) y charged-off (operaciones fallidas, en las que se considera que el cliente ya no va a volver a pagar y no se va a recuperar el préstamo; no necesariamente tienen que ser a partir de 120 días de impago, sino que es el momento en el que a nivel contable se imputa el importe pendiente como pérdidas). El término "default" suele estar asociado al uso y supervisión regulatoria de este tipo de modelos cuando se aplican en Entidades de Crédito (que también reciben depósitos de clientes y por tanto están sujetas a una supervisión mucho mayor que una plataforma que actúa de intermediario entre clientes o que concede créditos). De hecho, este término, que en ocasiones en la literatura se utiliza de manera genérica como "malo" una vez se han definido las dos clases de la variable target, suele indicar impagos de más de 90 días y algunos marcajes subjetivos[10].

Para definir adecuadamente la variable target, es necesario agrupar los distintos valores de la variable *loan_status* en dos categorías que se correspondan a observaciones malas y buenas. Obviamente, las malas deberían recoger los valores que acaben suponiendo pérdidas para la cartera o los prestamistas. La distribución de esta variable a lo largo de las distintas ventanas temporales es la siguiente:

	2016Q1	2016Q2	2016Q3	2016Q4	2017Q1	2017Q2	2017Q3	2017Q4	2018Q1	2018Q2	2018Q3	2018Q4
Charge-off	17,70%	17,10%	17,00%	14,50%	12,50%	11,50%	10,30%	8,00%	5,90%	4,30%	2,10%	0,80%
In grace	0,20%	0,30%	0,50%	0,50%	0,60%	0,70%	0,80%	0,80%	0,70%	0,80%	0,80%	0,60%
16-30	0,10%	0,10%	0,20%	0,20%	0,20%	0,20%	0,30%	0,40%	0,40%	0,40%	0,30%	0,30%
31-120	0,40%	0,60%	0,90%	1,20%	1,30%	1,50%	1,70%	1,80%	1,90%	2,10%	1,80%	1,50%
Default	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Current	81,60%	81,90%	81,40%	83,60%	85,40%	86,10%	86,90%	89,00%	91,10%	92,40%	95,00%	96,80%

Tabla 4: Valores *loan_status* por ventana

Se observa cierta mejora en la cartera en términos de tasas de malos de 2016 a 2017, que se podría atribuir a una mejora del comportamiento de crédito general de la población (mejor situación económica en EEUU) o a una mejora en la admisión de clientes (mejora de los modelos de admisión). La aparente mejora que se ve en 2018 no es tal, sino que es debida a que son operaciones que no permiten medir el desempeño completo de acuerdo a la definición que utilizan, dado que todavía no ha pasado un año desde la admisión de las operaciones.

[10] Para más detalles sobre la definición regulatoria de default, ver EBA (2016).

Por otro lado, aunque Lending Club no haga pública la definición de malo que utiliza para construir su modelo de admisión (en principio, recogido o relacionado con la variable *sub_grade*), sí que aporta datos de migraciones entre categorías de *loan_status*: sobre el 37% de las operaciones que tienen retrasos menores a 30 días, acaban siendo fallidas, mientras que en torno al 75% de las que tienen retrasos superiores a 30 días, acaban en fallido. Este hecho, junto con cierta estabilidad de esta definición en las ventanas de 2016 y 2017 (el cálculo del PSI, estadístico que se explicará en detalle en la sección 3.7 sobre contrastes en el modelo, arroja un resultado por debajo del 6% al comparar la muestra de 2016Q1 con la de 2017Q4, muy por debajo de umbrales que indicarían que la población es inestable) que se puede apreciar en la siguiente tabla, lleva a definir como buenas (0) aquellas operaciones que tienen menos de 31 días de retraso (valores current, in grace period, late 16 – 30), y como malas (1) las que superan este umbral (late 31 - 120, default, charged-off).

	2016Q1	2016Q2	2016Q3	2016Q4	2017Q1	2017Q2	2017Q3	2017Q4	2018Q1	2018Q2	2018Q3	2018Q4
Buenos	81,90%	82,30%	82,10%	84,30%	86,20%	87,00%	88,00%	90,20%	92,20%	93,60%	96,10%	97,70%
Malos	18,10%	17,70%	17,90%	15,70%	13,80%	13,00%	12,00%	9,80%	7,80%	6,40%	3,90%	2,30%

Tabla 5: Valores *target* por ventana

La definición de la variable *target*, y por tanto del evento que se va a modelizar, quedaría:

- Malos: operaciones que presentan o han presentado retrasos de al menos 31 días en algún pago **durante** el primer año de la operación (desde que se concede el préstamo). También aparecen marcadas como malas las operaciones que han presentado este tipo de impagos durante el año, aunque a los 12 meses del inicio estén al corriente de los pagos (se hayan recuperado, o tengan un plan de refinanciación o de carencia).
- Buenos: operaciones que no presentan este tipo de impagos durante el primer año de la operación. Es decir, si hubiera algún retraso, no excedería los 30 días.

Para definir las muestras o ventanas con las que se va a trabajar, es necesario preguntarse qué tipo de contrastes se quieren llevar a cabo sobre el modelo construido.

- La muestra de construcción se va a partir mediante muestreo aleatorio estratificado por la variable target en dos muestras, una muestra *train* que supondrá el 70%[11] de la población y sobre la que se entrenará el modelo, y una muestra *test* que contendrá el 30% restante y sobre la que se medirá el sobreajuste del modelo a la muestra *train*, comparando el poder discriminante[12] del modelo en ambas muestras. Así, la muestra de construcción será 2016Q1 - 2016Q4.

[11] Ver por ejemplo discusión en Cocea y Liu (2007). También es posible relajar el tamaño de la muestra *test* hasta un 80% - 20%, o incluso prescindir de ella, si no se dispone de suficientes registros en ambas clases, pero sí que se dispone de una muestra OOT suficientemente robusta.

[12] Al tratarse de modelos de clasificación en los que las dos categorías no están balanceadas, es más interesante fijarse en este tipo de estadísticos (KS, Gini, AUC) que en la matriz de confusión o estadísticos de determinación (R^2). En la discusión sobre contrastes del modelo se dará más detalle de este aspecto.

- Será necesario también evaluar la estabilidad temporal del poder discriminante del modelo, para lo que es necesario contar con otra ventana temporal diferente en la que también sea posible medir el desempeño y que sea la más reciente posible: es decir, que haya al menos un año entre la última observación y 2018Q4. Esta ventana suele conocerse como muestra de validación *out-of-time* o simplemente OOT, e incluirá de 2017Q1 a 2017Q4.

- Por último, también es necesario contrastar la estabilidad de la población que evaluaría el modelo. Para ello se utilizan las ventanas que no tienen completa la información de desempeño y que están más pegadas a un posible momento de aplicación o implementación del modelo. Esta muestra se conoce como muestra de validación *through-the-door* o simplemente TTD, y abarcará de 2018Q1 a 2018Q4.

Nótese que en los tres casos, las muestras comprenden un año de observaciones, lo que también permite anticiparse a posibles sesgos de estacionalidad que pudiera haber en los datos. La siguiente figura puede ayudar a entender la amplitud de estas ventanas, y, en el caso de las ventanas de construcción y OOT, dónde se observa el desempeño de las operaciones admitidas que contienen:

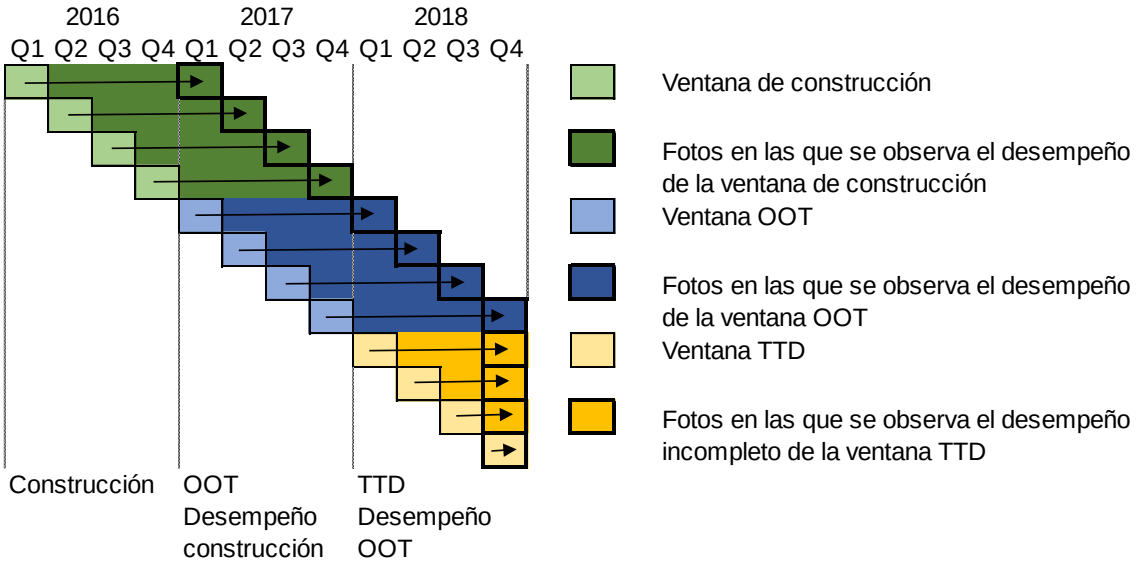

Figura 1: Ventanas de construcción, OOT y TTD

3.3 Tratamiento de datos y análisis descriptivos

En la creación de la variable *target* se observa que hay algunos registros que tienen esta variable a *missing* (valor sin informar). En otras variables, por la técnica que se va a utilizar para transformar las variables, es posible dejar valores *missing* o no hacer ninguna imputación sobre ellos (media, mediana, moda…), pero en el caso de la variable target contra la que se va a modelizar, al ser pocos registros, no tiene sentido dejarlos en las muestras, por lo que se han eliminado dichos registros. La volumetría es la siguiente:

Ventana	Observaciones	Target a missing
2016Q1	133.887	0
2016Q2	97.854	0
2016Q3	99.120	1
2016Q4	103.546	0
2017Q1	96.779	0
2017Q2	105.451	1
2017Q3	122.701	0
2017Q4	118.648	1
2018Q1	107.864	2
2018Q2	130.772	3
2018Q3	128.194	4
2018Q4	128.412	4

Tabla 6: Valores *target* sin informar

No se realiza ningún otro tipo de exclusión en los registros.

Una vez llegados a este punto, los análisis se realizan sobre todas las muestras en su conjunto, y las modificaciones se aplican en cada una de las muestras en caso de ser necesario (por ejemplo, cuando se transforma o elimina una variable). Es el turno de analizar cada una de las 143 variables que hay en la muestra. Por motivos de espacio, no se muestran los análisis realizados a cada una de las variables, aunque sí que se incluyen los más significativos o los de variables que tienen más "sentido de negocio" para estar dentro de un modelo de admisión de crédito, así como las transformaciones que se han llevado a cabo en las variables que lo han requerido.

En primer lugar, se analiza si las variables son continuas, texto o categóricas, y se modifica algunas para que recojan el tipo adecuado para esa variable: las variables *int_rate* y revol_util, recogían valores numéricos en porcentaje como si fueran textos, imposibilitando su tratamiento estadístico, por lo que se las ha cambiado de ese formato de texto con porcentaje (por ejemplo 5,13%) a numérico (0,0513). Las variables que contenían fechas (*issue_d, earliest_cr_line, last_pymnt_d, next_pymnt_d, last_credit_pull_d, sec_app_earliest_cr_line, payment_plan_start_date, debt_settlement_flag_date, settlement_date*) también se cargaban como textos y ha sido necesario una transformación para indicar que eran fechas y poder realizar un tratamiento más adecuado. Además, se ha generado la variable *aging_bureau*, que contiene los años que lleva el cliente en el bureau de crédito[13], como *issue_d - earliest_cr_line*, por si fuera de utilidad.

[13] El bureau estadounidense no es como el español. En nuestro caso, es un bureau negativo en el que sólo aparece información relativa a impagos con entidades financieras y otras empresas (suministros, telecos…), pero el americano también es un bureau positivo en el que aparece información positiva cuando el cliente ha estado al corriente en pagos de préstamos pasados. De hecho, es una práctica habitual en algunas familias poner un crédito de consumo a nombre de los hijos para que luego aparezcan en el bureau con registros positivos y tengan facilidades al solicitar ellos en el futuro un préstamo de vivienda. De ahí el sentido de crear esta variable, aunque luego no acabe entrando en el modelo final.

Las siguientes variables se han eliminado por tener todos sus registros vacíos: *member_id*, *url*. Dicha casuística se puede asociar en algún caso a motivos legales que garantizan que no se viola la privacidad de los clientes (identificador de cliente o dirección electrónica desde la que acceden).

También se han eliminado las siguientes variables por considerar que tienen información que no estaría disponible en el momento de evaluación de una operación, y por lo tanto no son susceptibles de ser incluidas en un modelo de admisión: *installment* (meses que todavía se deben), *pymnt_plan* (plan de refinanciación si tiene refinanciación), *out_prncp* (porcentaje por pagar), *out_prncp_inv* (porcentaje por pagar del préstamo puesto por inversores), *total_pymnt* (total pagado del préstamo), *total_pymnt_inv* (total pagado del préstamo de inversores), *total_rec_prncp* (total pagado del principal del préstamo), *total_rec_int* (total pagado de intereses), *total_rec_late_fee* (total de retrasos), *recoveries* (recuperaciones después de fallido), *collection_recovery_fee* (interés de recobro en caso de impagos), *hardship_flag* (indicador de plan de carencia), *hardship_type* (tipo de carencia), *hardship_reason* (razón de carencia), *hardship_status* (estado del plan de carencia), *deferral_term* (plazo de la carencia), *hardship_amount* (cantidad en plan de carencia), *hardship_start_date* (inicio de carencia), *hardship_end_date* (final de carencia), *payment_plan_start_date* (inicio de refinanciación), *hardship_length* (período de carencia), *hardship_dpd* (días de retraso antes de carencia), *last_pymnt_amnt* (último pago), *last_pymnt_d* (fecha del último pago), *hardship_loan_status* (estado de impago antes de carencia), *orig_projected_additional_accrued_interest* (intereses originales en planes de carencia), *hardship_payoff_balance_amount* (importe impagado en carencia), *hardship_last_payment_amount* (último pago si hay carencia), *debt_settlement_flag* (indicador de refinanciación), *debt_settlement_flag_date* (fecha de refinanciación), *settlement_status* (estado de refinanciación), *settlement_date* (fecha de refinanciación), *settlement_amount* (cantidad en refinanciación), *settlement_percentage* (porcentaje en refinanciación), *settlement_term* (plazo de refinanciación). Si alguna de estas variables fuera incluida en el modelo, podría derivar en un problema de estabilidad en la ventana TTD, que en realidad estaría anticipando una futura caída en el poder discriminante. Este tipo de variables sí que se deberían evaluar si se estuviera construyendo un modelo de comportamiento o de recuperaciones, pero no es el caso. Hay una variable que genera dudas en este sentido, el tipo de interés de las operaciones (*int_rate*), ya que con la descripción dada es difícil saber si es una variable que se tiene en el momento de evaluar la operación o se genera una vez aceptada la solicitud. Por el momento, se va a mantener entre las variables candidatas, aunque finalmente será descartada del modelo final por tener una alta correlación con otra de las variables incluidas (*grade*). También se ha eliminado la variable *loan_status* al estar ya contenida en la variable generada *target*.

Además, las técnicas que se van a utilizar no permiten el tratamiento de variables que incluyan texto libre. Es decir, es posible utilizar variables que son de texto pero que toman un número limitado de valores, pero no es posible utilizar variables en las que el cliente ha escrito texto libre y que podrían llegar a tomar tantos valores distintos como operaciones hay. Las variables que se han eliminado son *emp_title* (nombre del empleo, con 295.383 valores diferentes), *desc* (descripción de la finalidad del crédito; además, en este caso, la variable *purpose* recoge la finalidad del crédito en una lista predefinida

que presenta 14 posibles categorías). En cambio, la variable *emp_length* presentaba valores en texto de la forma “5 years”, que han sido transformados a valores numéricos (0 para los menores a 1, 1,…, 9, y 10 para los >9).

Por último, se elimina la variable *funded_amnt* por tomar el mismo valor que la variable *loan_amount* para todos los registros. En el caso de modelos de admisión en los que no se financia el 100% de la finalidad del crédito (hipotecas, préstamos vehículos) es interesante estudiar la variable *loan to value* (LTV), que mide el porcentaje que se está financiando. En este caso, o bien siempre se financia el 100%, o bien no se está recogiendo adecuadamente el valor del objeto por el que se pide el préstamo. Y, aunque no tomen exactamente los mismos valores, la variable *zip_code,* que contenía únicamente 2 de los 5 números del código postal, se ha eliminado por la limitación que tiene por motivos de privacidad, y porque la variable *addr_state* recoge el Estado en el que vive el cliente y por tanto tiene parte de la información del código postal.

A continuación se muestran algunas distribución de variables que se podrían considerar interesantes para entender la población con la que se está trabajando: *term* (plazo del préstamo), *sub_grade* (calificación que da a las operaciones Lending Club, se entiende que viene del modelo con el que están admitiendo operaciones), *grade* (igual que la variable anterior, pero menos granular, pues presenta 7 en lugar de 35 posibles categorías), *int_rate* (tipo de interés de la operación), *annual_income* (ingresos anuales), *purpose* (propósito). Para las variables continuas no aparece la distribución completa, sino algunos estadísticos de distribución.

Valor	Observaciones
36 months	1.067.007
60 months	421.879

Tabla 7: Distribución variable *term*

Valor	Observaciones	Valor	Observaciones	Valor	Observaciones	Valor	Observaciones	Valor	Observaciones
A1	71.158	B3	80.307	C5	80.334	E2	13.341	F4	2.717
A2	52.605	B4	93.011	D1	49.198	E3	13.092	F5	2.580
A3	55.250	B5	98.034	D2	47.987	E4	11.490	G1	2.242
A4	72.657	C1	98.711	D3	41.099	E5	13.804	G2	1.299
A5	70.209	C2	84.440	D4	33.366	F1	6.205	G3	1.114
B1	86.556	C3	84.660	D5	28.012	F2	3.920	G4	1.049
B2	84.218	C4	83.302	E1	16.567	F3	3.360	G5	992

Tabla 8: Distribución variable *sub_grade*

Valor	Observaciones
A	321.879
B	442.126
C	431.447
D	199.662
E	68.294
F	18.782
G	6.696

Tabla 9: Distribución variable *grade*

Estadístico	interest_rate	annual_income
Min	0,0531	0
Percentil 25	0,0943	47.000
Mediana	0,1199	67.000
Mediana	0,1297	80.309
Percentil 75	0,1577	96.000
Máximo	0,3099	110.000.000

Tabla 10: Estadísticos distribución variables *int_rate* y *anual_income*

Valor	Observaciones	Valor	Observaciones
car	16.173	medical	20.279
credit_card	342.521	moving	10.561
debt_consolidation	817.362	other	102.528
educational	1	renewable_energy	929
home_improvement	105.376	small_business	15.389
house	11.174	vacation	11.485
major_purchase	35.100	wedding	8

Tabla 11: Distribución variable *purpose*

Para finalizar el tratamiento de datos, es necesario señalar que no se hace ningún tipo de tratamiento sobre los valores *missing* ni sobre los *outliers,* debido a que la técnica de transformación de variables que se va a aplicar antes de incluirlas en los potenciales modelos (transformación a variables WOE) permite, aparte de capturar relaciones no lineales, agrupar esos valores *missing* o *outliers* con valores que presenten un comportamiento similar en términos de calidad crediticia, o incluso formar un grupo adicional si estuvieran describiendo un perfil de riesgo distinto (por ejemplo, si una variable que indica los ingresos del cliente está vacía, podría indicar que los ingresos son cero, y se esperaría un comportamiento peor que en el resto de población; por otro lado, si hay valores sin informar en una variable de tipo *bureau,* que recoge si el titular presenta impagos en otra entidad financiera, podría indicar que el cliente no tiene impagos, y por tanto formar un grupo con un mejor comportamiento en términos de capacidad de pago).

3.4 Análisis multivariante

Los tratamientos aplicados en el apartado anterior dejan un total de 99 variables candidatas, es decir, variables que podrían entrar a formar parte de nuestro modelo. De ellas, 85 son continuas, 11 son categóricas y 3 son fechas. Aunque se ha reducido el número, todavía son muchas para construir modelos que las incluyan, por lo que primero se va a estudiar la distribución cruzada con la variable target, para intentar buscar variables que ayuden a separar estas dos categorías (buenos y malos).

En este apartado no se elimina ninguna variable, ya que para eso se presentará un estadístico en la siguiente sección, pero a partir de las distribuciones cruzadas se puede empezar a intuir que hay algunas variables que son planas (tienen distribuciones similares para buenos y malos), y hay otras que sí que pueden acabar formando parte del modelo porque hay separación entre la distribución cruzada con buenos y malos. Es decir, es simplemente un análisis exploratorio para conocer mejor las variables con las que se trabaja.

A continuación se presentan diversos ejemplos de distribuciones cruzadas con la variable *target*. Por un lado, aparecen algunas de las variables del apartado anterior, que se han elegido de manera experta a priori por esperar, desde un punto de vista de negocio, que sean susceptibles de estar dentro de un modelo que distinga entre buenos y malos. Estas variables son: *term*, *sub_grade*, *grade*, *int_rate*, *annual_income_joint* (esta variable se ha sustituido por los ingresos anuales conjuntos del titular y avalistas, ya que se ve mejor cómo separa entre las dos clases). La variable *purpose* no aparece ya que al tener un mayor número de clases y que todavía no estén ordenadas en función del riesgo, es más difícil de graficar o de analizar en un espacio limitado. Además, se añade la variable *dti_joint* (ratio *debt to income*, es decir, deuda entre ingresos, del titular y avalistas), que es un buen ejemplo de separación entre clases para una variable continua, y las variables *total_acc* (número total de cuentas en el sistema bancario) e *initial_list_status* (lista de productos) para mostrar un ejemplo de variable continua y otro de categórica de variables que apenas separan entre las dos clases.

La manera de interpretar los gráficos es la siguiente:

- Para variables continuas, si discriminan entre las dos clases, la distribución al cruzarse con los malos debería estar desplazada respecto a la distribución al cruzarse con los buenos, hacia lo que serían valores de esa variable que presentan peores comportamiento de crédito. En el caso de variables que no separen bien entre buenos y malos, las distribuciones serán similares.
- Para variables discretas, hay que mirar, para cada una de las categorías, si se mantiene la distancia vertical entre los buenos y malos de cada categoría. Una variable que discrimine, debería presentar diferentes distancias verticales entre buenos y malos en cada categoría (categorías mejores, presentarán distancias mayores). En cambio, una variable que no discrimine, presentará distancias verticales entre buenos y malos similares para todas las categorías.

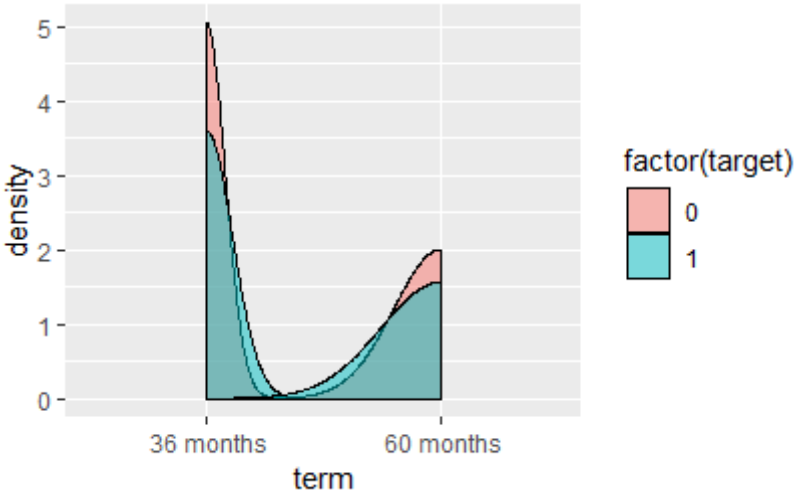

Figura 2: Distribución cruzada variable *term* con *target*

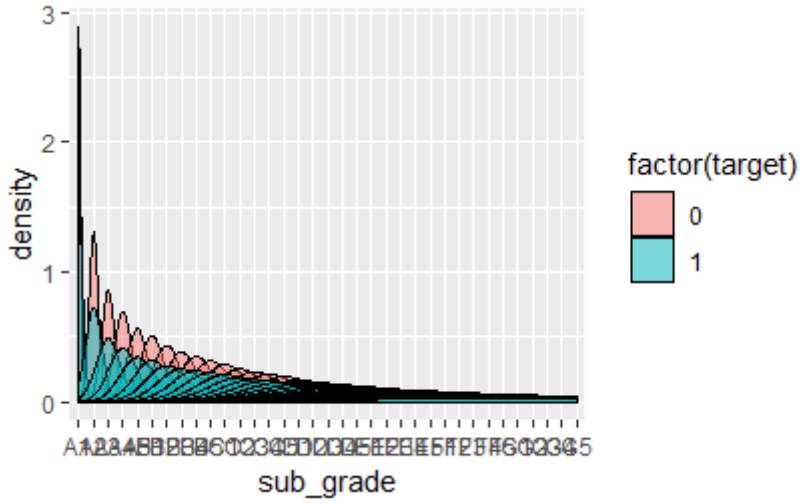

Figura 3: Distribución cruzada variable *sub_grade* con *target*

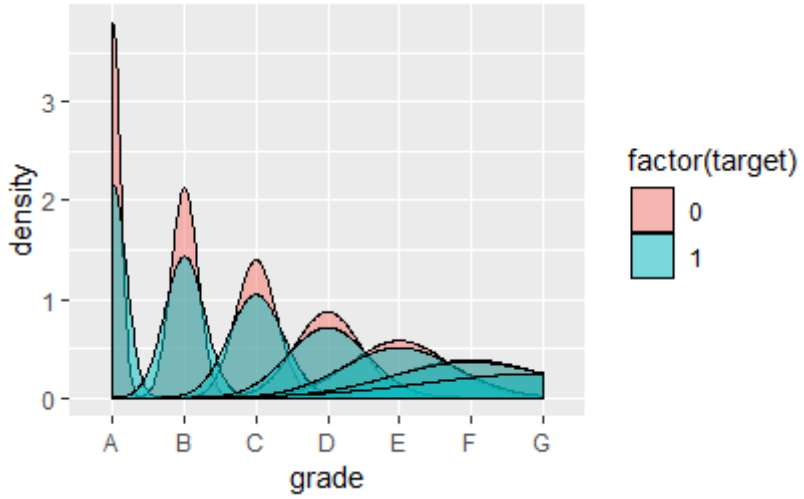

Figura 4: Distribución cruzada variable *grade* con *target*

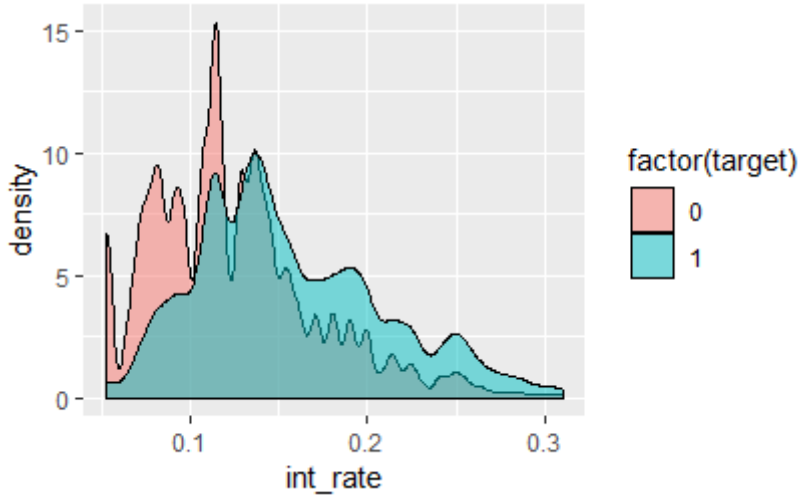

Figura 5: Distribución cruzada variable *int_rate* con *target*

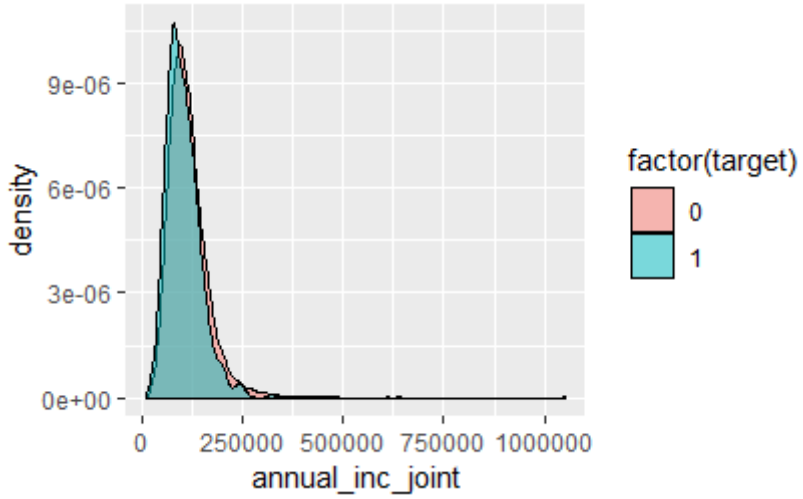

Figura 6: Distribución cruzada variable *anual_inc_joint* con *target*

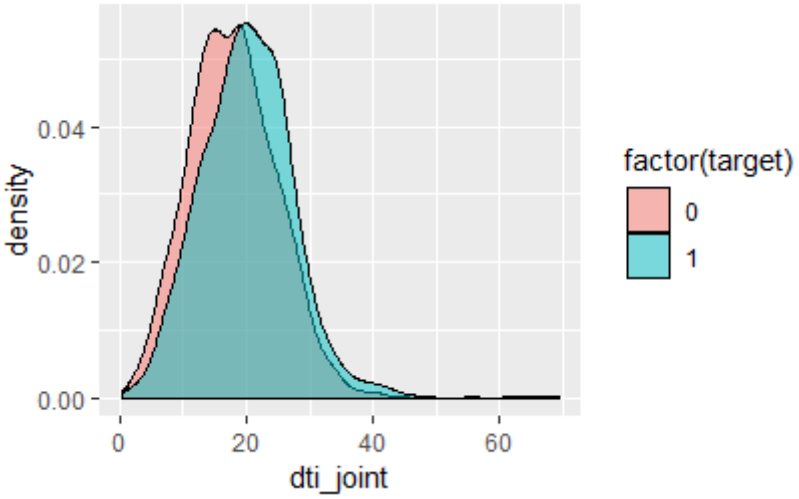

Figura 7: Distribución cruzada variable *dti_joint* con *target*

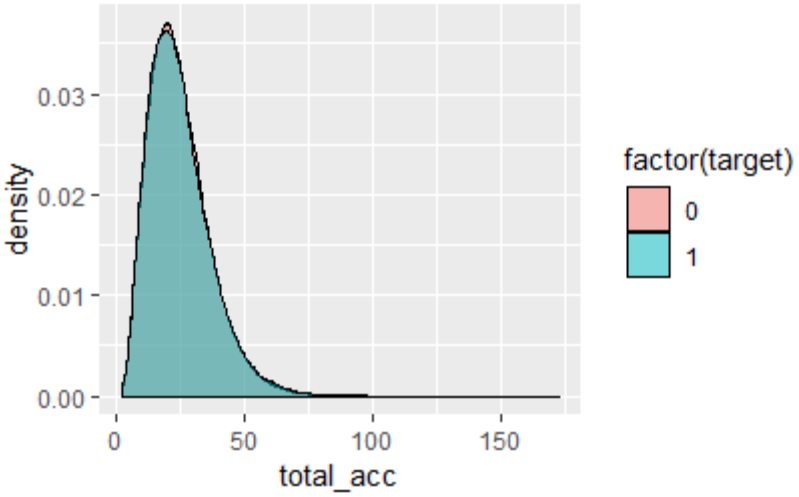

Figura 8: Distribución cruzada variable *total_acc* con *target*

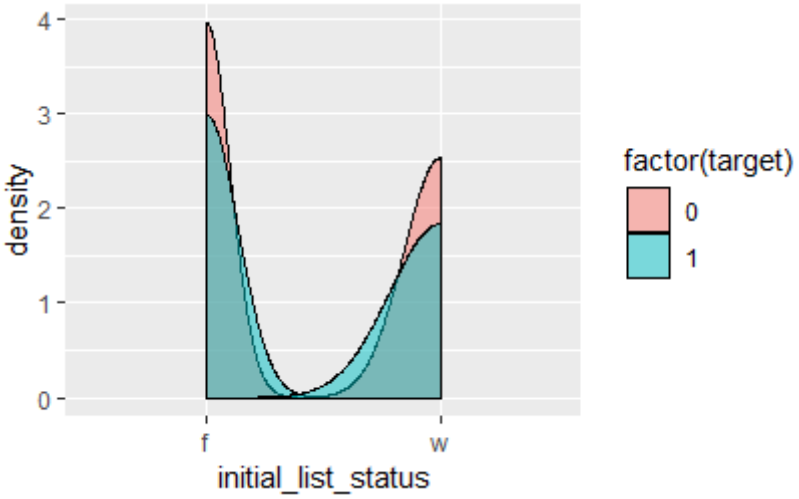

Figura 9: Distribución cruzada variable *total_acc* con *target*

Un análisis rápido nos permite concluir que:

- *term*: las operaciones con un plazo de 60 meses tienen peor comportamiento de pago.
- *grade*: existe una ordenación en la calidad crediticia por letra (A mejor que B, B mejor que C…).
- *sub_grade*: además añade subcategorías numéricas ordenadas a cada letra: 1 mejor que 2, 2 mejor que 3…
- *int_rate*: a menores tipos de interés, menores tasas de malos.
- *annual_inc_joint*: aunque la diferencia es muy sutil, hay peores tasas de malos cuando los ingresos conjuntos son mayores.
- *dti_joint*: cuanto mayor es el ratio de deuda sobre ingresos, mayor la tasa de malos.
- *total_acc*: no se aprecian diferencias en las tasas de malos por número de cuentas bancarias.
- *initial_list_status*: no se aprecian diferencias en tasas de malos entre los dos productos que tienen.

Es importante señalar que, además, los comportamientos observados tienen sentido económico y que ninguna de estas variables está presentando distribuciones cruzadas que van contra la intuición.

3.5 Transformación y selección de variables

Para reducir el conjunto de variables candidatas, que a estas alturas contiene 99 variables, se va a utilizar la métrica *information value*, valor de la información o IV. El IV es una medida de cantidad de información o entropía, que combina la separación entre las dos clases con el porcentaje de población, y que en el caso de una variable discreta se calcula como:

$$IV = \sum_{i=1}^{n} (\%B_i - \%M_i) \cdot \ln\left(\frac{\%B_i}{\%M_i}\right) \quad (1)$$

donde $i = 1, \dots, n$ son las distintas clases (valores) de la variable, y $\%B_i$ y $\%M_i$ son el porcentaje de observaciones buenas y malas que hay en ese grupo respecto al total de operaciones buenas y malas respectivamente.

En el caso de variables continuas, el cálculo sería:

$$IV = \int \left(\left(b(s) - m(s) \right) \cdot \ln\left(\frac{b(s)}{m(s)} \right) \right) ds \tag{2}$$

donde $b(s)$ y $m(s)$ serían las funciones de densidad de las distribuciones de buenos y malos en esa variable.

En realidad, sólo se va a hacer uso de la fórmula para variables discretas, ya que todas las variables continuas van a ser transformadas en variables discretas: haciendo grupos de valores que presenten comportamientos similares en términos de tasa de malos, con tasas de malos significativamente distintos entre grupos, y calculando el *weight of evidence*, peso de la evidencia o WOE de cada uno de los grupos.

El WOE de un grupo i de una variable discreta se calcula como:

$$WOE_i = \ln\left(\frac{\%B_i}{\%M_i} \right) = \ln\left(\frac{\frac{\%b_i}{\%m_i}}{\frac{\%B}{\%M}} \right) = \ln\left(\frac{\%b_i}{\%m_i} \right) - \ln\left(\frac{\%B}{\%M} \right) \tag{3}$$

donde $\%B_i$ y $\%M_i$ son el porcentaje de observaciones buenas y malas que hay en ese grupo respecto al total de operaciones buenas y malas respectivamente, $\%b_i$ y $\%m_i$ son el porcentaje de observaciones buenas y malas que hay en ese grupo respecto al total de operaciones, y $\%B$ y $\%M$ son el porcentaje de buenos y malos que hay en la población total.

La primera expresión es la más extendida para realizar el cálculo, pero la última sirve para entender el signo del WOE de un atributo al comparar la tasa de malos de ese atributo con la tasa de malos de la población total.

De hecho, el cálculo del IV y el WOE está relacionado, ya que:

$$IV = \sum_{i=1}^{n} (\%B_i - \%M_i) \cdot \ln\left(\frac{\%B_i}{\%M_i} \right) = \sum_{i=1}^{n} (\%B_i - \%M_i) \cdot WOE_i \tag{4}$$

Esta práctica está bastante extendida en la industria[14], y algunas de las ventajas de su uso son:

- Permite capturar relaciones no lineales entre variable dependiente e independiente.

[14] De acuerdo a Siddiqi (2005) y Siddiqi (2017), que podrían considerarse como los manuales teóricos que hay detrás de los manuales de programación de SAS, el principal proveedor de software para desarrollar modelos de *scoring* (y, en general, de modelos econométricos o estadísticos).

- Permite que la variable dependiente contenga valores *outliers*: estarán agrupados con valores que presenten un comportamiento similar.
- Permite trabajar con información *missing*: los registros que tienen esa variable a *missing* se agruparan junto a valores que presenten un comportamiento similar, y en algún caso, los registros que tienen esa variable a *missing* podrían llegar a configurar un grupo diferente
- Todas las variables tienen la misma escala y son sencillas de interpretar: si el WOE toma el valor 0, ese grupo presenta una tasa de malos similar a la población total; si el WOE toma valores negativos, ese grupo presenta una tasa de malos superior a la de la población; y si el WOE es positivo, el grupo tendrá una tasa de malos menor a la media. Además, cuanto mayor amplitud entre los WOE de los grupos de una variable, mayor capacidad discriminante tiene esta variable porque habrá mayor diferencia entre las tasas de malos de sus grupos.
- Otro posible enfoque, como el abuso de variables *dummy,* podría generar problemas de multicolinealidad. Además, el uso de variables *dummy* para variables categóricas asume que la diferencia o distancia de un grupo a otro es la misma[15].
- Estandariza las variables.

Para transformar las variables continuas en discretas, se construyen las agrupaciones de tal modo que se maximice el IV de esa variable. Existen otros métodos en los que maximiza otra métrica de información o discriminación como podría ser el Gini o KS. Este paso del desarrollo es el que resulta más costoso en términos computacionales, especialmente si hay un número considerable de variables involucradas. Existen también métodos más avanzados, pero con un enorme coste computacional, que utilizan para construir las agrupaciones alguna métrica como el Gini marginal que ganaría el modelo al meterlas, y se reevalúan las agrupaciones cada vez que una variable entra o sale del modelo (algoritmo MARS[16]).

Los principales proveedores de este tipo de modelos (SAS, FICO, Experian, Equifax), así como algunas empresas (bancos) que hacen mucho uso de modelos de *score*, tienen sus propios algoritmos para hacer eficiente la búsqueda de las agrupaciones, especialmente ahora que el número de variables con las que pueden trabajar se está disparando. El algoritmo que se usa en este caso es el proporcionado por el paquete *smbinning*[17] de R. Este paquete por ejemplo ya garantiza la monotonía de las agrupaciones y que las tasas de malos de los grupos sean significativamente distintas con un nivel de significatividad del 5% (contraste ANOVA), y también permite incorporar los *missings* y elegir el porcentaje mínimo que habrá en cada agrupación (por defecto, el 5%, y así se ha mantenido por considerar que la tasa de malos de la población total, que en la muestra *train* está sobre el 17%, es mucho mayor; si, por ejemplo, se tuviera una tasa de malos en la población del 5%, habría que intentar buscar perfiles en la

[15] De acerudo con Siddiqi (2005), capítulo 6, sección *Logistic Regression.*

[16] Para más detalle sobre el algoritmo *multivariate adaptive regression splies*, se puede consultar Friedman (1991).

[17] Para más detalles sobre el algoritmo, ver la descripción de la función en la documentación de la librería en https://cran.r-project.org/web/packages/smbinning/smbinning.pdf.

muestra menos poblados que permitieran capturar esa baja tasa de malos adecuadamente).

El mismo algoritmo ha sido utilizado para hacer las agrupaciones de las variables discretas: aunque ya existieran categorías en estas variables, es necesario agrupar las que tienen tasas de malos similares. En este caso, cabe destacar que al tener las variables *purpose, inq_last_6mths, sub_grade,* y *title* muchos valores posibles aun siendo categóricas, el algoritmo daba problemas para agruparlas. Se ha hecho una transformación a variable continua asignando a cada uno de los antiguos valores la tasa de malos de esa categoría, y luego se han agrupado como si fueran variables continuas.

Tras realizar las agrupaciones de las variables, calcular los WOEs de sus grupos, y calcular el IV de cada variable, las que superan el umbral del 0,02 de IV son las siguientes 33 variables candidatas:

Variable	IV	Variable	IV	Variable	IV
sub_grade	0,442	total_bc_limit	0,051	total_rev_hi_lim	0,041
grade	0,411	tot_hi_cred_lim	0,051	mort_acc	0,041
int_rate	0,426	mths_since_recent_inq	0,046	mo_sin_rcnt_rev_tl_op	0,040
acc_open_past_24mths	0,101	mo_sin_rcnt_tl	0,046	mths_since_recent_bc	0,039
open_rv_24m	0,077	open_rv_12m	0,045	home_ownership	0,035
num_tl_op_past_12m	0,076	open_acc_6m	0,045	il_util	0,035
verification_status	0,060	open_il_24m	0,045	all_util	0,035
dti	0,055	open_il_12m	0,045	annual_inc	0,032
bc_open_to_buy	0,054	term	0,044	mo_sin_old_rev_tl_op	0,032
inq_last_12m	0,054	mths_since_rcnt_il	0,042	max_bal_bc	0,031
avg_cur_bal	0,052	tot_cur_bal	0,042	inq_fi	0,028

Tabla 12: Variables candidatas e IV asociado

El umbral elegido para determinar las variables candidatas que podrían entrar en el modelo viene de la siguiente interpretación[18] del valor del IV:

IV	Interpretación
<0,02	Variable no predictiva
0,02 - 0,1	Poder predictivo débil
0,1 - 0,3	Poder predictivo medio
0,3 - 0,5	Poder predictivo fuerte
> 0,5	Variable sospechosa (*too good to be true*)

Tabla 13: Interpretación valores *information value*

En caso de no disponer de muchas variables, se podría relajar este umbral al 0,01 y luego ver si esas variables acaban siendo significativas en el modelo, aunque no se da este caso.

A continuación se muestran un ejemplo de agrupación para una variable continua (*dti*), y otro para una variable que ya era discreta (*term*), haciendo también uso de los gráficos que genera el algoritmo utilizado para el porcentaje de población, tasa de malos y WOE

[18] De acuerdo a Siddiqi (2005), capítulo 6, sección *Statistical Measures*.

de cada grupo. En cualquier caso, esta información se mostrará para cada una de las variables que finalmente conformen el modelo en la sección 3.8.

$$WOE\_dti = \begin{cases} 0{,}29, & dti \leq 12{,}83 \\ 0{,}13, & 28{,}72 < dti \leq 17{,}02 \\ 0{,}02, & 17{,}02 < dti \leq 21{,}76 \\ -0{,}09, & 21{,}76 < dti \leq 24{,}49 \\ -0{,}21, & 24{,}49 < dti \leq 28{,}72 \\ -0{,}31, & 28{,}72 < dti \leq 32{,}77 \\ -0{,}48, & dti > 32{,}77 \end{cases} \quad (5)$$

$$WOE\_term = \begin{cases} 0{,}13, & term = \text{"36 } months\text{"} \\ -0{,}33, & term = \text{"60 } months\text{"} \end{cases} \quad (6)$$

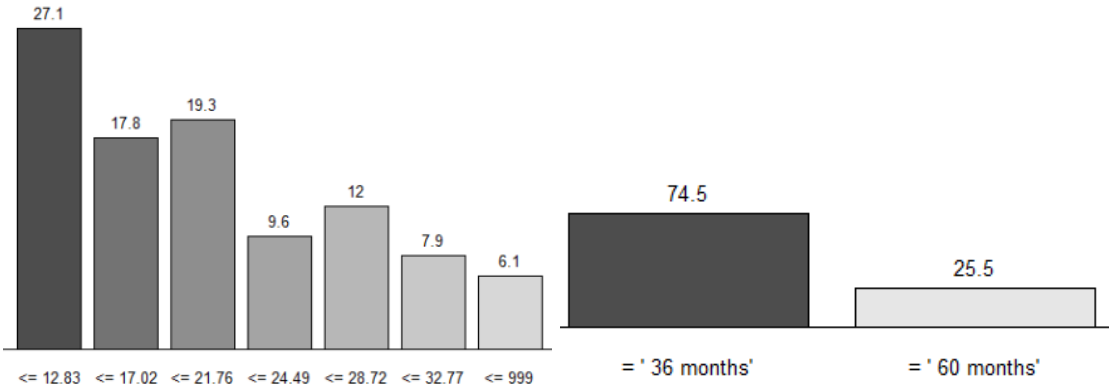

Figura 10: Porcentaje de población en cada grupo de la variable *dti* y *term*

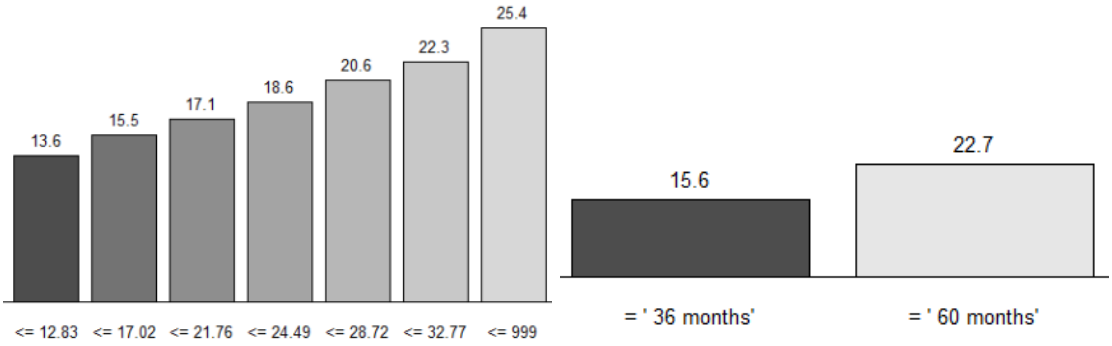

Figura 11: Porcentaje de malos en cada grupo de la variable *dti* y *term*

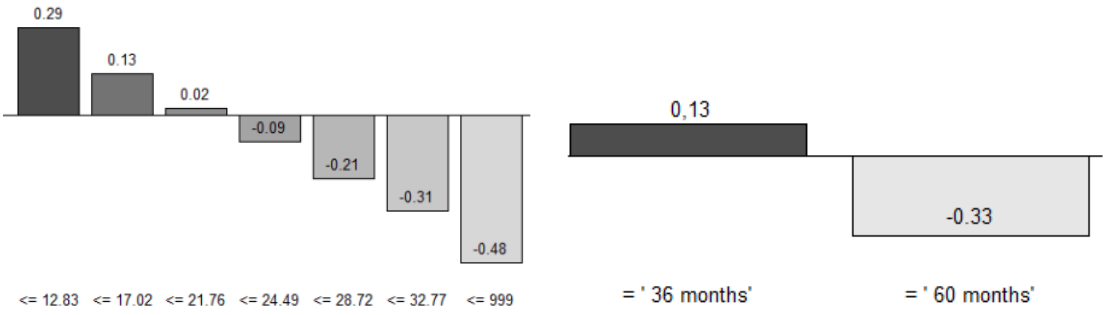

Figura 12: WOE de cada grupo de la variable *dti* y *term*

En el caso de la variable *dti*, aunque la última de las agrupaciones indique "<=999", ya que es el valor máximo que ha encontrado en la muestra *train*, si cuando se aplica un

modelo que contiene esta variable a otra muestra, dicha muestra contiene un registro con un valor superior a ese umbral, también se asignaría a este grupo. Además, ese grupo incorpora a los registros *missing*, que, aunque sólo eran 43 observaciones, tenían una tasa de malos superior al 30%.

Por último, se podría valorar quitar variables del conjunto de candidatas utilizando alguna métrica de correlación, pero al disponer de un número manejable de variables, este contraste se realizará una vez se tengan las variables que forman parte del modelo.

3.6 Construcción y selección del modelo

Regresión logística

Dentro de la familia de modelos de variable dependiente binaria, el que se va a utilizar es el modelo *logit*[19], que no modeliza directamente la probabilidad de un evento (de impago en un año, en nuestro caso), sino que lo hace a través de una función de enlace, los *odds* o *logodds*:

$$odds = \frac{p}{1-p} \tag{7}$$

$$logodds = \ln\left(\frac{p}{1-p}\right) \tag{8}$$

donde p es la probabilidad del evento a modelizar, y $\log$ es el logaritmo neperiano. Nótese que, en la práctica, los *odds* no son otra cosa que la proporción, observada o esperada, de malos entre buenos.

Este modelo hace uso de la función de distribución acumulada logística, lo que permite capturar relaciones más allá de lo lineal, y acota los posibles valores de la variable dependiente al rango (0,1), al tratarse de una probabilidad. Así, este modelo se puede ver como un modelo linear en los *logodds*, o como el modelo *logit*:

$$y = logodds = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \cdot x_1 + \cdots + \beta_k \cdot x_k \tag{9}$$

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \cdot x_1 + \cdots + \beta_k \cdot x_k)}} \tag{10}$$

Aunque la interpretación del significado de los coeficientes no sea tan sencilla como en los modelos de regresión lineal, el hecho de poder ver el modelo como una regresión lineal de los *logodds* facilita su comprensión.

Las estimaciones de los coeficientes se obtienen mediante máxima verosimilitud y no mediante mínimos cuadrados ordinarios. Algunas ventajas, aparte de las mencionadas,

[19] El modelo *logit* se encuadra dentro de los modelos lineales generalizados o GLM. Para más detalles sobre su especificación, asunciones y propiedades, se puede consultar la sección 11.3 de Matilla, Pérez y Sanz (2013), u otro manual de econometría (algo avanzada, ya que es un modelo que no se suele abordar en los manuales más introductorios) como Greene (2003), Peña (2002) o Johnson y Wichern (2007), o Siddiqi (2005) para ver un enfoque más orientado hacia la construcción de *scorecards*.

de esta especificación frente al modelo de regresión múltiple u otro modelo de variable dependiente binaria son:

- Es más eficiente.
- No asume normalidad de los errores[20], que es una hipótesis restrictiva en otros modelos, o que requiere de contrastes adicionales.
- No necesita asumir que la varianza de la variable dependiente y es constante[21] respecto a las variables independientes. La distribución condicionada de y sigue una distribución logística (Bernoulli).
- Permite utilizar variables independientes de cualquier tipo, tanto cuantitativas como cualitativas: continuas, discretas, fechas, categóricas.

En realidad, el valor estimado para una observación no es otra cosa que la probabilidad de que esa variable aleatoria tome el valor 1, o, dicho de otro modo, de que esa observación presente un impago:

$$\widehat{y_i} = \mathrm{E}(y_i) = 0 \cdot Pr(y_i = 0) + 1 \cdot Pr(y_i = 1) \tag{11}$$

Así, en nuestro caso, el modelo *logit* va a construir una predicción para la probabilidad de malo.

Además, no se van a utilizar las variables originales como variables explicativas, sino que se van a utilizar las transformaciones a variables WOE utilizando su *weight of evidence*, como ya se indicó en la sección anterior. El modelo utilizado será entonces:

$$y = logodds = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \cdot WOE_{x_1} + \cdots + \beta_k \cdot WOE_{x_k} \tag{12}$$

$$p = \frac{1}{1 + e^{-\left(\beta_0 + \beta_1 \cdot WOE_{x_1} + \cdots + \beta_k \cdot WOE_{x_k}\right)}} \tag{13}$$

Una de los problemas de trabajar con este tipo de modelos es que es necesario un número mínimo de observaciones en cada una de las categorías (buenos y malos) para poder hacer una construcción robusta. En este sentido, la r*ule of ten*[22] indica que se requieren al menos 10 observaciones de cada categoría (en este caso, la menos poblada es la categoría de malos) por cada variable y grupo que haya en el modelo. Por ejemplo, un modelo de 15 variables con 10 grupos cada una, necesitaría de al menos 1.500 observaciones malas en la muestra *train*. Algunos estudios sugieren incluso que se deben tener 20 o 50[23] observaciones por grupo, lo que requeriría hasta 7.500 observaciones malas siguiendo con el ejemplo anterior. En este desarrollo, ninguno de estos umbrales supone un problema, pero en otro tipo de ejercicios sí que es una restricción importante (por ejemplo, un modelo para puntuar una cartera de grandes empresas en la que se esperaría que haya pocos impagos, o un modelo de ratings

[20] De acuerdo a Nieto (2010).

[21] De acuerdo a Johnson y Wichern (2007), sección 11.7

[22] Para más detalle, ver Harrell, Lee y Mark (1996).

[23] Para más detalle, ver Eijkemans, Habbema, Harrell y Steyerberg (2000).

soberanos como los que construyen las agencias de calificación como Standard & Poor's o Moody's).

Selección de variables

A medida que el conjunto de variables candidatas crece, se hace necesario contar con algún método que ayude en la selección de variables.

Dado que se dispone de 33 variables, un posible enfoque, aunque muy costoso computacionalmente, sería construir todos los modelos posibles con estas variables: todos los modelos que tengan 1, 2, 3,…, 32, 33 variables, y, tras alguna restricción que se comentará a lo largo de este apartado (significatividad de las estimaciones, estimaciones negativas, correlación), utilizar alguna métrica, como el mayor poder discriminante en la muestra *test*, para elegir el modelo final.

Este enfoque podría parecer una locura a nivel computacional, porque exige simular millones de modelos, pero existen servidores potentes con numerosos procesadores, núcleos y por tanto hilos de ejecución, que permiten estimar miles de modelos al mismo tiempo gracias a ejecuciones distribuidas. Además, es posible reducir mucho el número de simulaciones si se utiliza el algoritmo de ramificación y poda (branch & bound[24]).

Otra posibilidad es utilizar métodos de selección basados en contrastes de significatividad conjunta (F) o individual (t), y que utilizan alguna métrica de aportación al modelo en términos de explicatividad/discriminación (Gini marginal, *information value* marginal) para decidir si introducir más variables, sacar alguna de las que forma parte del modelo, o parar. En este grupo se pueden incluir los métodos *forward*, *backward*, y *stepwise*[25], que requieren de un cierto número de iteraciones, pero reducen el tiempo de ejecución considerablemente respecto a la alternativa anterior, aunque ese nivel de automatización hace que se pierda la traza de lo que está pasando[26].

- *Forward* o hacia delante: se van añadiendo variables al modelo en función del poder predictivo que aportarían, siempre y cuando la nueva variable supere cierto umbral de significatividad.
- *Backward* o hacia atrás: se incluyen todas las variables candidatas al modelo, y en cada etapa se elimina la variable menos significativa, hasta que todas las variables incluidas en el modelo son significativas.
- *Stepwise* o paso a paso: es una combinación de ambos, ya que se van añadiendo variables una a una al modelo, pero también se evalúa cada vez que el modelo cambia si todas las variables incluidas siguen siendo significativas.

[24] Para más detalles sobre el uso del algoritmo de ramificación y poda en la selección de variables de modelos de regresión, ver Furnival y Wilson (2000).

[25] Por ejemplo, en la sección 8.2.4 de Greene (2003) se mencionan estos enfoques (*simple-to-general*, *general-to-simple*, y *stepwise model building*), y se detallan en la sección 3.3 de Hastie, Tibshirani y Friedman (2008) o sección 6.1 de James, Hastie, Witten y Tibshirani (2013).

[26] Algunos softwares como SAS sí que permiten ver qué variable va entrando o saliendo del modelo en cada iteración y por qué.

Para este ejercicio, se va a proceder de manera similar al *backward*, aunque manualmente, para tener trazabilidad de lo que está pasando y así poder revisar los signos de los coeficientes estimados, observar los p-valores de los coeficientes cada vez que se quita una variable, y calcular correlaciones de las variables que hay dentro del modelo.

Modelo 1

En primer lugar, se construye un modelo que incorpora las 33 variables WOE como variables independientes.

$$\hat{y} = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) = \widehat{\beta_0} + \widehat{\beta_1} \cdot WOE_1 + \cdots + \widehat{\beta_{33}} \cdot WOE_{33} \tag{14}$$

$$\hat{p} = \frac{1}{1 + e^{-\left(\widehat{\beta_0} + \widehat{\beta_1} \cdot WOE_1 + \cdots + \widehat{\beta_{33}} \cdot WOE_{33}\right)}} \tag{15}$$

Variable	Estimación	p-valor	Variable	Estimación	p-valor
Intercept	-1,537393	2E-16	WOE_acc_open_past_24mths	-0,358595	1,554E-07
WOE_int_rate	-0,155207	0,000154	WOE_avg_cur_bal	-0,095717	0,076214
WOE_annual_inc	-0,089313	0,024932	WOE_bc_open_to_buy	-0,313407	5,6E-12
WOE_dti	-0,445349	2E-16	WOE_mo_sin_old_rev_tl_op	-0,104004	0,000738
WOE_tot_hi_cred_lim	-0,442459	1,26594E-06	WOE_mo_sin_rcnt_rev_tl_op	0,094636	0,052296
WOE_open_acc_6m	-0,106155	0,003456	WOE_mo_sin_rcnt_tl	0,006919	0,878338
WOE_open_il_12m	0,087288	0,196221	WOE_mort_acc	-0,161174	7,169E-06
WOE_open_il_24m	0,055132	0,387396	WOE_mths_since_recent_bc	-0,324298	2E-16
WOE_mths_since_rcnt_il	-0,07667	0,088049	WOE_mths_since_recent_inq	-0,242376	1,02E-11
WOE_il_util	-0,082958	0,024222	WOE_num_tl_op_past_12m	-0,164949	0,044207
WOE_open_rv_12m	0,34571	5,43753E-05	WOE_term	-0,382495	2E-16
WOE_open_rv_24m	-0,296993	1,12582E-05	WOE_grade	-0,664698	2E-16
WOE_total_bc_limit	0,054999	0,38113	WOE_home_ownership	-0,437967	2E-16
WOE_all_util	-0,055898	0,176547	WOE_verification_status	-0,29506	2E-16
WOE_total_rev_hi_lim	-0,117211	0,027963	WOE_tot_cur_bal	0,083923	0,363374
WOE_inq_fi	-0,181607	3,80532E-07	WOE_max_bal_bc	-0,210307	1,358E-05
WOE_inq_last_12m	-0,097418	0,00764	WOE_subgrade	0,024932	0,610741

Tabla 14: Estimaciones y p-valores coeficientes modelo 1

En este primer modelo aparecen:

- 7 estimaciones de coeficientes con p-valor superior al 10%.
- Otras 3 estimaciones con p-valores superiores al 5%.
- Otras 4 estimaciones con p-valores superiores al 1%.
- Otras 2 estimaciones con p-valores superiores al 0,1%.
- 17 estimaciones con p-valores inferiores al 0,1%.
- 7 coeficientes estimados positivos, 5 de ellos con p-valores no significativos al 10%.

Ese p-valor corresponde al contraste de hipótesis:

$$\begin{cases} H_0: \beta_i = 0 \\ H_1: \beta_i \neq 0 \end{cases} \tag{16}$$

donde β_i sería el parámetro poblacional correspondiente al coeficiente de cada una de las variables $i = 0{,}1, \ldots, 33$.

Los betas estimados negativos estarían reflejando resultados espurios que aparecen por el alto número de variables que hay dentro del modelo y por la baja significatividad de algunos de los coeficientes estimados. Estos signos no son correctos debido a cómo se han definido las variables WOE:

- Si una variable WOE toma un valor negativo, es porque la tasa de malos en ese grupo es más alta que la media de la población.
- Si una variable WOE toma un valor positivo, es porque la tasa de malos en ese grupo es más baja que la media de la población.
- Si todas las variables WOE tomaran valor 0, el valor predicho por el modelo sería la tasa de malos de la población total, recogida en el intercepto: $E(p|WOE_1, \dots, WOE_{33} = 0) = \frac{1}{1+e^{-\hat{\beta}_0}} = \frac{1}{1+e^{-(-1,537393)}} = 0{,}1769$.
- Si una variable WOE toma un valor negativo, al estar multiplicada por un beta estimado negativo, lo que hace es "sumar" al intercepto. Por lo tanto, el valor de la estimación del *logodd* sería mayor, y así el valor esperado para la tasa de malo.
- Si una variable WOE toma un valor positivo, al estar multiplicada por un beta estimado negativo, lo que hace es "restar" al intercepto. Por lo tanto, el valor de la estimación del *logodd* sería menor, y así el valor esperado para la tasa de malo.
- Con las estimaciones negativas, a mayor WOE, menor valor en la estimación del *logodd*, y por tanto menor valor esperado para la probabilidad de malo. Si una estimación fuera positiva, se estaría diciendo lo contrario, que cuando una variable tiene un WOE mayor, y por tanto la población en ese grupo presenta una tasa de malo observada más baja que la población total, el valor predicho para la probabilidad de malo sería mayor.

Por otro lado, el valor del coeficiente Gini en la muestra *train* es del 39%, y es también del 39% en la muestra *test*, lo que indicaría ausencia de sobreajuste a la muestra *train*.

Coeficiente Gini

El estadístico que se va a utilizar para medir la capacidad discriminante del modelo, es decir, para medir la capacidad que tiene la salida[27] del modelo de separar las clases de buenos y malos, es el coeficiente de Gini. Cuando infiera un comportamiento, el modelo de *scoring* va a generar una ordenación en la población; éste coeficiente pretende evaluar cómo de buena es esa ordenación, si los registros malos están en la parte mala de la ordenación (si es una probabilidad, cerca de la probabilidad de malo inferida 100%), y viceversa.

El coeficiente Gini mide la distancia acumulada de la distribución del modelo para la clase de buenos frente a la de la clase de malos:

[27] El término "salida del modelo" se utiliza para referirse al comportamiento inferido por el modelo en una población, ya sea la propia muestra con la que se ha construido, u otra muestra (*test*, OOT, TTD).

$$Gini = 1 - \sum_{i=1}^{n-1} (M_{i+1} - M_i) \cdot (B_{i+1} + B_i) \quad (17)$$

donde M_i es el porcentaje acumulado de la distribución de malos hasta la puntuación i, B_i es el porcentaje acumulado de la distribución de buenos hasta la puntuación i, y n es el número de posibles salidas diferentes del modelo cuando se puntúa una determinada población.

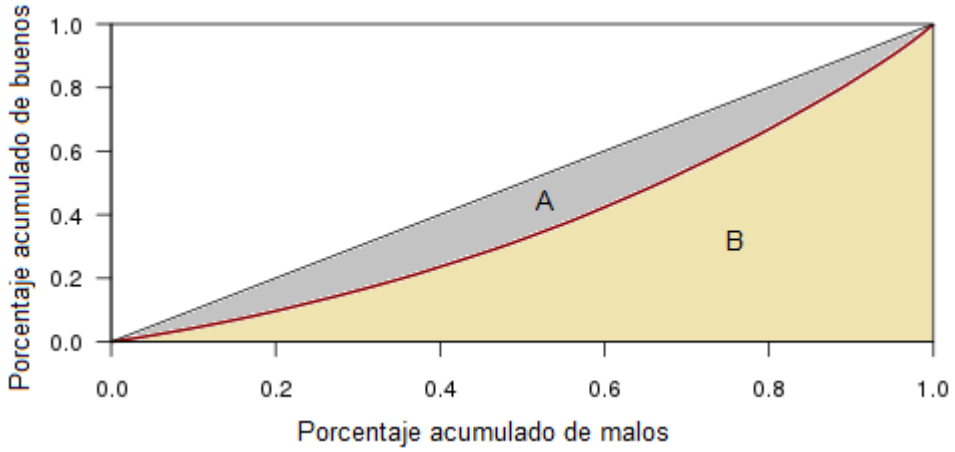

Figura 13: Ejemplo de cálculo coeficiente de Gini

En el ejemplo anterior, se puede deducir a partir de la fórmula dada que el coeficiente Gini es $Gini = \frac{A}{A+B} = 2 \cdot A$.

Este coeficiente toma valores entre 0%[28] y 100% (o 0 y 1), y se puede interpretar cómo:

- Si vale 0%, el modelo no discrimina entre buenos y malos, ya que las distribuciones de buenos y malos se acumulan exactamente igual.
- Si vale 100%, el modelo presentaría discriminación perfecta, ya que hasta cierta puntuación, acumula el 100% de los registros malos y el 0% de los buenos.
- Cuanto más cercano esté el coeficiente al 100%, mejor poder discriminante presenta el modelo.

Existen otros coeficientes[29] similares como el KS, que mide la distancia máxima entre las dos curvas de distribución acumulada, y el AUC (*area under the ROC curve*), que es equivalente al Gini (ver nota al pie 33).

Existen otros métodos para evaluar los modelos de clasificación, como puede ser la matriz de confusión y las métricas que se pueden derivar de ella (tasa de falsos positivos, de falsos negativos, de verdaderos positivos y de verdaderos negativos, precisión o exactitud), pero algunas pueden llevar a conclusiones erróneas cuando las clases de buenos y malos están desbalanceadas (por ejemplo, si sólo hay un 1% de malos, y se clasifican a todos como buenos, el modelo estaría clasificando

[28] Nótese que es posible que el coeficiente Gini tome valores negativos en alguna población, si el modelo discrimina en sentido contrario al observado.

[29] Ver, por ejemplo, discusiones en Rezac y Rezac (2011).

correctamente al 99% de las observaciones, pero al 0% de los malos), y además requiere de un punto de corte que sirva para determinar, en la población inferida, quiénes serán marcados como buenos y como malos (por ejemplo, se podría determinar que se van a marcar como malos inferidos las observaciones cuya probabilidad inferida por el modelo sea superior al 50% o a la tasa de malos que había en la población de desarrollo). En cambio, el coeficiente de Gini no necesita este tipo de puntos de corte.

Se va a proceder a construir sucesivos modelos eliminando las variables menos significativas, comprobando si se corrigen los signos negativos de algunas de las estimaciones o sacando esas variables si fuera necesario, y a su vez controlando que el valor del estadístico Gini, que mide el poder discriminante del modelo, se mantiene en sucesivos modelos.

Modelo 2

El segundo modelo se construye utilizando sólo 23 variables independientes, tras quitar las variables WOE *mo_sin_rcnt_rev_tl_op*, *avg_cur_bal*, *mths_since_rcnt_il*, *open_il_12m*, *open_il_24m*, *total_bc_limit*, *all_util*, *mo_sin_rcnt_tl*, *tot_cur_bal* y *sub_grade* por no poder rechazarse la hipótesis nula (parámetro poblacional del coeficiente igual a cero) con un nivel de significatividad del 5%.

En este modelo aparecen:

- 1 estimación con p-valor superior al 5%.
- Otras 2 estimaciones con p-valores superiores al 1%.
- Otras 2 estimaciones con p-valores superiores al 0,1%.
- 18 estimaciones con p-valores inferiores al 0,1%.
- 1 coeficiente estimado positivo.

El estadístico Gini obtenido en las muestras *train* y *test* es del 39% y 38% respectivamente.

Modelo 3

El tercer modelo se construye utilizando 22 variables WOE independientes: simplemente excluye la variable WOE *total_rev_hi_lim* por no ser significativa al 5%.

En este modelo se observa que todas las variables son significativas al 5%:

- 2 estimaciones con p-valores superiores al 1%.
- Otras 2 estimaciones con p-valores superiores al 0,1%.
- 18 estimaciones con p-valores inferiores al 0,1%.
- 1 coeficiente estimado positivo.

El coeficiente Gini toma el valor 39% en las muestras *train* y *test*.

Modelo 4

Este modelo se construye con 21 variables candidatas tras eliminar la variable WOE *open_rv_12m* por tener un coeficiente beta con estimación positiva.

En el nuevo modelo se tienen:

- 2 estimaciones con p-valores superiores al 1%.
- Otra estimación con p-valor superior al 0,1%.
- 18 estimaciones con p-valores inferiores al 0,1%.
- 1 coeficiente estimado positivo, que tiene un p-valor superior al 1%.

El estadístico Gini obtenido en las muestras *train* y *test* es del 39% y 38% respectivamente.

Modelo 5

El modelo 5 se construye tras eliminar la variable WOE *num_tl_op_past_12m* por tener una estimación positiva, contra 20 variables candidatas.

Los nuevos coeficientes presentan:

- 1 estimación con p-valor superiores al 5%.
- Otra estimación con p-valor superior al 1%.
- Otra estimación con p-valor superior al 0,1%.
- 17 estimaciones con p-valores inferiores al 0,1%.

El coeficiente Gini obtenido en las muestras *train* y *test* es del 39% y 38% respectivamente.

Modelo 6

El modelo 6 se construye con 19 variables independientes, tras eliminar la variable WOE *open_acc_6m* por no ser significativa al 5%.

Los nuevos coeficientes presentan:

- 1 estimación con p-valor superior al 1%.
- Otra estimación con p-valor superior al 0,1%.
- 17 estimaciones con p-valores inferiores al 0,1%.

El coeficiente Gini obtenido en las muestras *train* y *test* es del 39% y 38% respectivamente.

Modelo 7

El modelo 7 se construye con 18 variables independientes, tras eliminar la variable WOE *annual_inc* por no ser significativa al 1%.

Los nuevos coeficientes presentan:

- 1 estimación con p-valor superior al 0,1%.
- 17 estimaciones con p-valores inferiores al 0,1%.

El coeficiente Gini obtenido en las muestras *train* y *test* es del 39% y 38% respectivamente.

Llegados a este punto, se podría empezar a pensar en el modelo como finalista, y aplicar los contrastes de la sección siguiente. Por otro lado, de acuerdo al principio de

parsimonia[30], se va a explorar si existen modelos más sencillos que mantienen la capacidad discriminante con un número menor de variables. Aunque en el siguiente modelo se va a reducir el número de variables, las que conforman el modelo 7 se tendrían en cuenta en caso de que en alguno de los contrastes que se aplican en el siguiente modelo indicara la necesidad de eliminar una variable (por correlación, inestabilidad de la población, o inestabilidad del poder discriminante) y eso causara una caída en el poder discriminante del modelo.

Modelo 8

Este modelo se ha construido únicamente con 12 variables, eliminando una a una las variables menos significativas (aun siendo todas significativas al 1%) y obteniendo sucesivos modelos que mantenían el poder discriminante. Las variables que han sido eliminadas son: *inq_last_12m*, *open_rv_24m*, *il_util*, *inq_fi*, *mo_sin_old_rev_tl_op*, *mort_acc*.

En este nuevo modelo, las 12 variables son significativas para un nivel de significatividad tan extremo como $1 \cdot 10^{-15}$. Además, el valor del estadístico Gini calculado en las muestras *train* y *test* sigue siendo del 39% y 38% respectivamente. Sobre este modelo se realizarán los contrastes de correlación, estabilidad del poder discriminante y estabilidad de la población en la siguiente sección.

Modelo vigente

Para tener una idea del poder discriminante del modelo que actualmente está utilizando Lending Club, se construye un modelo contra la variable *sub_grade*, ya que es la calificación más granular que se ofrece a los inversores.

$$\hat{y} = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1{,}208871 - 1{,}409431 \cdot WOE\_sub\_grade \qquad (17)$$

Este modelo, con coeficientes estimados significativos, presenta un valor de Gini en *train* y *test* del 27,7% y 27,4% respectivamente. Por lo tanto, el modelo 8 mejoraría el poder discriminante en 11,3 y 10,6 puntos de Gini, lo que supone un incremento del 41% y 39% relativo en esas muestras.

3.7 Contrastes aplicados en el modelo

Correlación

En primer lugar, se va a evaluar la correlación de las variables que componen el modelo. Aunque el índice de correlación de Pearson puede dar una idea de la correlación de las variables, ya que al tomar valores numéricos se podrían considerar variables continuas,

[30] Ver sección 2.5 en Gujarati y Porter (2010) para el principio de parsimonia, y sección 3.4 en Hastie, Tibshirani y Friedman (2008) o sección 6.2 en James, Hastie, Witten y Tibshirani (2013) para discusiones sobre métodos de regresiones penalizadas como *lasso* o *ridge regression*.

lo más adecuado para medir la correlación entre variables categóricas es utilizar el índice V de Cramer[31], que se calcula como:

$$V = \sqrt{\frac{\chi^2/_n}{\min(k_1 - 1, k_2 - 1)}} \tag{18}$$

donde n es el número de observaciones, k_1 y k_2 son el número de categorías de las dos variables que se están estudiando, y χ^2 es el contraste chi-cuadrado de Pearson, que se obtiene como:

$$\chi^2 = \sum_{i,j} \frac{\left(n_{i,j} - \frac{n_i \cdot n_j}{n}\right)^2}{\frac{n_i \cdot n_j}{n}} \tag{19}$$

donde i y j son cada uno de los atributos de las dos variables comparadas, n_i y n_j son el número de observaciones que toman el valor de las categorías i y j respectivamente, y $n_{i,j}$ es el número de observaciones que toman el valor de las categorías i y j para el mismo registro.

De acuerdo a esta métrica, dos variables se consideran correlacionadas si el índice es superior al 40%[31].

Los resultados de este estadístico de asociación para cada par de variables son:

	int_rate	dti	tot_hi_cred_lim	acc_open_past_24mths	bc_open_to_buy	mths_since_recent_bc	mths_since_recent_inq	term	grade	home_ownership	verification_status	max_bal_bc
int_rate	100%	9,13%	7,15%	7,88%	14,00%	4,59%	6,36%	35,74%	68,27%	6,21%	20,91%	4,67%
dti	9,13%	100%	7,72%	6,40%	2,70%	1,42%	2,18%	6,38%	10,02%	1,91%	12,11%	5,34%
tot_hi_cred_lim	7,15%	7,72%	100%	6,33%	11,23%	2,01%	4,60%	14,33%	6,64%	36,46%	8,33%	15,42%
acc_open_past_24mths	7,88%	6,40%	6,33%	100%	5,51%	18,58%	15,12%	2,86%	9,19%	4,80%	6,65%	6,01%
bc_open_to_buy	14,00%	2,70%	11,23%	5,51%	100%	37,75%	2,22%	3,80%	14,30%	5,77%	13,77%	11,77%
mths_since_recent_bc	4,59%	1,42%	2,01%	18,58%	37,75%	100%	12,10%	2,98%	5,15%	2,29%	3,88%	8,38%
mths_since_recent_inq	6,36%	2,18%	4,60%	15,12%	2,22%	12,10%	100%	2,92%	7,74%	5,87%	4,34%	4,14%
term	35,74%	6,38%	14,33%	2,86%	3,80%	2,98%	2,92%	100%	37,90%	10,12%	6,39%	13,03%
grade	68,27%	10,02%	6,64%	9,19%	14,30%	5,15%	7,74%	37,90%	100%	5,74%	18,89%	4,43%
home_ownership	6,21%	1,91%	36,46%	4,80%	5,77%	2,29%	5,87%	10,12%	5,74%	100%	6,52%	9,04%
verification_status	20,91%	12,11%	8,33%	6,65%	13,77%	3,88%	4,34%	6,39%	18,89%	6,52%	100%	5,31%
max_bal_bc	4,67%	5,34%	15,42%	6,01%	11,77%	8,38%	4,14%	13,03%	4,43%	9,04%	5,31%	100%

Tabla 15: Matriz de correlaciones V de Cramer

[31] Para más detalle sobre los distintos coeficientes de correlación y su interpretación, ver Smeeton y Sprent (2001).

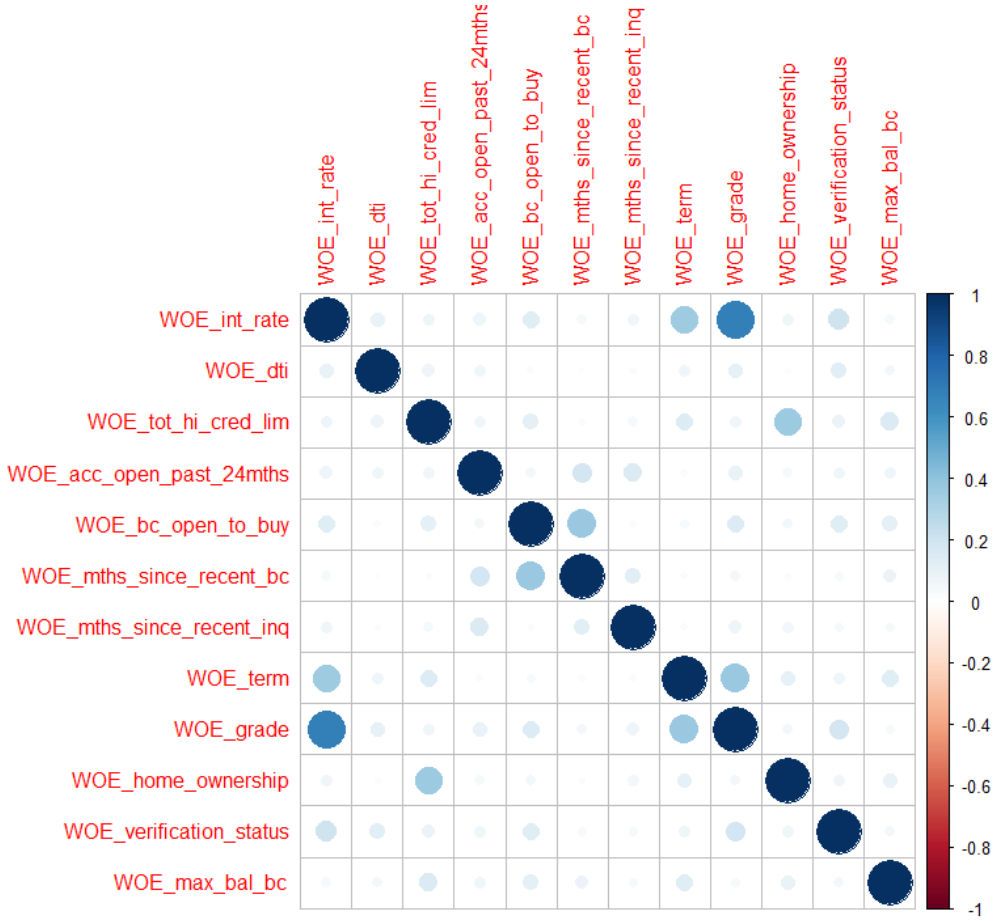

Figura 14: Mapa de calor sobre la matriz de correlaciones V de Cramer

Las variables *int_rate* y *grade* presentan una correlación alta (68,27%). Dado que desde un principio la inclusión de la variable que determina el tipo de interés a la operación generaba dudas, ya que es difícil determinar si es una variable que se dispone en el momento de estudiar la admisión o después, se va a eliminar del modelo.

Obviamente, una vez eliminada no será necesario recalcular la matriz de correlación, ya que la correlación de dos variables no depende del resto de variables, y por tanto estos valores no cambiarán al eliminar la variable *int_rate*.

Modelo 9 (final)

Este modelo se ha construido con 11 variables[32], eliminando la variable *int_rate* por la alta correlación con la variable *grade*. Si la política de precios de Lending Club está basada en riesgos, era esperable esta correlación, ya que a clientes con mayor riesgo se les cobrará un tipo de interés mayor, y viceversa.

[32] Se puede consultar el significado de cada una de las variables en el Anexo 2.

$$\hat{y} = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1{,}562775 - 0{,}457695 \cdot WOE\_dti - 0{,}569306 \cdot WOE\_tot\_hi\_cred\_lim - 0{,}558581 \cdot WOE\_acc\_open\_past\_24mths - 0{,}358613 \cdot WOE\_bc\_open\_to\_buy - 0{,}311551 \cdot WOE\_mths\_since\_recent\_bc - 0{,}351475 \cdot WOE\_mths\_since\_recent\_inq - 0{,}348293 \cdot WOE\_term - 0{,}740077 \cdot WOE\_grade - 0{,}500393 \cdot WOE\_home\_ownership - 0{,}312858 \cdot WOE\_verification\_status - 0{,}289146 \cdot WOE\_max\_bal\_bc \quad (20)$$

En el modelo final, las 11 variables son significativas para niveles de significatividad habituales. Por último, el valor del estadístico Gini calculado en las muestras *train* y *test* alcanza el 39,11% y 38,27% respectivamente, lo que supone un incremento relativo del 41,19% y del 39,67% respecto al modelo que contiene únicamente la variable WOE *sub_grade*.

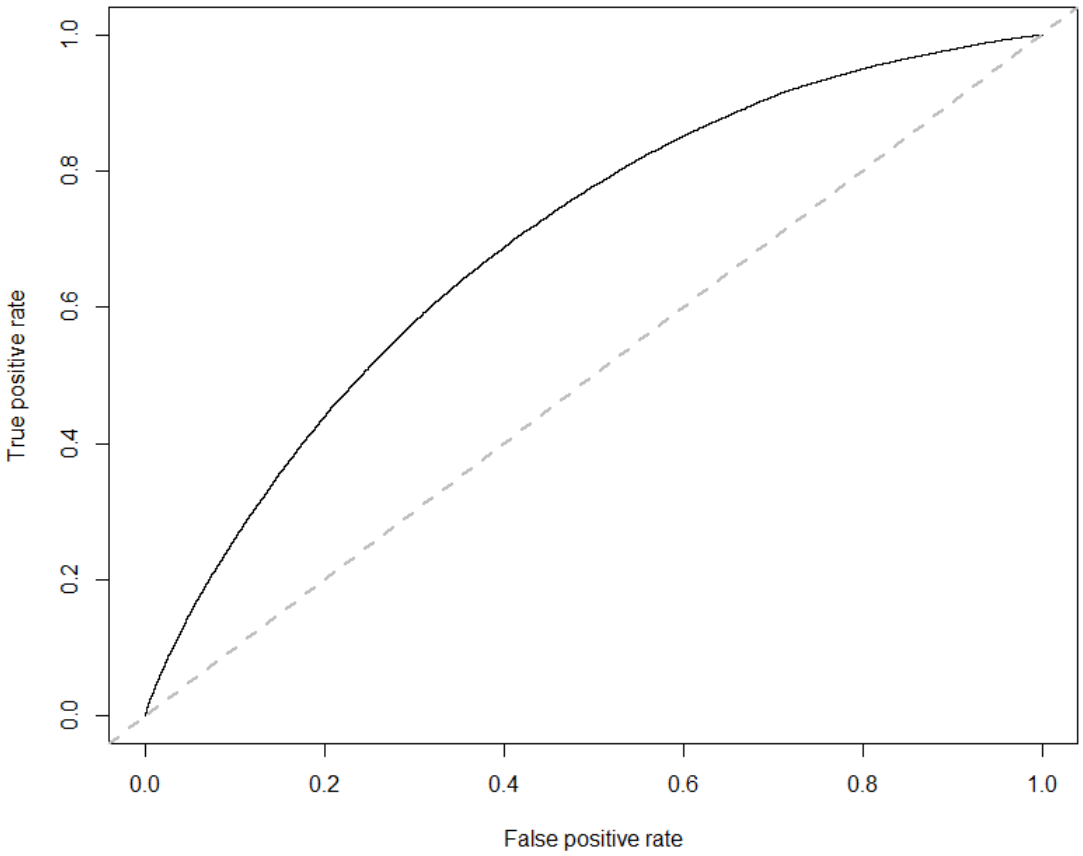

Figura 15: Curva ROC[33] del modelo final

[33] Es posible obtener el coeficiente Gini a partir de la curva ROC como Gini=2·AUC-1, donde AUC es el área bajo la curva ROC. La curva de un modelo con una discriminación aleatoria estaría sobre la línea diagonal intermitente, mientras que la curva de un modelo con una discriminación perfecta recorrería el eje vertical izquierdo y horizontal superior. Para más detalle, ver Hand y Till (2001).

Estabilidad del poder discriminante

Este tipo de modelos se ve afectado por el comportamiento macro y microeconómico, y también por los cambios en las políticas de crédito que aplique el prestamista (aceptar mejores o peores perfiles de riesgo). Por tanto, es necesario evaluar si la discriminación es estable a lo largo del tiempo, comprobando que el modelo presenta niveles de discriminación similares en otra ventana temporal.

La muestra *out-of-time* (OOT), que contiene operaciones admitidas en 2017Q1 - 2017Q4, fue construida con tal propósito. Las variables WOE también han sido construidas en la muestra OOT para poder inferir un comportamiento a esta población. El resultado es un coeficiente Gini del 38,06%, frente al 38,27% que presentaba en la muestra *test*, por lo que se observa que no hay un deterioro temporal del poder discriminante del modelo. Además, el modelo construido sólo contra la variable WOE *sub_grade* arroja un Gini del 27,99%, por lo que el modelo final supondría una mejora relativa del 35,98% en esta muestra.

Si en este contraste se hubiera encontrado una fuerte caída de la capacidad discriminante del modelo, hubiera sido necesario repetir el análisis a nivel variable (comparando el IV o Gini de las variables en esta venta frente al que tenían en la muestra *train*) para detectar qué variables estarían causando la caída, con el fin de sacarlas del modelo final o de reemplazarlas por alguna de las variables que fueron eliminadas cuando se simplificó el modelo (parsimonia).

Estabilidad de la población

Cambios en la población pueden ser un indicador temprano de deterioro del poder discriminante: por ejemplo, si hay un cambio en la población de la variable que más aporta al modelo, y ahora toda la población cae en el mismo grupo, esta variable dejará de discriminar, aunque todavía no sea posible medir el desempeño de la población (por no haber pasado un año desde que se admitieron las operaciones).

La ventana *through-the-door*, con información sobre las operaciones admitidas en 2018Q1 - 2018Q4, fue construida para poder medir la estabilidad de la población en una ventana más reciente en la que no fuera posible medir completamente el desempeño de las operaciones.

Para medir la estabilidad se va a utilizar la métrica índice de estabilidad de la población, *population stability index* o PSI, que compara la distribución de la población en una población de referencia (*train*) frente a la distribución en una población más reciente (TTD). La variable cuya distribución se va a comparar en las dos poblaciones es la probabilidad de malo que asigna (es decir, infiere) el modelo a cada observación. Para su cálculo, es necesario definir grupos en los que se va comparar la distribución de la población, normalmente deciles de la población de referencia; en ese caso, un PSI superior al 25% indicaría que hay un cambio significativo en la población, mientras que un resultado superior al 10% indicaría un ligero cambio en la distribución. El índice PSI se calcula como:

$$PSI = \sum_{i=1}^{10} (p_{i,ref} - p_{i,act}) \cdot \ln\left(\frac{p_{i,ref}}{p_{i,act}}\right) \quad (21)$$

donde $i = 1, \dots, 10$ son cada uno de los deciles en la población de referencia (*train*), $p_{i,ref}$ es el porcentaje de población en el decil i en la población de referencia (debería estar en torno al 10%), mientras que $p_{i,act}$ es el porcentaje de la población más reciente (TTD) en ese mismo decil.

Existen otras métricas para evaluar la estabilidad de la población como el KS[34], que en lugar de comparar las distribuciones de buenos y malos como cuando es utilizado para medir la discriminación de un modelo, compara la distribución de la predicción en la población *train* y TTD.

El cálculo del PSI en la muestra TTD da un valor del 17,16%. Dado que el valor supera el umbral del 10%, se va a repetir el cálculo a nivel variable para comprobar si hubiera alguna variable sufriendo un cambio de población significativo (superior al 25%), para proceder a eliminarla o sustituirla por una variable que permitiera mantener niveles de discriminación similares.

El cálculo del PSI para variables es similar al cálculo del PSI para la población, que al fin y al cabo es el PSI para la variable estimada de probabilidad del evento malo. Cuando las variables son continuas, se procedería por tanto utilizando los deciles, pero cuando las variables son categóricas, se compara el porcentaje de población en cada uno de los grupos.

El PSI para cada una de las variables que forman parte del modelo final viene recogido en la siguiente tabla:

Variable	PSI
dti	3,14%
tot_hi_cred_lim	0,35%
acc_open_past_24mths	1,55%
bc_open_to_buy	9,11%
mths_since_recent_bc	0,99%
mths_since_recent_inq	1,14%
term	1,84%
grade	10,27%
home_ownership	1,07%
verification_status	7,31%
max_bal_bc	0,65%

Tabla 16: PSI de las variables que componen el modelo final

Ninguna de ellas supera el umbral del 25%, por lo que no se considera necesario eliminarla del modelo final o sustituirla por otra variable, aunque sí que sería necesario prestar atención a cómo evoluciona la población de ciertas variables (*grade*, que supera

[34] De acuerdo a Siddiqi (2005), capítulo 8, sección *System Stability Report*.

el umbral del 10%, pero también *bc_open_to_buy* y *verification_status*) en futuras ventanas temporales.

Indicador temprano del desempeño del modelo

Aunque en la muestra TTD no se disponga del desempeño completo de las operaciones aceptadas, que se medía hasta 12 meses después de ser aceptadas, sí que es posible medir un desempeño temprano: por ejemplo, las operaciones aceptadas en el primer trimestre podrían presentar retrasos en los primeros 11, 10 o 9 meses. Este valor debe ser tomado como algo orientativo, pero como en este caso se dispone de 24.701 malos en la muestra TTD, merece la pena su cálculo. En cualquier caso, sólo se debería considerar modificar el modelo obtenido si se obtuviera un deterioro muy significativo en el coeficiente Gini.

El Gini del modelo final sobre la muestra TTD toma el valor 39,19%, por lo que parece que el modelo también tiene una capacidad discriminante similar para el desempeño temprano. Por otro lado, el modelo que sólo contiene la variable *sub_grade* da un Gini del 32,22%.

3.8 Tarjeta de puntuación

Si bien ya se podría considerar que el modelo está finalizado, es habitual transformar las predicciones de estos modelos a una escala que facilite su entendimiento y uso. Este proceso se suele conocer como escalado o calibración[35], y da como resultado una tarjeta de puntuación o *scorecard*[36].

Algunos beneficios del uso de una escala son:

- Gestión más sencilla y no necesariamente tan granular como con una probabilidad; dificultad para diferenciar entre una probabilidad de malo del 4,50% o del 4,60%, y para establecer puntos de corte.
- Escala familiar para gestión y negocio, sin necesidad de tener tantos conocimientos analíticos.
- La salida del modelo se puede "desescalar/descalibrar", aun manteniendo una ordenación correcta; esto puede pasar si por ejemplo hay una caída/subida fuerte de la tasa de malos de la población total. Se esperaría que el modelo siga ordenando bien en función del riesgo, pero que haya una diferencia entre la tasa de malos observada y esperada.

[35] Para más detalle, ver Siddiqi (2005), capítulo 6, sección *Scaling*.

[36] El uso de las *scorecards* en los procesos de admisión bancarios es previo al uso de modelos estadísticos para su construcción. Se elegían de manera experta ciertas variables que se creían importantes de cara a conceder un crédito a un cliente, y en función del valor que tomara cada una de las variables, se daban más o menos puntos. Tras sumar los puntos de cada una de las variables que formaban la tarjeta de puntuación, se concedía el préstamo en función de si la operación superaba cierto umbral de puntos, como de hecho ocurre en la actualidad en bancos en los que no se asignan precios por riesgo sino por segmentos de clientes, y el punto de corte es común para grupos de operaciones.

- Si hay una gestión con una escala detrás, es más sencillo comprender que la puntuación 100, que tiene una tasa de malos esperada del 10%, está presentando una tasa de malos observada del 9% o del 11%.
- El uso de una escala permite transformar la salida del modelo en una tarjeta de puntuación, que otorga diferentes puntos a cada variable en función del valor que tome. Es posible obtener los puntos que corresponden a cada uno de los valores de las variables. Esto permite hacerse una idea de la importancia de las variables dentro del modelo.
- Si se deniega una operación, es sencillo identificar a qué variable es debido. En algunos entornos regulados, es obligatorio decir a un cliente o regulador por qué se le está denegando un crédito, y hacerlo en base a la capacidad de pago del cliente.
- Es sencillo ver cómo le afectaría a un cliente/solicitud un cambio en una variable, y si se le seguiría aceptando/denegando un préstamo.
- Permite una gestión sencilla de segmentos por puntuaciones. Por ejemplo, para determinar precios para distintos segmentos de operaciones/clientes.

La escala se podría construir definiendo los parámetros a y b de la siguiente relación:

$$\ln(\widehat{odds}) = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) = a + b \cdot \widehat{score} \tag{22}$$

Aunque, en la práctica, se realiza a través de los parámetros PEO y PDO[37]:

- PEO, o *points at even odds*, son los puntos de *score* que hay en los *odds* 1:1, es decir, cuando la tasa de malos es del 50%. Se va a utilizar un parámetro PEO de 600.
- PDO, o *points to double the odds*, son los puntos necesarios para doblar los *odds*. Cuando más grande sea este valor, más granular será la escala construida. Se va a utilizar un parámetro PDO de 20.

Así, se tiene que:

$$\begin{aligned}\widehat{score} &= \frac{\ln(2)}{b} \cdot \frac{\ln(\widehat{odds})}{\ln(2)} - \frac{a}{b} = PDO \cdot \frac{\ln(\widehat{odds})}{\ln(2)} + PEO \\ &= \frac{20}{\ln(2)} \cdot \ln(\widehat{odds}) + 600\end{aligned} \tag{23}$$

Y por tanto, en esa relación original:

$$PDO = 20 = \frac{\ln(2)}{b} \tag{24}$$

[37] De hecho, el software SAS proporciona directamente las puntuaciones que se han obtenido manualmente con R en este proceso una vez se le proporcionan estos dos parámetros, tanto para las operaciones, como la tarjeta de puntuación que distribuye el *score* entre variables y grupos.

$$PEO = 600 = -\frac{a}{b} \tag{25}$$

Dicho de otro modo, la escala que se utilizará será:

$$\ln(\widehat{odds}) = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) = \frac{\ln(2)}{20} \cdot (\widehat{score} - 600) \tag{26}$$

Como la estimación que se tiene para cada observación es sobre la probabilidad de malo p, para cada uno de los registros se genera la variable *score* redondeando al entero la siguiente expresión:

$$\widehat{score} = \frac{20}{\ln(2)} \cdot \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) + 600 \tag{27}$$

Para en cierto modo repartir la puntuación del modelo entre las diferentes variables y sus grupos, es necesario aplicar las siguientes fórmulas:

$$\widehat{score}_{i,J} = \widehat{\beta_i} \cdot WOE_{i,j} \cdot \frac{20}{\ln(2)} + \left(\widehat{\beta_0} \cdot \frac{20}{\ln(2)} + 600\right)/n \tag{28}$$

donde $score_{i,j}$ es la puntuación que le corresponderá al grupo j de la variable i, $\widehat{\beta_i}$ es la estimación del coeficiente de la variable WOE $i, WOE_{i,j}$ es el valor del WOE de la variable i en el grupo j, $\widehat{\beta_0}$ es la estimación del intercepto, y n es el número de variables incluidas en el modelo.

Además, para tener una mejor idea la importancia de las variables en el modelo, se va a calcular la siguiente métrica de peso para cada una de ellas:

$$peso(X_i) = \frac{\widehat{\beta_i} \cdot S_{X_i}}{\sum_{j=1}^{11} \widehat{\beta_j} \cdot S_{X_j}} \tag{29}$$

donde X_i es cada una de las 11 variables WOE que forman parte del modelo, $\widehat{\beta_i}$ la estimación de su coeficiente en el modelo, y S_{X_i} su desviación estándar.

En la siguiente tabla se muestra el modelo final en el formato de *scorecard*:

Variable	Estimación	Grupo	Población	WOE	Tasa de malos	Score	Peso
Intercept	-1,5628	-				-	-
WOE_dti	-0,4577	>32,77, missing	6,2%	-0,49	25,4%	57	7,38%
		<=32,77	7,9%	-0,31	22,3%	55	
		<=28,72	12,0%	-0,21	20,6%	53	
		<=24,49	9,6%	-0,09	18,6%	52	
		<=21,76	19,3%	0,02	17,1%	50	
		<=17,02	17,8%	0,13	15,0%	49	
		<=12,83	27,1%	0,29	13,6%	47	
WOE_tot_hi_cred_lim	-0,5693	>412.370	35,5%	-0,20	20,5%	54	9,20%
		<=412.370	15,6%	-0,13	19,3%	53	
		<=292.718	7,5%	0,01	17,2%	50	
		<=208.511	8,8%	0,08	16,2%	49	
		<=157.924	12,4%	0,20	14,7%	47	
		<=119.456	10,7%	0,29	13,6%	46	
		<=70.370	9,5%	0,48	11,5%	42	
WOE_acc_open_past_24mths	-0,5586	>9	8,3%	-0,57	27,1%	60	12,39%
		<=9	9,0%	-0,33	22,6%	56	
		<=7	7,2%	-0,26	21,4%	55	
		<=6	9,4%	-0,17	19,9%	53	
		<=5	11,8%	-0,06	18,3%	51	
		<=4	13,7%	0,07	16,4%	49	
		<=3	14,5%	0,21	14,5%	47	
		<=2	12,9%	0,34	13,0%	45	
		<=1	13,1%	0,52	11,1%	42	
WOE_bc_open_to_buy	-0,3586	<=158	5,0%	-0,26	21,4%	53	6,17%
		<=4.758	42,0%	-0,16	19,8%	52	
		<=6.504	8,5%	-0,08	18,6%	51	
		<=12.535, missing	18,7%	0,04	16,8%	50	
		<=17.235	7,5%	0,17	15,0%	49	
		<=29.443	9,7%	0,32	13,2%	47	
		>29.443	8,5%	0,64	10,0%	44	
WOE_mths_since_recent_bc	-0,3116	<=2	9,0%	-0,26	21,4%	53	4,42%
		<=3	5,3%	-0,18	20,1%	52	
		<=10	27,8%	-0,12	19,2%	52	
		<=14	10,8%	-0,06	18,2%	51	
		<=20	11,5%	0,01	17,3%	50	
		<=28, missing	11,2%	0,09	16,1%	50	
		<=57	14,6%	0,20	14,7%	49	
		>57	9,8%	0,48	11,5%	46	
WOE_mths_since_recent_inq	-0,3515	<=1	18,4%	-0,27	21,6%	53	5,36%
		<=2	8,2%	-0,17	20,0%	52	
		<=3	7,6%	-0,12	19,1%	52	
		<=6	17,7%	-0,04	18,0%	51	
		<=8	9,1%	0,01	17,2%	50	
		<=10	7,0%	0,07	16,4%	50	
		<=12	5,5%	0,12	15,7%	49	
		<=18	10,7%	0,19	14,8%	48	
		>18	5,4%	0,29	13,5%	47	
		Missing	10,5%	0,45	11,8%	46	
WOE_term	-0,3483	60 months	25,5%	-0,33	22,7%	54	4,92%
		36 months	74,5%	0,13	15,6%	49	
WOE_grade	-0,7401	E, F, G	8,7%	-1,05	37,6%	73	34,54%
		D	13,6%	-0,57	27,2%	63	
		C	30,4%	-0,12	19,1%	53	
		B	31,0%	0,44	11,9%	41	
		A	16,3%	1,26	5,6%	23	
WOE_home_ownership	-0,5004	RENT	39,0%	-0,21	20,6%	53	6,54%
		OWN	12,3%	-0,01	17,4%	51	
		MORTGAGE, ANY	48,7%	0,19	15,4%	48	
WOE_verification_status	-0,3129	Verified	29,0%	-0,28	21,8%	53	5,43%
		Source Verified	41,7%	-0,01	17,5%	51	
		Not Verified	29,3%	0,37	12,7%	47	
WOE_max_bal_bc	-0,2891	<=2.564, missing	27,8%	-0,18	20,0%	52	3,65%
		<=5.090	30,6%	-0,06	18,3%	51	
		<=6.517	11,0%	0,03	16,9%	50	
		<=8449	10,2%	0,11	15,8%	50	
		<=11.861	9,8%	0,22	14,4%	49	
		>11.861	10,5%	0,40	12,3%	47	

Tabla 17: *Scorecard* del modelo final

Las siguientes gráficas ayudan a entender la distribución del modelo (de la probabilidad inferida por el modelo una vez es transformada a *scores*, la escala presentada al principio de esta sección) en la muestra *train*: en primer lugar, aparece la distribución del *score* (porcentaje de población por punto de *score*); a continuación aparece la distribución del *score* pero diferenciando entre buenos y malos, lo que permite observar cómo el modelo separa estas dos poblaciones; por último, y con el mismo fin, aparecen las distribuciones acumuladas de buenos y malos.

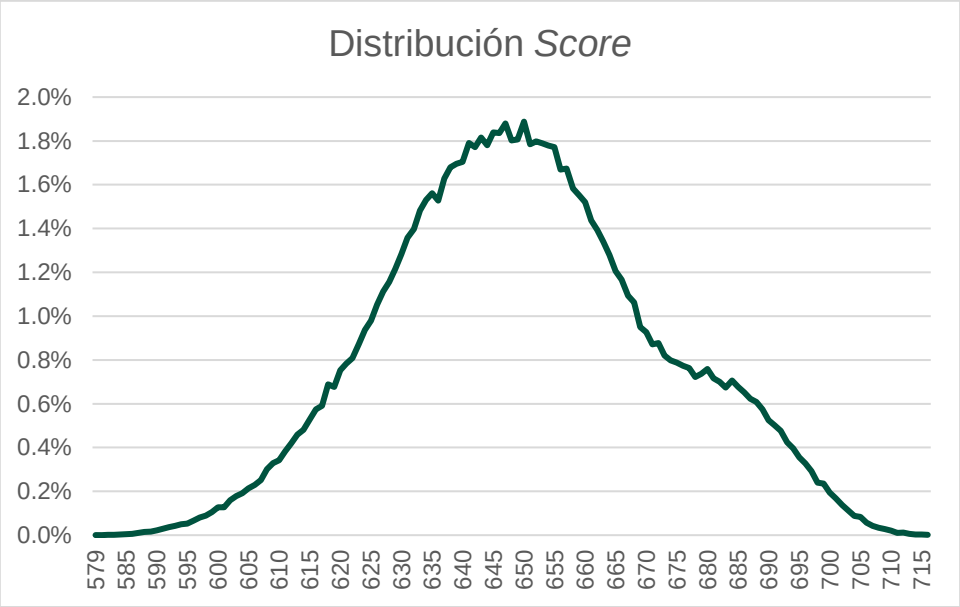

Figura 16: Distribución *score* en muestra *train*

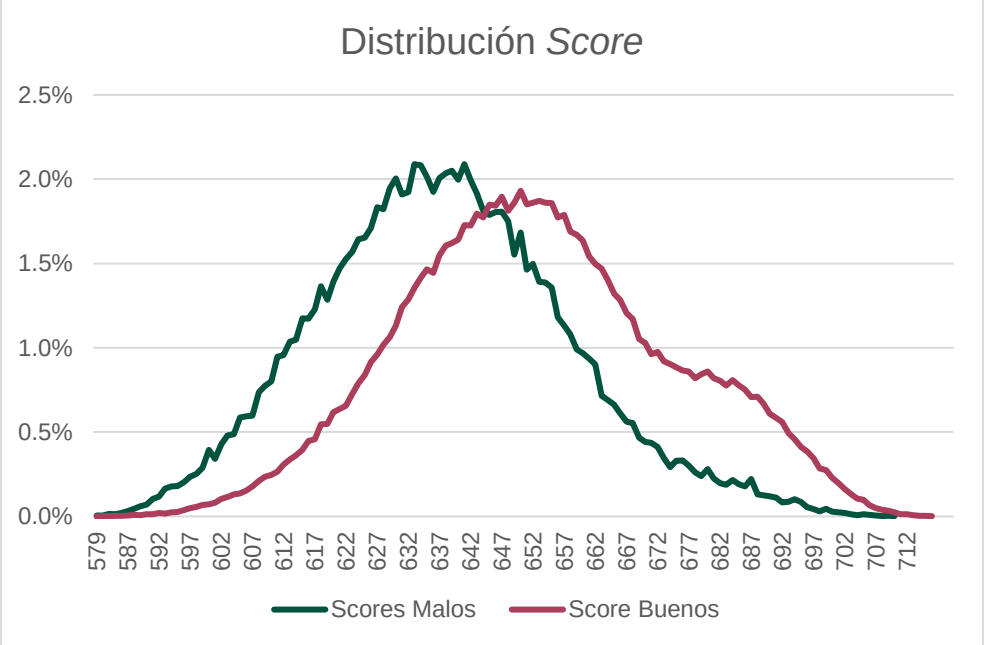

Figura 17: Distribución *score* buenos vs *score* malos en muestra *train*

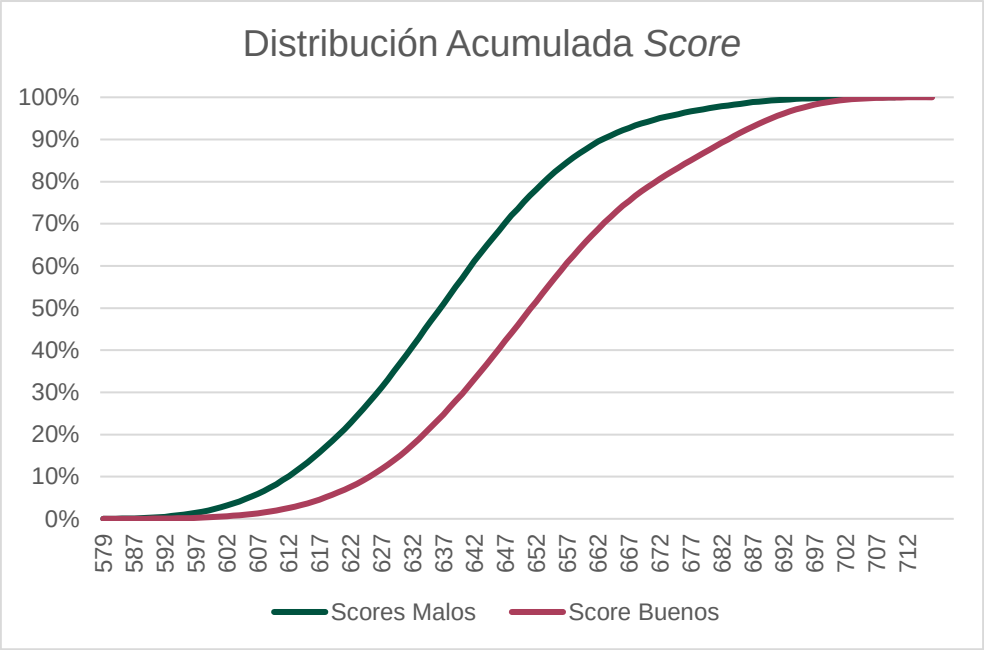

Figura 18: Distribución acumulada *score* buenos vs *score* malos en muestra *train*

De esta gráfica, por ejemplo, se podría obtener otra de las métricas de discriminación más populares, el estadístico Kolmogorov - Smirnov o KS, que es el valor máximo de la diferencia entre las dos distribuciones acumuladas a lo largo de los posibles puntos de *score*. Gráficamente, sería la máxima distancia vertical entre ambas frecuencias acumuladas. En este caso, toma el valor 28,2%, en la puntuación 644.

3.9 Enfoque utilizando técnicas de *Machine Learning*

Este apartado no pretende hacer un análisis tan exhaustivo como para el modelo de regresión logística, sino hacer una breve presentación de las técnicas más utilizadas y ver si se pueden construir modelos con niveles de discriminación similares a los de la regresión logística, así como presentar los problemas de interpretación asociados a ellas. Para más detalles teóricos sobre cada una de las técnicas, o justificaciones de las principales propiedades que se expongan, se puede consultar Hastie, Tibshirani y Friedman (2008), o James, Hastie, Witten y Tibshirani (2013).

Para las técnicas presentadas, se van a intentar construir modelos tanto con variables WOE como con variables crudas (antes de hacer la transformación a WOE). Las variables candidatas utilizadas van a ser las 33 variables candidatas sobre las que se han construido las regresiones logísticas. Las estimaciones de estos modelos también serán la probabilidad de pertenecer a la categoría de malos. La mayoría de los algoritmos requieren un tratamiento adicional para puntuar nuevas muestras si se han construido con variables crudas, ya que hay que decidir cómo puntuar a observaciones que presentan una variable a *missing* o con un valor que no aparecía en la muestra de construcción (en el caso de variables categóricas); en cambio, las variables WOE lidian con este problema asignando un valor WOE de cero a esas observaciones. Por la extensión del presente trabajo, no se va a hacer el tratamiento adicional que requerirían los modelos con variables crudas para algunos algoritmos.

Es necesario también señalar la importancia de los hiperparámetros o parametrizaciones utilizadas en las construcciones de estos modelos. Por ejemplo, en

los modelos *random forest*, la profundidad de los árboles será fundamental, porque dará lugar a modelos muy sobreajustados a la muestra de entrenamiento si no se controla; mientras que en los modelos de redes neuronales el número de capas ocultas es un parámetro fundamental que puede dar lugar a modelos muy diferentes. El problema asociado a la elección de los hiperparámetros es que no existen unas parametrizaciones que sean "las mejores" para todos los problemas o conjuntos de datos, por lo que es necesaria una exploración de distintas combinaciones a través de un *grid*, que simula en paralelo muchos modelos hasta elegir uno de acuerdo a una función objetivo (por ejemplo, el mayor Gini en la muestra *test*). Dichas simulaciones se pueden hacer también de manera secuencial, pero consumirían una cantidad ingente de tiempo, y dado que no se dispone de un servidor distribuido en el que construir, por ejemplo, 1.000 modelos a la vez, se van a elegir hiperparámetros de manera "experta" para mostrar algunos ejemplos de modelos utilizando estas técnicas. Para más detalle sobre técnicas de exploración de hiperparámetros, se puede consultar Bengio y Bergstra (2012), o Claesen y De Moor (2015).

Por último, estas técnicas no sólo son útiles para la propia construcción de los modelos, sino que también pueden ayudar en la generación de las categorías para transformar las variables a WOE, o en la búsqueda de segmentaciones en la población; es decir, en la generación de subpoblaciones en las que construir modelos diferentes. Estos dos enfoques no serán tratados en este trabajo.

Árbol de regresión[38]

Los árboles de regresión, decisión o clasificación (cuando la variable independiente es categórica) son modelos que van generando cortes en las variables candidatas para crear subpoblaciones con comportamientos similares, utilizando alguna métrica de reducción del error o de poder discriminante para determinar el orden en el que las variables entran en el modelo. Nótese que es posible que entre la misma variable en el modelo más de una vez en distintas ramas o niveles de profundidad. A cada observación se le asigna la probabilidad de malo inferida de acuerdo a la tasa de malos que hay en el nodo final al que pertenece.

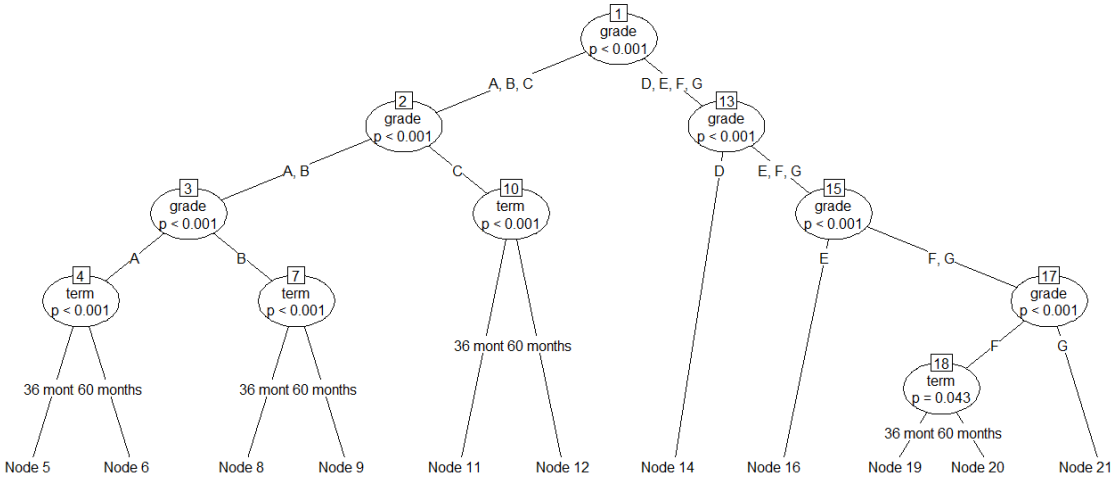

Figura 19: Ejemplo de árbol de decisión utilizando las variables *term* y *grade*

[38] Se ha utilizado el algoritmo *ctree* del paquete *partykit* de R.

El modelo construido con variables crudas tiene un Gini sobre la muestra *train* del 39,06%, aunque no es posible puntuar sin tratamiento adicional el resto de muestras. En cambio, el modelo construido con variables WOE sí que permite puntuar todas las muestras, y presenta valores Gini del 39,13% en la muestra *train,* del 37,25% en la muestra *test*, 36,88% en OOT y 37,66% en TTD. Estos valores son similares, aunque algo peores, que los obtenidos con el modelo de regresión logística, y mejor significativamente que los niveles de discriminación del modelo en vigor (variable *sub_grade*).

Los árboles presentan una ventaja respecto a otros algoritmos, y es que son más sencillos de entender y es posible sacar alguna métrica de importancia de las variables dentro de los árboles. En este caso, se va a mostrar un ejemplo en el que se utiliza, para el árbol sobre variables WOE, la reducción media de la precisión (*accuracy*) del modelo (medida utilizando el Gini) cuando esa variable sufre permutaciones aleatorias en las observaciones. Además, se estandariza esta variable dividiendo entre la suma los valores medios de cada variable, para obtener una métrica de importancia relativa. Destaca que en este modelo han entrado 29 de las 33 variables candidatas, pero la importancia de *grade* es del 52%, frente a algunas variables que se encuentran por debajo del 1%.

Variable	Reducción	Importancia relativa	Variable	Reducción	Importancia relativa	Variable	Reducción	Importancia relativa
sub_grade	0,044%	2,916%	total_bc_limit	-	-	total_rev_hi_lim	0,002%	0,152%
grade	0,780%	52,269%	tot_hi_cred_lim	0,102%	6,831%	mort_acc	0,006%	0,386%
int_rate	0,217%	14,546%	mths_since_recent_inq	0,004%	0,268%	mo_sin_rcnt_rev_tl_op	0,001%	0,039%
acc_open_past_24mths	0,162%	10,861%	mo_sin_rcnt_tl	0,000%	0,017%	mths_since_recent_bc	-	-
open_rv_24m	0,015%	0,975%	open_rv_12m	-	-	home_ownership	0,022%	1,498%
num_tl_op_past_12m	0,003%	0,193%	open_acc_6m	0,002%	0,130%	il_util	-	-
verification_status	0,004%	0,285%	open_il_24m	0,000%	0,010%	all_util	0,002%	0,104%
dti	0,045%	2,998%	open_il_12m	0,001%	0,070%	annual_inc	0,001%	0,055%
bc_open_to_buy	0,008%	0,552%	term	0,004%	0,283%	mo_sin_old_rev_tl_op	0,001%	0,070%
inq_last_12m	0,013%	0,846%	mths_since_rcnt_il	0,001%	0,098%	max_bal_bc	0,007%	0,442%
avg_cur_bal	0,039%	2,647%	tot_cur_bal	0,003%	0,226%	inq_fi	0,003%	0,233%

Tabla 18: Importancia de las variables WOE incluidas en el árbol de regresión

Random forest[39]

Los modelos *random forest* o bosques aleatorios están basados en la construcción de un conjunto de árboles de decisión. Para la construcción de cada uno de estos árboles sólo se utiliza una parte de las observaciones y variables. Para realizar la inferencia, a cada observación se le asigna la probabilidad de malo media que le correspondería por cada uno de los árboles que forman parte del *random forest*. La principal ventaja frente a los árboles de decisión es que son más robustos a cambios en los conjuntos de entrenamiento, aunque se debe vigilar la profundidad de los árboles (cuántas variables y cortes se realizan) para evitar grandes sobreajustes. Algunos algoritmos permiten controlar directamente la profundidad de los árboles, mientras que otros controlan el número total de nodos o el número mínimo de observaciones en nodos finales. El resto de parametrizaciones se van a dejar en: 100 árboles y sólo 6 (raíz del número de variables candidatas totales) variables candidatas por árbol.

[39] Se ha utilizado el algoritmo *randomForest* del paquete *randomForest* de R.

De nuevo, si se construye el modelo sobre variables crudas, necesitaría de un tratamiento manual adicional sobre el resto de muestras para poder puntuarlas, por lo que sólo se va a trabajar sobre el caso de variables WOE.

Si no se controla la profundidad del árbol, el resultado sobre variables WOE arrojaría un modelo con un Gini del 99,97% en *train*, 35,55% en *test*, 35,73% en OOT y *36,27%* en TTD, lo que muestra el gran nivel de sobreajuste a *train* que se está produciendo. Si se construye sobre variables crudas, el modelo tendría igualmente un 99,95% de Gini sobre la muestra *train*.

El algoritmo que se va a utilizado permite controlar indirectamente la profundidad mediante el número total de nodos o el número mínimo de observaciones en nodos finales. Si por ejemplo se limita el número máximo de nodos a 2.000, se obtendría un modelo con un Gini del 54,30% en *train*, 25,84% en *test*, 24,71% en OOT y 24,32% en TTD, que reduce el sobreajuste, pero no alcanza los niveles de discriminación del árbol de regresión ni los del modelo vigente. En cambio, si se limita el número mínimo de observaciones en nodos finales a 50, se consigue un modelo con un Gini del 61,37% en *train*, 30,59% en *test*, 29,95% en OOT y 30,11% en TTD, que reduce parte del sobreajuste, y, aunque no alcanza el nivel de discriminación del árbol de regresión, sí que supera al modelo en vigor (excepto en la muestra TTD, que en realidad sólo permite medir el desempeño temprano).

Si se dispusiera de un *grid* que permitiera simular muchas parametrizaciones distintas, lo esperable sería que se mejoraran los resultados del árbol de regresión, lo que daría resultados similares o incluso mejores que para la regresión logística. En cualquier caso, aunque no se haya realizado un *grid*, se han utilizado diversas parametrizaciones y se muestran únicamente las que han arrojado mejores resultados.

Nuevamente, se puede calcular la importancia de las variables utilizando (y estandarizando) la pérdida media de Gini al aplicar permutaciones a cada variable. La principal diferencia en los pesos respecto al árbol de clasificación o la regresión logística es que, por forzar a que en cada uno de los árboles entren sólo hasta 6 variables, el peso está mucho más repartido. A continuación se muestran los pesos de las variables en el caso construido con variables WOE:

Variable	Importancia relativa	Variable	Importancia relativa	Variable	Importancia relativa
sub_grade	2,876%	total_bc_limit	3,264%	total_rev_hi_lim	3,475%
grade	3,447%	tot_hi_cred_lim	2,471%	mort_acc	2,870%
int_rate	3,418%	mths_since_recent_inq	4,614%	mo_sin_rcnt_rev_tl_op	3,794%
acc_open_past_24mths	3,506%	mo_sin_rcnt_tl	3,568%	mths_since_recent_bc	3,908%
open_rv_24m	3,234%	open_rv_12m	2,017%	home_ownership	1,705%
num_tl_op_past_12m	2,677%	open_acc_6m	2,005%	il_util	3,271%
verification_status	2,173%	open_il_24m	2,776%	all_util	3,996%
dti	4,155%	open_il_12m	1,269%	annual_inc	4,126%
bc_open_to_buy	3,354%	term	1,156%	mo_sin_old_rev_tl_op	3,517%
inq_last_12m	3,722%	mths_since_rcnt_il	3,359%	max_bal_bc	3,282%
avg_cur_bal	2,699%	tot_cur_bal	1,332%	inq_fi	2,966%

Tabla 19: Importancia de las variables WOE incluidas en el *random forest*

Árboles Adaboost[40]

Los árboles iterados, *gradient boosting*, *adapting boosting* o árboles *adaboost* son árboles de decisión que se van iterando sobreponderando los registros en los que el árbol anterior cometía un error mayor, para corregir ese error en los entrenamientos de árboles sucesivos. Además, se utiliza una penalización (*shrinkage*) para que el aprendizaje sea lento y no se sobreajuste el modelo. El modelo final será similar a un *random forest*, pero calculando la media ponderada de cada uno de los modelos construidos. Se trata de modelos que, en general, producen menos sobreajuste que los árboles o *random forest*, pero que requieren de más tiempo de ejecución.

Los hiperparámetros utilizados han sido número de árboles a iterar (100 y 200), profundidad máxima de los árboles (2, 3, 4, 5 y 6), y velocidad de aprendizaje o *shrinkage* (0,5, 0,2, 0,1, 0,05, 0,01, 0,001). De las posibles combinaciones, el mejor modelo obtenido sobre variables WOE, parametrizado con 200 árboles, una profundidad máxima de 5, y un aprendizaje del 0,2, arroja un Gini del 41,42% en la muestra *train*, un 39,03% en *test*, 39,03% en OOT, y 39,43% en TTD. Este modelo superaría a la regresión logística en todas las ventanas. Además, el algoritmo empleado incorpora un tratamiento para valores nuevos o *missing* en el caso de variables crudas, por lo que es posible construir este modelo con variables crudas y realizar inferencia en el resto de muestras: se obtiene un Gini del 42,13% en *train* y del 39,11% en *test*, con lo que superaría al modelo con variables WOE, pero al realizar la inferencia en las otras dos ventanas encuentra más valores nuevos o *missing* que no estaban incluidos en la muestra de construcción, por lo que el poder discriminante baja hasta un Gini del 33,92% en OOT y del 33,22% en TTD.

De nuevo, al tratarse de modelos basados en árboles, es posible calcular métricas de importancia relativa a partir de la pérdida de precisión del modelo cuando se realizan permutaciones aleatorias de los valores de la variable en cuestión. La influencia relativa para el modelo que contiene las variables WOE es la siguiente:

Variable	Importancia relativa	Variable	Importancia relativa	Variable	Importancia relativa
sub_grade	9,198%	total_bc_limit	1,050%	total_rev_hi_lim	0,618%
grade	27,786%	tot_hi_cred_lim	1,691%	mort_acc	1,039%
int_rate	25,645%	mths_since_recent_inq	1,191%	mo_sin_rcnt_rev_tl_op	0,665%
acc_open_past_24mths	5,575%	mo_sin_rcnt_tl	0,614%	mths_since_recent_bc	1,273%
open_rv_24m	2,448%	open_rv_12m	0,300%	home_ownership	2,255%
num_tl_op_past_12m	0,634%	open_acc_6m	0,394%	il_util	0,700%
verification_status	1,214%	open_il_24m	0,627%	all_util	0,660%
dti	2,762%	open_il_12m	0,490%	annual_inc	1,271%
bc_open_to_buy	1,045%	term	1,001%	mo_sin_old_rev_tl_op	0,999%
inq_last_12m	1,352%	mths_since_rcnt_il	0,824%	max_bal_bc	0,928%
avg_cur_bal	2,500%	tot_cur_bal	0,237%	inq_fi	1,014%

Tabla 20: Importancia de las variables WOE incluidas en el *adaboost*

[40] Se ha utilizado el algoritmo *gbm* del paquete *gbm* de R.

Árboles XGBoost[41]

Los *XGBoost* o *extreme gradient boosting* son una implementación de árboles iterados que optimizan la reducción del error en árboles sucesivos respecto a los *adaboost* y que reducen considerablemente el tiempo de ejecución al aprovechar los núcleos de los procesadores y estar programados sobre matrices numéricas en lugar de sobre todo tipo de datos. Esto permite una mayor exploración de hiperparámetros, aunque el precio a pagar es tener que transformar las tablas a matrices con valores numéricos, por lo que es necesario estimar el modelo sobre variables WOE (u otra transformación a variables numéricas).

Este algoritmo es relativamente nuevo y no aparece en los manuales más conocidos de *machine learning*, pero se ha convertido en un algoritmo popular por haber resultado ganador en diversas competiciones de Kaggel[42]. Para consultar detalles teóricos de este modelo, se puede ver la documentación de la función y librería en el Anexo 4, o Chen y Guestrin (2016).

Las parametrizaciones utilizadas han sido el número de árboles iterados (10, 20, 25, 50, 100, 200), la profundidad (2, 3, 4, 5, 6, 7) y la velocidad de aprendizaje (0,1, 0,2, 0,25, 0,3, 0,4. 0,5, 0,75). El mejor modelo ha sido obtenido con los parámetros 100, 3 y 0,25 respectivamente, y cuenta con un Gini del 41,23% en la muestra *train*, un 39,22% en *test*, 39,25% en OOT, y 39,73% en TTD, por lo que supera por poco al mejor modelo *adaboost* construido, y por tanto a la regresión logística y al modelo vigente en Lending Club (variable *sub_grade*). La importancia de las variables incluidas, calculada como en un *adaboost*, se recoge en la siguiente tabla:

Variable	Importancia relativa	Variable	Importancia relativa	Variable	Importancia relativa
sub_grade	14,617%	total_bc_limit	0,875%	total_rev_hi_lim	0,235%
grade	26,106%	tot_hi_cred_lim	2,268%	mort_acc	0,944%
int_rate	29,472%	mths_since_recent_inq	0,993%	mo_sin_rcnt_rev_tl_op	0,839%
acc_open_past_24mths	6,042%	mo_sin_rcnt_tl	0,254%	mths_since_recent_bc	1,001%
open_rv_24m	1,507%	open_rv_12m	0,105%	home_ownership	2,166%
num_tl_op_past_12m	0,370%	open_acc_6m	0,133%	il_util	0,618%
verification_status	1,064%	open_il_24m	0,185%	all_util	0,520%
dti	2,424%	open_il_12m	0,179%	annual_inc	0,695%
bc_open_to_buy	1,072%	term	0,701%	mo_sin_old_rev_tl_op	0,324%
inq_last_12m	0,962%	mths_since_rcnt_il	0,428%	max_bal_bc	0,536%
avg_cur_bal	1,716%	tot_cur_bal	0,187%	inq_fi	0,462%

Tabla 21: Importancia de las variables WOE incluidas en el *xgboost*

3.10 Debilidades

Como principales debilidades del trabajo realizado, destacan:

- El hecho de no poder incorporar información de denegados al modelo, ya que la información almacenada por Lending Club respecto a estas operaciones es distinta a la de las operaciones admitidas, así como muy limitada. Este hecho

[41] Se ha utilizado el algoritmo *xgboost* del paquete *xgboost* de R.

[42] De acuerdo a https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions

hace que el modelo construido presente un sesgo hacia la población de operaciones admitidas, y además limita ejercicios de optimización de ingresos o de rentabilidad en la cartera.

- El cambio estructural en el contexto macroeconómico, que podría afectar a la adecuación del modelo, por los cambios que se esperan en la población que evaluaría.

3.11 *Future work*

Como posibles ampliaciones del presente trabajo, destacan:

- Estudiar posibles segmentaciones en el desarrollo del modelo de *scoring*, por ejemplo, utilizando la variable *term* (plazo del préstamo) o *purpose* (finalidad del crédito), y desarrollar modelos independientes para las subpoblaciones definidas por estas variables. La gran volumetría de observaciones permitiría construir modelos robustos a nivel subpoblación.
- Estudiar la explicabilidad y la especificación de los modelos construidos utilizando técnicas de *machine learning*.
- Explorar más parametrizaciones de modelos de *machine learning* mediante el uso de *grids*.
- Construir modelos utilizando las técnicas SVM y redes neuronales. Estos modelos han sido probados en este trabajo, pero requerían de un mayor coste computacional y no ha sido posible obtener resultados satisfactorios. Al final de este apartado se proporciona una breve introducción de estas dos técnicas.
- Intentar extraer información de las variables de texto libre; por ejemplo, utilizando la librería *word2vec* de Python, que transforma los textos en vectores numéricos y permite generar una variable tipo *score* a partir de ellos.
- Explorar el uso de técnicas de *machine learning* en problemas econométricos distintos a los de clasificación. Por ejemplo, sería interesante construir un modelo para describir una serie temporal utilizando la metodología Box-Jenkins[43] y comparar los resultados con un modelo de redes neuronales recursivas (RNN).

SVM

Las máquinas de soporte vectorial o SVM (*support vector machine*) son modelos en los que se realiza una transformación del espacio de observaciones (a través de una función *kernel*) de tal modo que se maximice la distancia entre las dos clases (buenos y malos), que son separadas por un hiperplano. En general, se trata de un algoritmo sencillo de interpretar y manejar cuando el número de variables es muy reducido (hasta 3 variables permitiría incluso visualizar la transformación y separación del espacio), pero complejo a medida que el número de variables incrementa.

Redes Neuronales

Reciben este nombre por su similitud con las redes neuronales humanas, ya que se estructuran en capas de neuronas que transmiten impulsos iniciales (información, desde las variables de entrada) a capas de neuronas intermedias (capas ocultas) que se

[43] Para más detalle, ver Box y Jenkins (1970).

activan o no (funciones de activación), y transmiten más impulsos a otras capas de neuronas hasta llegar a la última neurona, que sería la variable de salida.

4. Conclusiones

La regresión logística construida mejora significativamente la capacidad discriminante del modelo que publica Lending Club. La mejora respecto a dicho modelo es de un 40% relativo en la muestra *test* y un 36% relativo en la muestra OOT.

Además, incluso considerando que no se ha hecho una exploración intensiva de sus parametrizaciones, algunos de los modelos que utilizan técnicas de *machine learning* alcanzan niveles de discriminación similares e incluso mejores que la regresión logística, aunque es más complicado entender su funcionamiento y resultado. Si se pudieran explorar nuevas fuentes de datos, o tratar datos que han tenido que ser descartados por su formato (por ejemplo, variables de texto libre), se esperaría que el uso de estas técnicas diera mejores resultados. También es interesante destacar que estos modelos, a pesar de lo que se piensa, no han arrojado resultados significativamente mejores a los de la regresión logística cuando han sido entrenados con la misma cantidad de información.

	Poder discriminante			Indicador temprano de poder discriminante
Modelo	**Train**	**Test**	**OOT**	**TTD**
Modelo vigente (variable WOE subgrade)	27,70%	27,38%	27,99%	32,22%
Regresión logística, 11 variables WOE	**39,11%**	**38,27%**	**38,06%**	**39,19%**
Árbol de regresión, variables crudas	39,06%	-	-	-
Árbol de regresión, variables WOE	39,13%	37,25%	36,88%	37,66%
Random forest, variables crudas	99,95%	-	-	-
Random forest, variables WOE	99,97%	35,55%	35,73%	36,27%
Random forest, variables WOE, máx 2.000 nodos	54,30%	25,84%	24,71%	24,32%
Random forest, variables WOE, mín 50 observs.	61,37%	30,59%	29,95%	30,11%
Árboles adaboost, variables crudas	42,13%	39,11%	33,92%	33,22%
Árboles adaboost, variables WOE	41,42%	39,03%	39,03%	39,43%
XGBoost, variables WOE	41,23%	39,22%	39,25%	39,73%

Tabla 22: Resultados de los principales modelos construidos y del modelo en vigor

5. Bibliografía

Altman, E. I. y Saunders, A. (1998). *Credit risk measurement: Developments over the last 20 years*, en Journal of Banking & Finance, vol. 21, issue 11-12, 1721-1742.

Anderson, B., Haller, S. y Siddiqi, N. (2009). *Reject Inference Techniques Implemented in Credit Scoring for SAS Enterprise Miner*, en SAS Global Forum, Paper 305-2009.

Bengio, Y. y Bergstra, J. (2012). *Random Search for Hyper-Parameter Optimization*, en Journal of Machine Learning Research 13, 281-305.

Berkson, J. (1944). *Application of the Logistic Function to Bio-Assay*, en Journal of the American Statistical Association Vol. 39, No. 227, pp. 357-365.

Box, G. E. P. y Jenkins, G. M. (1970). *Time Series Analysis, Forecasting and Control*. Holden Day. San Francisco.

Chen, T. y Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*, en Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 13-17.

Claesen, M. y De Moor, B. (2015). *Hyperparameter Search in Machine Learning*, en MIC 2015: The XI Metaheuristics International Conference.

Cocea, M. y Liu, H. (2007). *Semi-random partitioning of data into training and test sets in granular computing context*, en Springer, Granul. Comput. 2, 357–386.

Cramer, J. S. (2004). *The early origins of the logit model*, en Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences, Volume 35, Issue 4, December 2004, Pages 613-626.

Davidson, R. y MacKinnon, J. G. (2004). *Econometric Theory and Methods*, Oxford University Press.

EBA (2016). *Guidelines on the application of the definition of default under Article 178 of Regulation (EU) No 575/2013*, European Banking Authority, EBA/GL/2016/07.

EBA (2020). *Guidelines on legislative and non-legislative moratoria on loan repayments applied in the light of the COVID-19 crisis*, European Banking Authority, EBA/GL/2020/02.

Eijkemans, M. J., Habbema, J. D., Harrell, F. E. y Steyerberg, E. W. (2000). *Prognostic modelling with logistic regression analysis: a comparison of selection and estimation methods in small data sets*, en Statist. Med., 19: 1059-1079.

Friedman, J. H. (1991). *Multivariate Adaptive Regression Splies*, The Annals of Statistics, 1991, Vol.19, No. 1, 1-141.

Furnival, G. M. y Wilson, R. W. (2000). *Regressions by leaps and bounds*, en Technometrics 42, 69-79, 10.2307/1271435.

Greene, W. H. (2003). *Econometric Analysis*, 5th ed. Pearson Education LTD.

Gujarati, D. N. y Porter, D. C. (2010). *Econometría*, 5ª ed. McGraw Hill.

Fernández, A. (2019). *Inteligencia artificial en los servicios financieros*, en Boletín Económico 2/2019, Artículos Analíticos, Banco de España.

Gambacorta, L., Huang, Y., Qiu, H. y Wang, J. (2019). *How do machine learning and non-traditional data affect credit scoring? New evidence from a Chinese fintech firm*, en BIS Working Papers, No 834.

Hand, D. J. y Till, R. J. (2001). *A simple generalization of the area under the ROC curve for multiple class classification problems*, Machine Learning, 45, 171–186.

Harrell, F. E., Lee, K. L. y Mark, D. B. (1996). *Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors*, en Statist. Med., 15(4):361-87.

Hastie, T., Tibshirani, R. y Friedman, J. (2008). *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. 2nd ed. Springer.

Heiberger, R. M. y Holland, B. (2015). *Statistical Analysis and Data Display. An Intermediate Course with Examples in R*, 2nd ed. Springer Texts in Statistics book series, Springer.

Heumann, C. y Shalabh, M. S. (2016). *Introduction to Statistics and Data Analysis. With Exercises, Solutions and Applications in R*, Springer.

Ieno, E. N., Meesters, E. H. W. G. y Zuur, A. F. (2009). *A Beginner's Guide to R*, Use R! book series, Springer.

James, G., Hastie, T., Witten, D. y Tibshirani, R. (2013). *An Introduction to Statistical Learning. With Applications in R*, Springer.

Johnson, R. A. y Wichern, D. W. (2007). *Applied Multivariate Statistical Analysis*, 6th ed. Pearson Education LTD.

Matilla, M., Pérez, P. y Sanz, B. (2013). *Econometría y Predicción*, McGraw-Hill.

Pastor, I. (2019). *Machine Learning with R Quick Start Guide. A beginner's guide to implementing machine learning techniques from scratch using R 3.5,* Packt.

Peña, D. (2002). *Análisis de Datos Multivariantes*, McGraw Hill.

Peña, D. (2010). *Regresión y Diseño de Experimentos*, Alianza Editorial.

Rezac, F. y Rezac, M. (2011). *How to Measure the Quality of Credit Scoring Models,* en Czech Journal of Economics and Finance (Finance a uver), 61, 486-507.

Siddiqi, N. (2005). *Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring*, SAS Institute, Wiley and Sons

Siddiqi, N. (2017). *Intelligent Credit Scoring: Building and Implementing Better Credit Risk Scorecards*, SAS Institute, Wiley and Sons.

Skiena, S. S. (2017). *The Data Science Design Manual*, Text in Computer Science, Springer.

Smeeton, N. C. y Sprent, P. (2001). *Applied Nonparametric Statistical Methods*, Chapman & Hall/CRC.

Stock, J. H. y Watson, M. W. (2007). *Introduction to Econometrics*, 2nd ed. Adison-Wesley series in Economics.

Thomas, L. C. (2000). *A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers*, en International Journal of Forecasting, Volume 16, Issue 2, Pages 149-172.

Wooldrige, J. M. (2006). *Introducción a la econometría. Un enfoque moderno*, Thomson - Paraninfo.

6. Anexos

6.1 Anexo 1: Diccionario de datos

LCDataDictionary.xlsx

6.2 Anexo 2: Significado variables incluidas en el modelo 9 (final)

Variable	Significado
dti	Ratio *debt to income:* deudas mensuales entre ingresos mensuales (porcentaje de ingresos dedicado a pagar deudas), excluyendo deudas por hipotécas y la que generaría el nuevo préstamo
tot_hi_cred_lim	Importe de créditos totales entre límite de crédito asociado a ese cliente
acc_open_past_24mths	Número de cuentas bancarias abiertas en los últimos 24 meses
bc_open_to_buy	Número de tarjetas de crédito
mths_since_recent_bc	Meses desde la última cuenta de tarjeta de crédito abierta
mths_since_recent_inq	Meses desde el último impago en una tarjeta de crédito
term	Plazo de la operación (36 o 60 meses)
grade	Calificación interna dada por Lending Club; existen sólo 7 calificaciones, que en otra variable (*subgrade*) son más granulares (5 subgrupos por cada calificación)
home_ownership	Estado que indica en qué medida el solicitante es el propietario de su casa: alquiler, propietario, hipoteca, u otros
verification_status	Variable que indica si los ingresos que ha declarado el solicitante han sido verificados por Lending Club; si los ingresos no han sido verificados, pero sí que se ha verificado la fuente de ingresos; o si no se ha verificado nada
max_bal_bc	Deuda máxima entre todas sus tarjetas o líneas de crédito

Tabla 23: Significado de las variables que componen el modelo final

6.3 Anexo 3: Proyecto RStudio

A continuación se adjunta el proyecto de RStudio, aunque en formato de texto plano, debido a que el proyecto contiene datos cargados y un peso considerable para ser adjuntado. Nótese que serán necesarios pequeños cambios para poder reproducir los resultados de este trabajo, como cambiar las rutas de carga y descarga de datos, así como instalar todas las librerías de las que se hace uso (el comando para instalarlas aparece en código comentado).

Proyecto.txt

6.4 Anexo 4: Documentación Librerías R

Debido al uso de software libre, se hace necesario especificar tanto la versión de R utilizada, R 3.5.2, como la documentación de las librerías utilizadas. Algunas de estas librerías podrían no estar disponibles para versiones anteriores de R, y algunas de ellas podrían evolucionar con el tiempo, por ejemplo añadiendo nuevas funcionalidades, necesitando ciertas adaptaciones en el código. En el código del proyecto se cargan cada una de estas librerías y se explica por qué son necesarias (por ejemplo, la librería *ggplot2* se utiliza para generar gráficos). Si, por ejemplo, se necesita comprobar cómo se ha calculado el *information value* de una variable, sería necesario ir a la documentación del paquete *smbinning* y leer los detalles técnicos de la función *smbinning.sumiv.*

- https://cran.r-project.org/web/packages/dplyr/dplyr.pdf
- https://cran.r-project.org/web/packages/RSQLite/RSQLite.pdf
- https://cran.r-project.org/web/packages/proto/proto.pdf
- https://cran.r-project.org/web/packages/gsubfn/gsubfn.pdf
- https://cran.r-project.org/web/packages/sqldf/sqldf.pdf
- https://cran.r-project.org/web/packages/descr/descr.pdf
- https://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf
- https://cran.r-project.org/web/packages/plotly/plotly.pdf
- https://cran.r-project.org/web/packages/stringr/stringr.pdf
- https://cran.r-project.org/web/packages/fBasics/fBasics.pdf
- https://cran.r-project.org/web/packages/data.table/data.table.pdf
- https://cran.r-project.org/web/packages/rlang/rlang.pdf
- https://cran.r-project.org/web/packages/scales/scales.pdf
- https://cran.r-project.org/web/packages/DataExplorer/DataExplorer.pdf
- https://cran.r-project.org/web/packages/psych/psych.pdf
- https://cran.r-project.org/web/packages/caTools/caTools.pdf
- https://cran.r-project.org/web/packages/smbinning/smbinning.pdf
- https://cran.r-project.org/web/packages/corrplot/corrplot.pdf
- https://cran.r-project.org/web/packages/vcd/vcd.pdf
- https://cran.r-project.org/web/packages/ROCR/ROCR.pdf
- https://cran.r-project.org/web/packages/randomForest/randomForest.pdf
- https://cran.r-project.org/web/packages/adabag/adabag.pdf
- https://cran.r-project.org/web/packages/gbm/gbm.pdf
- https://cran.r-project.org/web/packages/xgboost/xgboost.pdf
- https://cran.r-project.org/web/packages/kernlab/kernlab.pdf
- https://cran.r-project.org/web/packages/neuralnet/neuralnet.pdf

6.5 Anexo 5: Equipo utilizado

Los modelos han sido ejecutados en un equipo con las siguientes características:

- Procesador Intel Core i5-8250U, con 4 núcleos y que permite ejecuciones simultáneas de 8 hilos, con una frecuencia máxima de 3,40 GHz.
- Memoria RAM de 16 Gb.

Nótese que si se utiliza un ordenador de peores características, los procesos que son intensivos en procesador requerirán de más tiempo de ejecución. Por contra, si se

dispone de un servidor distribuido o de una tarjeta gráfica potente, se podrían explorar más modelos mediante grids o entrenar más modelos que requieren de mayores tiempos de ejecución.