Proyecto Final – Segunda Entrega

Ciencia de Datos Aplicada
Universidad de los Andes, Bogotá, Colombia
Iván Saavedra Villamil, Salomón Novoa Montenegro, Julián Sanabria Mejía
{ir.saavedra232, s.novoa485, jj.sanabria}@uniandes.edu.co
Fecha de presentación: noviembre 3 de 2022

Tabla de contenido

1 Preparación de datos

Sempli es un *Fintech* dedicada a hacer créditos a pequeñas y medianas empresas que estén legalmente constituidas en Colombia. Para este ejercicio, Sempli nos ha suministrado un *dataset* con información del comportamiento de pago de sus clientes en cada crédito. El problema a resolver será clasificar a los clientes que pagarán o no pagarán el crédito de vuelta a Sempli, así mismo identificaremos el perfil de un mal cliente a partir de variables que describen a las empresas incluidas en el *dataset* como el monto desembolsado, la tasa de interés, la cantidad de empleados, el sector económico, etc.

Para hacer una preparación efectiva de los datos, tuvimos una reunión con el director de riesgo de Sempli quien nos explicó en qué consiste la gestión de riesgo dentro de la empresa y nos sugirió las variables que podían tener mayor impacto en el modelo, todo esto desde la perspectiva del negocio y la experiencia de la administración de Sempli.

Teniendo en cuenta las conclusiones de esta reunión, procedimos a hacer una limpieza del *dataset* el cual originalmente tenía 142 variables para finalmente dejarlo de solo 27 variable. La limpieza adicionalmente incluyó la eliminación de algunos créditos con valores nulos en variables importantes y la imputación de datos para evitar borrar créditos y disminuir la población de créditos que teníamos para analizar.

Variables descartadas

A continuación, describimos de forma general las razones por las cuales se descartaron 115 variables, no detallamos cada caso por la alta cantidad de variables descartadas.

- Descartamos algunas variables identificadoras (contienen id) que no ofrecían la posibilidad de entender el perfil de un cliente riesgoso por su naturaleza. Un ejemplo es el identificador del trato.
- Descartamos variables que contenían información semejante al de otras variables, pero con ligeras variaciones como es el caso de canal de captura y canal de captura ok dejando solo 1 de las variables.
- Descartamos variables con una alta cantidad de registros nulos como el *score Experian,* el cual, por problemas en el almacenamiento de datos, solo se registraba para 300 créditos.
- Descartamos variables con un alto nivel de detalle como la Clasificación de Actividades Económicas (CIIU) ya que tenía más de 1.000 variaciones dejando en su lugar la variable *Actividad eco SEMPLI* que agrupa los valores del CIIU en solo 12 variaciones
- Descartamos variables que el director de riesgo nos explicó no eran determinantes como el cargo de la persona solicitante del crédito
- Descartamos la mora de cada uno de los meses del crédito para quedarnos solo con la *variable predictora* que resume la información de todos los meses en 1 sola variable. Esta variable marca con 1 (alto riesgo de no pagar un crédito) a los créditos con al menos 1 cuota con más de 15 días de mora en los últimos 6 meses de vida del crédito y marca con 0 (bajo riesgo de no pagar un crédito) a los que tienen 15 días o menos de mora en los últimos 6 meses de vida del crédito.

Créditos descartados

En la reunión con el director de crédito entendimos que debíamos descartar los créditos clasificados como COVID en la variable *Estado,* ya que por temas operativos estos fueron créditos que se cancelaron durante la pandemia para luego convertirse en créditos totalmente nuevos que heredaban las características del crédito “padre” agregando las nuevas condiciones que se asignaban para dar alivios financieros a las empresas que durante la cuarentena no tenían liquidez para pagar sus obligaciones. Los nuevos créditos “hijos” sí los tenemos en cuenta en el conjunto de datos a analizar.

Valores imputados

- En la variable *Edad Empresarios* encontramos 1 registro con error de digitación que se validó con Sempli y se corrigió manualmente en el dataset.
- En la variable *Tipo de Garantía* encontramos errores ortográficos y diferentes formas de escribir el mismo tipo de garantía, bajo aprobación del director de riesgo de Sempli hicimos las correcciones pertinentes y agrupamos los tipos de garantías en solo 5 posibles variaciones.
- En la variable *Sempli Score* teníamos cerca de 600 valores en blancos (missing values) que correspondían a un error en la extracción de la data, Sempli nos compartió los valores faltantes y los actualizamos en el dataset del modelo.
- En la columna

Valores atípicos (outliers) ignorados

- En la variable *Número de empleados* encontramos 3 empresas con una cantidad de empleados considerablemente alta en comparación con las otras empresas, por esta razón descartamos los créditos con outliers en el *Número de empleados.*
- En la variable *Número de empleados mujeres* encontramos 5 empresas con una alta cantidad de empleados mujeres afectando la distribución de esta variable, por esta razón descartamos los créditos con outliers en el *Número de empleados mujeres.*
- En la variable *Año de Constitución* encontramos 10 empresas con el año de constitución considerablemente antiguo en comparación con las otras empresas.

Diccionario de datos

2 Estrategia de validación y selección del modelo

Ya que el objetivo del proyecto es construir un modelo que prediga si un cliente potencial de Sempli incurrirá en mora en el pago de su crédito; se crearán y entrenarán diferentes tipos de modelos de clasificación basados en las características de los préstamos y los clientes del historial de créditos proporcionado por Sempli.

Luego de realizar el proceso de preparación de datos, el *dataset* (historial de créditos de Sempli) cuenta con 1.890 instancias, de los cuales el 70% (1.323 instancias) será utilizado para entrenar los modelos, mientras el 30% restante (567 instancias) se utilizará para validar y probar los modelos.

Para nuestro proyecto, la clase positiva se presenta cuando los clientes entran en mora, teniendo en cuenta esto, el peor escenario para Sempli son los **falsos negativos** ya que se pierde más dinero si un crédito entra en default, en comparación a cuando se deja de percibir los intereses de un crédito que no se otorgó. Por ello, para evaluar los modelos construidos y entrenados, se hará uso de *Recall*. El modelo con el mayor *Recall* será seleccionado como el mejor, esta decisión también la apoyará la matriz de confusión que facilita la visualización de la predicción de los modelos.

Nuestra estrategia será probar diferentes modelos y utilizar los tres mejores, en cada uno de los modelos se aplicarán varios tipos de transformaciones, como lo son Estándar, Minmax y la polinomización sobre los *datasets* de entrenamiento y pruebas con el fin observar diferentes comportamientos de las características y encontrar el modelo de mejor predicción. De igual forma, se distribuirán los *datasets* de entrenamiento y pruebas de tal manera que la variable objetivo quede estratificada y se tendrá en cuenta el desbalance de las clases.

3 Construcción del modelo

Con en el objetivo de construir el modelo que mejor prediga si un cliente potencial de Sempli entrará en mora o no, se entrenaron tres algoritmos de clasificación (Random Forest, SVC y

Regresión Logística) con variedad de valores en sus hiper-parámetros. Los tipos de algoritmos entrenados y sus hiper-parámetros son los siguientes:

3.1 Random Forest

El primer tipo de algoritmo utilizado fue el Random Forest para cual se entrenaron diferentes modelos cambiando los hiper-parámetros en busca de un mejor desempeño, los cambios realizados sobre sus hiper-parámetros fueron:

- Se utilizaron los criterios gini y entropy, el mejor resultado para los valores de Recall y F1 Score se obtuvo con gini.
- Para max_features se probó con sqrt y log2, el mejor resultado se obtuvo con su valor por defecto (sqrt).
- Para class_weight, dado que las clases estaban desbalanceadas (70% - 30%), se utilizó el valor de balanced.
- Se probaron hasta nivel 4 de polinomialidad, pero en lugar de mejorar el resultado del Recall, disminuyó considerablemente por lo cual se utilizó el grado 1.

Luego de probar el algoritmo modificando diferentes combinaciones de los hiper-parámetros, para Random Forest se obtuvo el modelo con los mejores resultados aplicando los siguientes hiper-parámetros:

Hiper-parámetro	Random Forest
random_stat	0
n_estimators	1000
class_weight	balanced

Tabla 1. Hiper-parámetros para el mejor modelo de Random Forest

3.2 SVC

El segundo algoritmo trabajado fue el SVC, al igual que el anterior, se cambiaron los diferentes hiper-parámetros para lograr un mejor desempeño de los modelos, los cambios realizados sobre los hiper-parámetros fueron:

- Se utilizaron diferentes valores para el kernel (linear, poly, rbf, sigmoid), el mejor resultado para los valores de Recall y F1 Score se obtuvo con linear.
- Se probaron las dos opciones para probability (False o True), sin embargo, no se hubo diferencia significativa.
- Para class_weight, dado que las clases estaban desbalanceadas (70% - 30%), se utilizó el valor de balanced.
- Se probaron hasta nivel 4 de polinomialidad, pero en lugar de mejorar el resultado del Recall, disminuyó considerablemente por lo cual se utilizó el grado 1.

Luego de probar el algoritmo modificando diferentes combinaciones de los hiper-parámetros, para SVC se obtuvo el modelo con los mejores resultados aplicando los siguientes hiper-parámetros:

Hiper-parámetro	SVC
kernel	linear

class_weight	balanced

Tabla 2. Hiper-parámetros para el mejor modelo de SVC

3.3 Regresión Logística

El tercer y último tipo de algoritmo utilizado fue Logistic Regression, para este algoritmo se cambiaron los diferentes hiper-parámetros para lograr un mejor desempeño de los modelos, los cambios realizados sobre los hiper-parámetros fueron:

- Se utilizaron diferentes valores para el solver (newton-cg, lbfgs, liblinear, sag, saga), el mejor resultado para los valores de Recall y F1 Score se obtuvieron son 'sag'.
- La cantidad máxima de iteraciones se disminuyó progresivamente desde el valor de 100 y se encontró que el mejor resultado se obtiene con el valor de 23.
- Para class_weight, dado que las clases estaban desbalanceadas (70% - 30%), se utilizó el valor de balanced.
- Se probaron hasta nivel 4 de polinomialidad, pero en lugar de mejorar el resultado del Recall, disminuyó considerablemente por lo cual se utilizó el grado 1.

Luego de probar el algoritmo modificando diferentes combinaciones de los hiper-parámetros, para Regresión Logística se obtuvo el modelo con los mejores resultados aplicando los siguientes hiper-parámetros:

Hiper-parámetro	Logistic Regression
penalty	l2
solver	sag
class_weight	balanced
random_state	80
max_iter	23

Tabla 3. Hiper-parámetros para el mejor modelo de Logistic Regression.

4 Evaluación del modelo

Se entrenaron gran cantidad de modelos sobre los tres tipos de algoritmos seleccionados, cada uno con diferentes hiper-parámetros. Como ya se mencionó, se utilizó Recall para evaluar y determinar el mejor modelo, sin embargo, también se calculó las métricas de F1 Score y Precisión. La siguiente tabla ilustra los resultados obtenidos para el mejor modelo encontrado para cada uno de los algoritmos implementados, para facilidad de lectura y visualización, los valores de la tabla de resultados fueron redondeados a cuatro cifras decimales. El detalle de cómo se calcularon los valores de la tabla y su matriz de confusión se evidencian en la Notebook entregado.

Algoritmo	ENTRENAMIENTO			PRUEBAS		
	Precisión	Recall	F1 Score	Precisión	Recall	F1 Score
Random Forest	0.4752	0.7023	0.5582	0.4344	0.6154	0.5093
SVC	0.4532	0.6804	0.5440	0.4148	0.6090	0.4935
Logistic Regression	0.3868	**0.8099**	0.5236	0.3664	0.7564	0.4937

Tabla 4. Resultado de evaluar los modelos.

Como se observa en la Tabla 4, el mejor modelo de clasificación que se encontró es la Regresión Logística para la cual se obtuvo un *Recall* de 0.8099 en entramiento y 0.7564 en pruebas.

Dado que para Sempli, es más grave aceptar créditos de empresas con mayor propensión a quedar en mora, se tomó como medida principal el Recall y como medida secundaria el F1; dado que el Recall, permite hacer comparaciones con base en los Falsos negativos. Por esta razón y comparando los diferentes resultados de los tres mejores modelos escogidos, se escoge como modelo para calcular la propensión, la Regresión Logística.

Habiendo escogido el modelo, se realizó comparación de Underfitting y Overfitting, comparando el Recall para los datos train y test; Luego se seleccionar el mejor modelo (Regresión Logística), se realizó la comparación de sus métricas de Precsion, Recall y F1 Score para los datasets de entrenamiento y pruebas con el fin de determinar si presenta Underfitting u Overfitting.

	Train	Test
Precision	0.386	0.366
Recall	0.809	0.756
F1	0.523	0.756

Al comprar los valores obtenidos para estas métricas, se evidencia que no hay sobreajuste ni tampoco subajuste en el modelo.

Igualmente, al obtener un Recall igual a 0.756 se afirma que el modelo identifica de manera correcta el 75.56% de los clientes que realmente caen en mora

5 Conclusiones

Si bien la data que nos compartió Sempli era abundante en variables y registros, al hacer un análisis exploratorio de datos y tener una reunión con el director de riesgo de Sempli encontramos que buena parte del dataset se resumía tanto en créditos como en variables. También encontramos algunos problemas con la extracción de la data que fueron reportados a Sempli y solucionados con una actualización de los datos de parte de Sempli, con imputación y eliminando algunas variables cuando era el caso.

A pesar de que la regresión logística es un considerado uno de los algoritmos más sencillos dentro del *Machine Learning*, es una herramienta poderosa que genera buenos resultados;

evidencia de ellos son los resultados obtenidos en este proyecto, donde la regresión logística nos brinda mejores valores de *Recall* entre todos los modelos entrenados.