UNIVERSIDAD DE ANTIOQUIA

ANÁLISIS DEL USO DE TÉCNICAS DE *MACHINE LEARNING* PARA LA IDENTIFICACIÓN AUTOMÁTICA DE FALLAS EN LA PRESCRIPCIÓN DE ANTIBIÓTICOS

Autora

Deiry Sofía Navas Muriel

Universidad de Antioquia

Facultad, Departamento de Ingeniería de Sistemas

Medellín, Colombia

2021

Análisis del uso de técnicas de *Machine Learning* para la identificación automática de fallas en la prescripción de antibióticos

Deiry Sofía Navas Muriel

Proyecto de investigación
como requisito parcial para optar al título de
Ingeniera de Sistemas

Asesor:
Julián David Arias Londoño, PhD

Grupo de Investigación:
Grupo Intelligent Information Systems Lab (IN2LAB)

Universidad de Antioquia
Facultad, Departamento de Ingeniería de Sistemas
Medellín, Colombia
2021

Agradecimientos

Durante el desarrollo de este trabajo conocí la muerte el 13 de octubre de 2020, cuando ocurrió el feminicidio de Natalia, mi mejor amiga. Día en el que perdí una amistad auténtica, que a lo largo de nuestra relación nos separamos por el tiempo o circunstancias y ahora, trasciende más allá de la vida. Fue terriblemente penosa la pérdida de mi querida amiga y sigue siendo difícil despedirse, pero nada es eterno. Sólo se alcanza la inmortalidad a través de nosotros las que la conocieron y las que no, para mí ella participó en cada uno de estos párrafos y, a pesar del dolor, tengo mucho más que ofrecer luego de haber culminado este período de aprendizaje que me enriqueció profesionalmente y sobre todo personalmente. Ahora puedo asegurar que no estamos preparados para enfrentarnos con sentimientos tan complejos como la muerte, y sobre todo, lo que es capaz de despertar; sólo el amor puede aportar energías infinitas que permitan superar las adversidades e infundir esperanza.

Después de 10 meses cumplí la cita con mi futuro y llegó la hora de agradecer.

Agradezco a mi valía de llegar hasta aquí y tenerme paciencia cuando parece todo perdido y se avista la frustración.

A mi asesor, el profesor Julián que por su rigurosidad entregué un trabajo mejor de lo que esperaba, por hacer más ameno este duelo y sin duda, por compartirme sus mejores consejos y lo más valioso, su tiempo.

A mi familia, mi mamá, mi papá, mi hermana y, donde esté mi abuela, que a pesar que no entendieran del tema técnico me escucharon y me dieron aliento.

A mis amigos, Paula, Jose Luis, Laura, Amy, Angélica, Osorno, Marín, Ronal, Andrés Caicedo, Juan José por ser parte de mi vida, de momentos tristes y alegres.

En honor a Paula Natalia Fernández Montoya.

Análisis del uso de técnicas de Machine Learning para la identificación automática de fallas en la prescripción de antibióticos

1. Contexto del problema

Con el objetivo de cumplir con el sistema de vigilancia para el adecuado uso de antimicrobianos en la clínica León XIII de la IPS Universitaria, fue necesario asignar personal para hacer la revisión de las prescripciones de antibióticos realizadas diariamente a los pacientes hospitalizados; dicho proceso es altamente demandado y costoso ya que el personal que se requiere es de alta formación y la tarea de revisión de cada prescripción es demandante. Por lo tanto, el propósito de este trabajo es evaluar la capacidad de técnicas de Machine Learning (ML) para apoyar la toma de decisiones en este contexto, permitiendo de esa manera liberar de carga al personal médico de la unidad encargada. Los modelos de ML podrían determinar la probabilidad de que una prescripción se haya realizado de forma adecuada, de modo que le facilite al conjunto de especialistas del Departamento de Infectología (DI) direccionar sus esfuerzos a los casos que tengan mayor probabilidad de presentar errores. Este problema se aborda como un problema de clasificación de dos clases, una clase positiva que indica que tiene una alta probabilidad de que sea necesario hacer revisión manual por ser una prescripción inadecuada y la clase negativa, es decir, que no es necesaria la revisión por ser un tratamiento adecuado. Posteriormente se presenta una variante en la que la clase positiva se subdivide en tres grupos dependiendo de las causas por las cuales la prescripción se considera inadecuada.

2. Objetivos del trabajo de investigación

Evaluar la capacidad discriminante de algoritmos de *Machine Learning* para determinar automáticamente la pertinencia de la prescripción de antibióticos, de acuerdo con el modelo de análisis diseñado por el Departamento de Infectología de la IPS Universitaria de Medellín.

2.1 Objetivos del informe

Reportar los hallazgos que se han realizado durante el entrenamiento de los modelos y mostrar el comportamiento anormal detectado en la base de datos y cuáles fueron las alternativas de solución.

3. Metodología

La figura 1 presenta un diagrama con las diferentes etapas del proceso de análisis llevado a cabo en el presente trabajo.

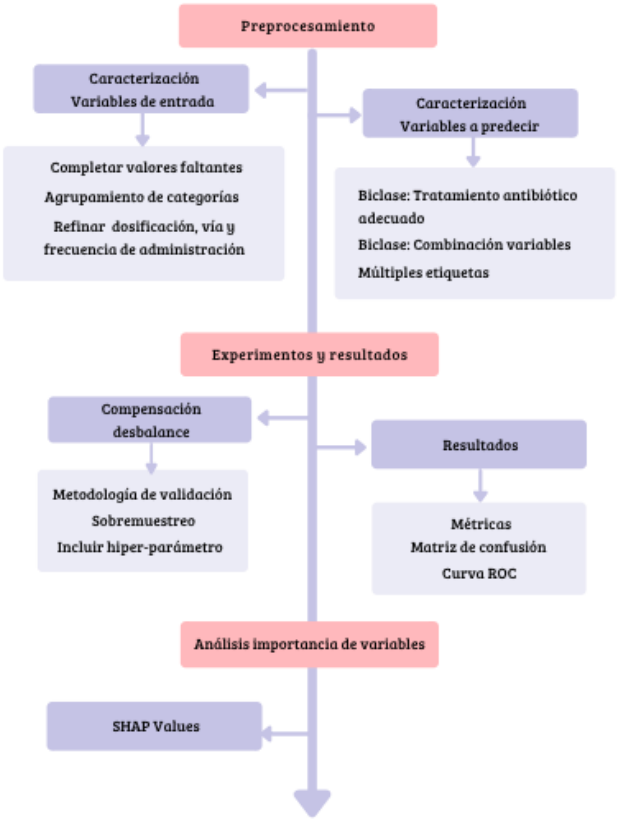

Figura 1. Etapas del proyecto

La primera etapa del proyecto implica un proceso inicial el a la base de datos donde se caracteriza las variables de entrada y las variables de salida de la cual se derivan tres problemas predictivos y se hace especial énfasis en la distribución desbalanceada de las muestras entre las clases a considerar en el estudio, razón por la cual en la etapa de experimentos se discuten las estrategias para contrarrestar el desbalance, incluyendo la metodología de validación, algunas técnicas de sobremuestreo y el uso de pesos diferentes para el error de cada clase durante la etapa de entrenamiento. Luego, se presentan los resultados de los modelos entrenados en los tres problemas predictivos con las correspondientes métricas de desempeño, la matriz de confusión y la curva ROC como apoyo visual para observar el desempeño de los mejores modelos y, en la tercera etapa, el análisis de importancia de las variables utilizando la técnica de los SHAP values [1], la cual se enfoca en darle interpretabilidad al resultado final de los modelos entrenados para entender cuáles fueron las variables que más aportaron a la tarea predictiva abordada. A continuación se presentarán de manera más detallada cada una de las etapas enunciadas.

3.1 Preprocesamiento

La base de datos recolectada por el DI contiene 5.888 prescripciones, cada prescripción está descrita por 44 variables, de las cuales 26 están compuestas por variables que caracterizan al paciente y su condición como la edad, el género, el servicio que prescribe, el diagnóstico de quién prescribe, el tipo de infección, la clasificación del cultivo, el antibiótico, la dosis, la frecuencia y el tiempo del tratamiento, entre otras y las 18 restantes están relacionadas con las acciones que sugiere el equipo del DI frente a la prescripción, a partir de las cuales se define la etiqueta final para el entrenamiento de los modelos de ML.

3.2 Caracterización de variables de entrada

Se realizó un análisis de las 26 características de entrada, las cuales están descritas en la tabla No. 1 con su nombre, tipo de codificación y el rango de valores según su tipo; si es fecha se incluye el intervalo de tiempo, si es numérica se muestra el valor mínimo y máximo, además, de la magnitud en la que está expresada y por último, la cantidad de categorías en los casos en que la variable sea de tipo categórica. A modo de resumen, se encuentran 2 variables tipo fecha, 4 numéricas y 20 categóricas.

Tabla 1. Descripción de las 26 variables de entrada por su nombre, tipo de codificación y rango

Nombre	Tipo	Rango
Ingreso	F	18/10/15 - 21/07/20
Inicio tratamiento	F	01/02/16 - 22/07/20
Año	N	2016-2020
Edad (Años)	N	5-104
Tiempo (días)	N	0-10
Dosificación (mg)	N	0-1000
Frecuencia (horas)	C	8
Vía administración	C	5
Mes de Evaluación	C	12
Sexo	CB	2
Servicio que prescribe	C	38
Antibiótico	C	31
Otro antibiótico	C	4
Diagnostico quien prescribe	C	78
Clasificación infección	CB	2
Tipo infección	C	4
Tratamiento	C	3
Cultivo	CB	2
Clasificación Cultivo	C	7
Aislamiento Micro-biológico	C	31
Tipo muestra 1	C	39
Germen 1	C	49
Perfil Germen1	C	21
Tipo muestra 2	C	24
Germen 2	C	31
Perfil Germen 2	C	15

Luego del análisis anterior, se aplicaron cuatro acciones para asegurar la calidad y consistencia de la información utilizada para el proceso de entrenamiento de modelos. En primer lugar, se eliminó la columna de otro antibiótico, debido a que sólo tiene 8 muestras y las 5880 restantes son valores vacíos. En segundo lugar, a las variables dosificación, frecuencia administración y vía administración se les aplica un proceso de imputación de datos para llenar los valores faltantes.

Para el tercer procedimiento, se encontró que las variables antibiótico, diagnóstico quien prescribe, germen 1, servicio que prescribe y tipo muestra 1, contaban con categorías con muy pocas muestras, por tanto, se agruparon en una nueva categoría para que los modelos puedan encontrar alguna similitud con otros registros. El agrupamiento se realizó teniendo en cuenta criterios clínicos. Por ejemplo, la variable antibiótico tiene una categoría llamada: albendazol, con sólo 1 registro y para agruparla se creó una categoría llamada antiparasitario. Esta nueva categoría terminó compuesta por dos antibióticos: albendazol y metronidazol. Este mismo análisis se realizó para todos los 33 antibióticos que finalmente se agruparon en 19 categorías, y por lo tanto se añadió una nueva variable llamada clasificación de antibiótico.

Tabla 2. Descripción de las 21 variables de entradas depuradas

Abrev.	Nombre	Tipo	Rango
Edad	Edad (Años)	N	5-104
Sexo	Sexo	C	2
DíasDiff	Días diferencia	C	0-8
Ant	Antibiótico	C	31
ClasfAnt	Clasificación Antibiótico	C	19
ClasfServPre	Clasficación servicio que prescribe	C	38
Tiempo	Tiempo (días)	N	0-10
Dosis	Dosificación (mg)	N	0-1000
Frec	Frecuencia (horas)	C	8
ViaAdmin	Vía admin.	C	5
DiagPres	Diagnostico quien prescribe	C	78
ClasfDiag	Clasificación Diagnostico quien prescribe	C	26
ClasfInf	Clasificación infección	C	2
Tto	Tratamiento	C	3
Cultivo	Cultivo	C	2
ClasfClv	Clasificación Cultivo	C	7
AislMicrob	Aislamiento Micro-biológico	C	31
TipMtra	Tipo muestra	C	29
Germ	Germen	C	41
PflGermen	Perfil Germen	C	21

Además del proceso seguido con la variable antibiótico, se realizaron los siguientes ajustes: con base en la variable diagnóstico quien prescribe, que contenía 78 categorías, se añade otra variable con 26 categorías llamada clasificación de diagnóstico; germen 1 de 49 categorías se termina reemplazando por una nueva de 41 categorías; servicio quien prescribe que cuenta con 38 categorías, fue sustituida por una nueva variable con 17 categorías; tipo muestra 1 pasa de 39 a 29 categorías. Por último se incluye la variable días de diferencia, esta se construye a partir de la fecha ingreso e inicio del tratamiento haciendo la resta entre ellas para sólo conservar esta

nueva variable de tipo numérica. Después de este procesamiento se llega finalmente a 21 variables como se muestra en la tabla No. 2. Para las variables categóricas, se aplicó la técnica de codificación One-Hot-Encoding resultando en 276 variables de entrada para el entrenamiento de los modelos.

3.3 Definición de las variables a predecir

De las 18 variables de salida se descartan las variables diagnóstico de infectología y condición al alta, debido a que, aunque discuten la exactitud del diagnóstico inicial, no están definiendo directamente la idoneidad de la prescripción inicial, que es el propósito de los modelos de ML a construir. En definitiva, son 16 variables las que permiten definir la etiqueta final en el entrenamiento de los modelos de ML, estas variables incluyen las recomendaciones de cambio relacionadas con el antibiótico, la dosis, el tiempo, la posibilidad de daño colateral, o la inclusión o no en el POS, entre otras. La figura 2 muestra el listado de variables asociadas a las recomendaciones realizadas por el equipo del DI y se muestra el número de prescripciones en la categoría que corresponde.

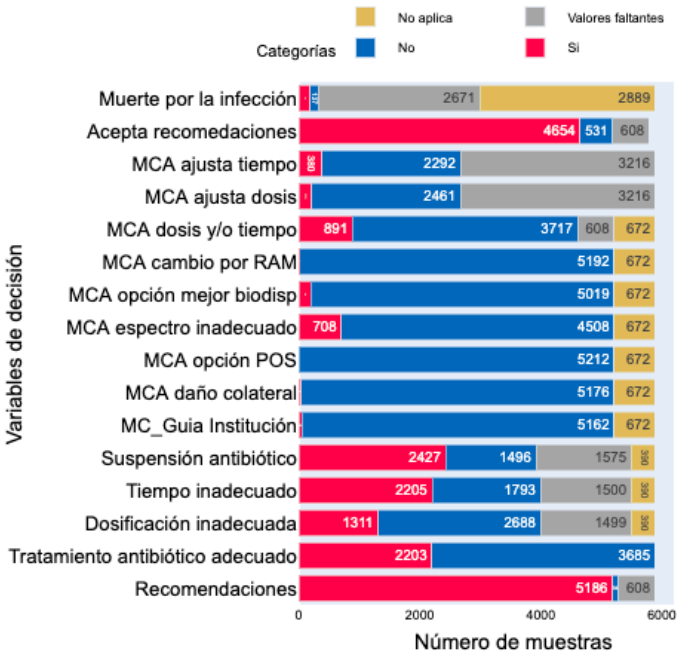

Figura 2. Distribución de las variables de salida para la etiqueta final.
MCA: Motivo Cambio Antibiótico. MC: Motivo Cambio

A partir de la fig. 2 se definieron tres problemas predictivos diferentes por la variable de salida. Para el primer grupo se utilizó sólo la variable tratamiento antibiótico adecuado, la cual configura un problema biclase usando la convención preestablecida de positivo: tratamiento inadecuado y negativo: tratamiento adecuado. La figura 3 muestra la distribución de registros de acuerdo con dicha variable para el problema 1.

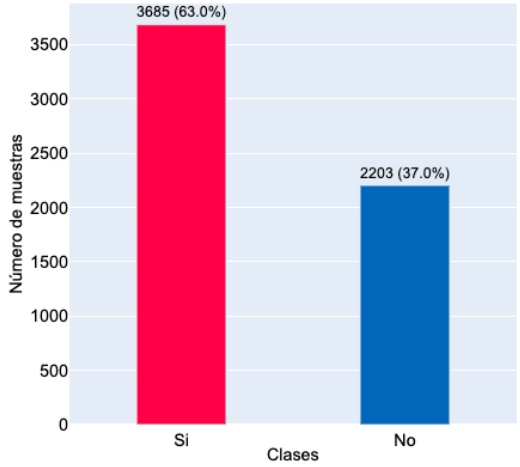

Figura 3. Distribución de las muestras para el problema 1. La barra con la etiqueta *Si* representa las prescripciones que requieren revisión.

Si bien la variable tratamiento antibiótico adecuado permitió etiquetar todos los registros de la base de datos y no contenía valores faltantes, existen otras variables relacionadas con requerimientos de modificación de las prescripciones, por lo que se decidió definir una etiqueta alternativa y por lo tanto un nuevo problema predictivo. Para el segundo problema, la variable de salida se construye a partir de la combinación de las 16 columnas que definen acciones sobre la prescripción bajo análisis. La columna "Recomendaciones" fue usada como variable base para este proceso, teniendo en cuenta que es la que tiene la menor cantidad de valores faltantes (608 registros) entre las variables analizadas. Para asignar la etiqueta a los 608 registros con valores faltantes en la variable Recomendaciones, se utilizó la información de las 15 columnas restantes; de los 608, 360 registros fueron asignados a la clase positiva, es decir la clase asociada a las prescripciones que requieren ajustes, y los restantes a la clase negativa. La figura 4 muestra la distribución de registros de acuerdo con la etiqueta establecida en el problema predictivo 2.

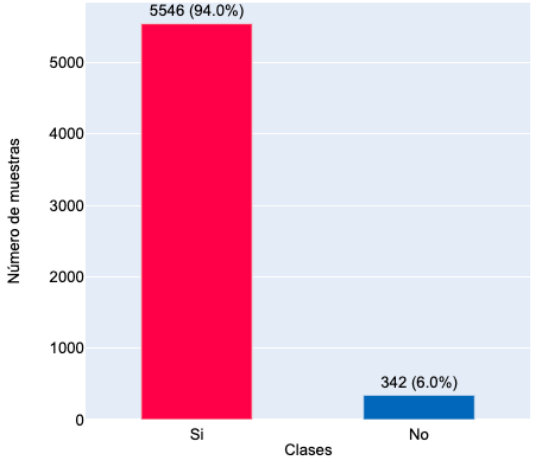

Figura 4. Distribución de las muestras para el problema 2. La barra con la etiqueta *Si* representa las prescripciones que requieren revisión.

En la figura 4 se puede evidenciar que la distribución de muestras está desbalanceada, ya que se observa que hay un 94% de muestras de la clase positiva, en contraste con un 6% de la clase negativa, esto marca una proporción de 16:1, indicando que por una muestra de la clase 0 hay 16 de la clase 1, es decir, que para la mayoría de prescripciones en la base de datos, fue necesario hacer revisión en la prescripción por parte del equipo de DI.

Desde el punto de vista de ML, la distribución de clases convierte al problema en uno de tipo desbalanceado, por lo tanto, como se verá en secciones posteriores, durante la fase experimental se tendrán en cuenta algunas estrategias para su compensación. Sin embargo, llama la atención que la clase mayoritaria en este caso, sea la clase que corresponde a las prescripciones que requieren de ajuste y no a las que no lo requieren. Es importante recordar que el objetivo último de este análisis, es revisar la viabilidad de construir un sistema automático que a partir de técnicas de clasificación y pueda filtrar los casos que deben ser analizados por parte del personal de DI, es decir, identificar los casos que no requieren de ningún ajuste, para que el personal del DI se pueda centrar en los casos que sí requieren modificaciones en la prescripción. No obstante, si el 94% de los casos requieren modificaciones, eso significa que el potencial de liberación de tiempo por parte del personal, debido a la implementación de la solución, sería muy limitada filtrando, en el mejor de los casos, el 6% de los casos que deben analizarse.

El tercer y último problema predictivo abordado en el trabajo, tiene como propósito evaluar la capacidad de los modelos de ML para determinar la causa por la cual una prescripción fue identificada como no apropiada. Es importante tener en cuenta que una prescripción pudo haber sido clasificada como no adecuada por varias causas, por lo que el problema desde el punto de ML se considera como de múltiples etiquetas. El problema está compuesto de las siguientes tres etiquetas:

(1) La primera clase corresponde a una causa asociada al tipo de antibiótico prescrito. La etiqueta es la unión entre las variables: tratamiento antibiótico adecuado, suspensión de antibiótico y MCA por espectro inadecuado. A esta clase se le asignó el nombre de antibiótico adecuado y cuenta con una relación de 3:1 en la base de datos.
(2) La segunda clase está relacionada con causas debidas a problemas con la dosis o el tiempo de prescripción, más no con el tipo de antibiótico. La etiqueta combina 5 variables: dosificación inadecuada, tiempo inadecuado, MC de antibiótico por dosis, MC de antibiótico por tiempo y MC de antibiótico por dosis y/o tiempo; se tomó como referencia esta última debido a que tiene la combinación de los dos cambios y para completar los vacíos se realiza con las variables restantes, además se eliminaron los 608 registros con la clase no aplica, dado que no se puede clasificar en ninguna de las dos clases. Esta etiqueta se denomina MC dosis y/o tiempo y presenta una proporción de 4:1 en la base de datos.
(3) La última clase agrupa las causas restantes y su etiqueta se obtiene a partir de la combinación de las variables: MC por guía institución, opción POS, daño colateral y cambio por RAM; se denomina MC admin y además esta etiqueta es la que tiene más desbalance en comparación con las dos anteriores, con una tasa de desbalance significativa de 44:1.

Para este tercer problema se usaron 5.216 muestras de las 5.888 originales para realizar los procesos de entrenamiento y validación de los modelos. La figura 5 muestra la distribución de registros de acuerdo con la etiqueta establecida en el problema predictivo 3.

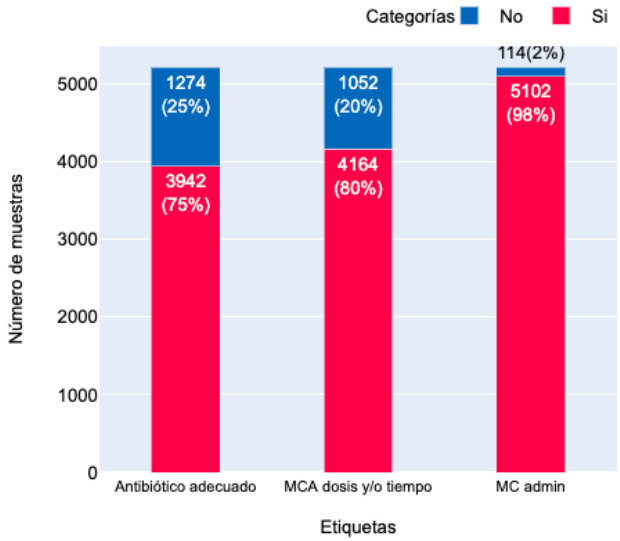

Figura 5. Distribución de muestras para el problema predictivo 3

En resumen, se plantean 3 problemas predictivos donde cada uno tiene desbalance de diferentes proporciones; también es importante recordar que los problemas 1 y 2 son biclase y el 3 es de múltiples etiquetas, es decir, un registro puede pertenecer a varias clases de manera simultánea ya que la razón por la cual se identifica como no apropiada la prescripción, puede estar asociada a diferentes causas. No obstante el fenómeno de desbalance descrito, se realizaron las pruebas de entrenamiento y validación de modelos de los tres problemas predictivos planteados, siguiendo el objetivo trazado inicialmente.

4. Modelos de *Machine Learning* y técnicas de análisis de importancia de variables

A continuación se presenta una breve descripción de las técnicas usadas, tanto en la fase de modelamiento de los datos como para la identificación y análisis de las variables de mayor interés. Este último análisis es relevante, teniendo en cuenta que, de cara a implementar en un futuro un sistema de información que realice la clasificación automática de las prescripciones, es necesario automatizar también la obtención de las características y, por lo tanto, identificar las características menos relevantes, da luces sobre la envergadura y retos a enfrentar para el desarrollo del sistema de información.

4.1 *Modelos de Machine Learning*

Los algoritmos seleccionados para abordar los deferentes problemas de clasificación definidos fueron XGBoost [1], *Random Forest* (RF) [2], Redes Neuronales Artificiales (RNA) [3] y las Máquina de Vectores de Soporte (en inglés *Support Vector Machines* - SVM) [4]. Todos los modelos corresponden a técnicas ampliamente usadas en el estado del arte, y que cuentan con parámetros o estrategias apropiadas para compensar el desbalance que se encuentra en la base datos, puesto que tienen la capacidad de considerar el sesgo inicial cuando se introduce los datos al algoritmo y así, tratar el error de cada clase de manera diferente. De esta manera se intenta evitar que el modelo se incline hacia la clase que tiene mayor presencia en el conjunto de muestras de entrenamiento.

Como los dos primeros modelos están basados en combinaciones de árboles de decisión, estos pueden introducir el sesgo durante el proceso de muestreo para el entrenamiento de cada árbol del conjunto, no obstante una de las principales diferencias entre el RF y XGBoost está en la técnica con que ensamblan los árboles débiles, por un lado, el RF utiliza la estrategia de *bagging,* que realiza una combinación en paralelo y así aprovecha la independencia de cada árbol para posteriormente definir una salida única a partir de una regla de consenso. Por el contrario, el XGBoost emplea el método *boosting* que funciona de forma secuencial con el propósito de mejorar los errores que no han podido ser resueltos por los modelos previos. Ambos modelos son muy empleados en el estado del arte y están presentes liderando muchas de las competencias actuales que involucran tareas de ML.

En los siguientes dos modelos, las RNA y SVM, la compensación del desbalance se realiza introduciendo un peso dentro de la función de costo que se usa durante el entrenamiento del modelo. Sin embargo, los mecanismos de aprendizaje son diferentes, las RNA están basadas en un algoritmo de optimización por gradiente descendente llamado *backpropagation*, en el cual los parámetros de la red se ajustan de manera iterativa a través de la minimización de la función de costo, mientras que las SVM aplican funciones *kernel* para transformar los datos a espacios de dimensión mayor, que permitan encontrar una frontera de decisión con mejores características de separación entre las clases usando el criterio de máximo margen.

4.2 Técnicas de análisis de importancia de variables

Como ya se comentó, es importante determinar cuáles son las variables que impulsan al modelo para que clasifique si una prescripción requiere de revisión o no y así. Por esta razón, se implementaron métodos de ML que permitieran dar una interpretación sobre la salida de los modelos evaluados. Algunos de los modelos evaluados cuentan con estrategias para determinar la importancia de las variables durante el proceso de entrenamiento, en particular los modelos basados en árboles pueden usar estadísticos basados en el uso de la frecuencia de uso de las variables en los nodos de decisión, o en el grado de impureza que éstas aportan, para determinar las variables con mayor capacidad discriminante. Sin embargo, teniendo en cuenta que las particiones que se realizan en los modelos basados en árbol son aleatorias, los índices de importancia de variables derivados son, en algunos casos, inestables.

Adicionalmente, estos índices permiten determinar las variables importantes durante el entrenamiento, pero no dan información sobre las variables determinantes cuando se evalúa una prescripción particular.

Para solventar los problemas descritos antes, recientemente se propuso la técnica conocida como *SHapley Additive exPlanation*s (*SHAP values*) [5], la cual es ampliamente utilizada en aplicaciones médicas y funciona independientemente del modelo usando una estrategia aditiva para determinar el impacto (positivo o negativo) de cada una de las variables en la decisión tomada por el modelo [5], por lo tanto no sólo representa la contribución de las variables en términos generales sino que también, puede analizar las predicciones individuales. Por lo tanto los SHAP values pueden identificar las variables que más aporte tuvieron para la decisión del modelo y servirán de guía para que el personal del DI pueda guiar el análisis, encontrar e intervenir las causas por las cuales las prescripciones clasificadas como deficientes deben ser ajustadas.

5. Experimentos y resultados

5.1 Diseño experimental

Para la fase de entrenamiento se tuvo en cuenta que en los tres problemas predictivos hay desbalance y para compensarlo se emplearon 3 estrategias de compensación según el tipo de clasificador (biclase o múltiples etiquetas). En primer lugar la metodología de validación usada garantiza que la distribución de clases o etiquetas se mantenga balanceada a la hora de separar el conjunto de datos por entrenamiento, validación y test, en particular, para los problemas 1 y 2 se utilizó la metodología conocida como validación cruzada estratificada[6] y para el problema 3 la validación estratificación iterativa (en inglés iterative stratification) [7]. No obstante para los tres problemas se realizó la misma distribución, para el conjunto de test se destinó el 20% del total de los datos y el 80% restante se dividió en 5 folds para el entrenamiento y validación.

La segunda estrategia para compensar el desbalance fue evaluar técnicas de sobremuestreo, esto es con el propósito de generar muestras artificiales de la clase minoritaria. Entre las técnicas de sobremuestreo disponibles, se usó SMOTE (Synthetic Minority Oversampling Technique) [8] que es una de las más ampliamente utilizadas y también se realizaron pruebas con algunas variantes, tales como, SMOTE-NC [9], útil para las características que son de tipo categóricas y ADASYN[10] que genera de forma adaptativa las muestras de acuerdo a su distribución, sin embargo, los mejores resultados fueron alcanzados con la técnica SMOTE convencional. En el caso de los dos problemas biclase los porcentajes de sobremuestreo que se analizaron fueron 10%, 30% y 40%. Para el caso del problema 3, teniendo en cuenta que cada una de las etiquetas de salida tiene diferentes proporciones de desbalance (ver figura 4), no se puede aplicar el mismo porcentaje para las 3 etiquetas (Antibiótico adecuado, MC dosis y/o tiempo y MC admin.), es por ello que se utiliza el cociente de desbalance (en inglés Imbalanced Ratio - IR) para preguntar cuál es el porcentaje acorde con la magnitud del desbalance. Este cociente se define como la división entre el número de muestras de la clase mayoritaria y las muestras de la minoritaria,

indicando que entre más grande es el cociente, el desbalance es más significativo, así que se probaron varios porcentajes de sobremuestreo así:

- Si el cociente es mayor o igual que 40: El sobremuestreo es entre 1200%,1100% y 1000% .
- Si el cociente es mayor o igual que 10 y menor que 40: El sobremuestreo es entre 100%, 50% y 20%.
- Finalmente, si el cociente es menor que 10: el sobremuestreo es entre 12%,9% y 3%.

Para utilizar la librería SMOTE en el caso del problema 3, se debió hacer un procesamiento previo llamado transformación del problema. Consiste en combinar las 3 etiquetas de salida en una sóla y así queda una salida multiclase apta para aplicar SMOTE como en los problemas 1 y 2. Posteriormente, se debe revertir el proceso a múltiples etiquetas con el sobremuestreo incluido para realizar el proceso de entrenamiento de los modelos. Cabe señalar que se excluye el conjunto de test para conservar intacto los datos originales fuera del sobremuestreo en los tres problemas.

Por otro lado, la tercera y última estrategia utilizada para la compensación del desbalance fue el ajuste de un hiper parámetro en los modelos evaluados, para que tengan en cuenta de manera diferente los pesos asociados al error de cada clase durante el entrenamiento, esto aplica de igual forma para los tres problemas predictivos como se explicó en la sección de las técnicas de importancia de las variables.

Por el otro lado, las medidas de desempeño en los dos problemas biclase fueron: *accuracy (ACC)*, el *balanced accuracy* (BACC) y el F1. Teniendo en cuenta el desbalance, era importante considerar medidas que proporcionarán información sobre el desempeño real del modelo bajo las condiciones de la base de datos. En ese sentido las medidas de desempeño con mayor interés son el BACC y el F1. En relación con el problema de múltiples etiquetas se considera incluir las métricas *accuracy (ACC)*, *Hamming Loss (HAL)* [11], *ranking loss (RAL)*[12] *y zero-one loss* [13]. De las métricas *HAL*, *RAL* y *zero-one loss* es importante tener en cuenta que son funciones de pérdida, por tanto, entre menor sea su valor, mejor es el rendimiento del algoritmo analizado; para un clasificador ideal, la pérdida es 0. La métrica *HAL* entrega el promedio de los porcentaje de error de clasificación de cada etiqueta; por su parte la métrica *RAL* impone una penalización al clasificador cuando un par de etiquetas se clasifica incorrectamente, donde la tasa de clasificación errónea $r_i(j)$ es la probabilidad de que una muestra para la etiqueta positiva sea menor que la de la etiqueta negativa, en síntesis se considera esta función como la expectativa de la tasa de clasificación errónea $r_i(j)$ sobre todos los pares posibles de etiquetas[12], por último, la métrica *zero-one loss* es una función de costo que devuelve 1 si son correctas las predicciones para todas las etiquetas en una muestra y 0 de lo contrario. De manera formal, dado un conjunto de muestras $\{x_i, y_i\}$, $i = 1 \ldots N$observaciones con L número de etiquetas y sea $\hat{y_i}$ las etiquetas predichas por el modelo, las expresiones que definen las métricas de evaluación usadas están dadas por:

$$Hamming\ Loss\ = \frac{1}{NL}\sum_{i=1}^{N}\sum_{j=1}^{L} I[\widehat{y_{ij}} \neq y_{ij}] \quad (1)$$

$$Ranking\ loss\ = \frac{1}{N}\sum_{i=1}^{N} \sum_{(j,k):\widehat{y_j} > \widehat{y_k}}^{L} (I[r_i(j) < r_i(k)] + \frac{1}{2} I[r_i(j) = r_i(k)]) \quad (2)$$

$$donde\ r_i(j) := Es\ la\ probabilidad\ de\ la\ etiqueta\ j\ para\ la\ instancia\ x_i$$

$$Zero\ one\ loss\ = \frac{1}{N}\sum_{i=1}^{N} I[\widehat{y_{ij}} \neq y_{ij}] \quad (3)$$

En las definiciones anteriores, $I[c]$ es una función indicador que retorna 1 si la condición se cumple y 0 en caso contrario.
Además de las métricas anteriores, los resultados incluyen el área bajo la curva ROC y la matriz de confusión en los 3 problemas evaluados.

Teniendo en cuenta los modelos descritos en la sección 4.1, a continuación se relaciona en la tabla 3 cuáles fueron los hiper parámetros y el rango de valores que se utilizaron durante el proceso de ajuste y entrenamiento de los 4 modelos evaluados.

Tabla 3. Malla de valores para entrenar los algoritmos

Modelo	Hiper parámetros
XGBoost	eta: 0.001, 0.01, 0.1, 0.2 max_depth: 1, 2, 4, 6, 8, 10 n_estimators: 100, 120, 140, 160
RF	max_depth: 2, 3, 4, 6 n_estimators: 100,120,140 max_features: sqrt, log2, 50, 100
RNA	epochs: 25, 50, 100 batches: 16, 32, 64 layers: 1, 2, 3 neurons: 16, 32, 64 drop_out = 0.2, 0.3, 0.4, 0.5
SVM	kernel: rbf, linear gamma: 0.01, 0.001, 0.0001 C: 1, 10, 100, 200,500

5.2 Resultados y discusión

A continuación se van a presentar los resultados de los tres problemas predictivos descritos en las secciones anteriores. Los resultados de los procesos de entrenamiento y validación de los modelos se presentan como experimentos 1 a 3, en correspondencia con cada uno de los tres problemas de predicción.

La tabla 4 muestra los resultados obtenidos para el problema 1. Como se puede observar en la tabla, el modelo con mejor resultado es la RNA con un 61% en la métrica BACC y utilizando SMOTE. Es importante resaltar que, aunque de los tres problemas definidos éste es el que menos desbalance presenta, se pueden observar mejores tasas de acierto cuando se utilizan las estrategias de compensación basadas en sobremuestreo en comparación a cuando sólo se usan las muestras reales. El mejor resultado obtenido y reportado en la tabla 4, correspondió a un sobremuestreo del 30%. Así mismo, la RNA que obtuvo los mejores resultados estaba configurada de la siguiente manera: dos capas ocultas, la primera con 64 neuronas y la segunda con 16, un dropout de 0.4, batch de 32 y 25 épocas durante el entrenamiento.

Tabla 4. Resultados clasificación automática de tratamiento antibiótico adecuado

Modelo	**ACC**		**BACC**		**F1**		**AUC**	
	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A
XGBoost	0.59	0.64	0.60	0.58	0.53	0.40	0.64	0.64
Random Forest	0.61	0.63	0.592	0.60	0.492	0.50	0.63	0.64
Redes Neuronales	0.61	0.60	0.61	0.58	0.54	0.50	0.59	0.60
SVM	0.57	0.61	0.57	0.57	0.49	0.44	0.6	0.59

La figura 6 presenta la matriz de confusión del modelo de RNA que obtuvo los mejores resultados para el problema predictivo 1.

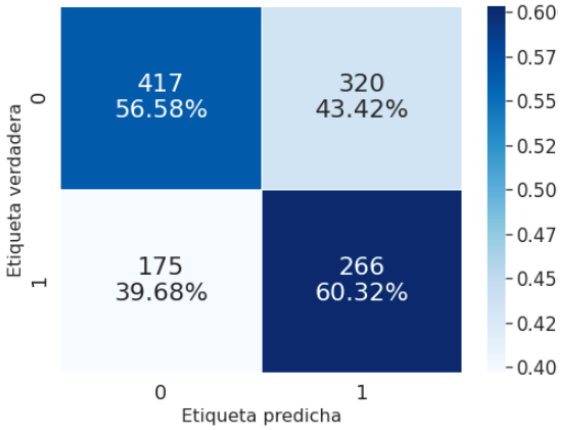

Figura 6. Matriz de confusión del mejor modelo (RNA) en el problema 1.

Es necesario recalcar que la clase positiva corresponde a un tratamiento de antibiótico inadecuado, por consiguiente, necesita de revisión del DI; por lo tanto el mayor riesgo en los errores del sistema se presenta en los falsos negativos, que corresponden a casos en los que el modelo predice que la prescripción bajo análisis contiene un tratamiento adecuado cuando en realidad es incorrecto. Analizando la fig. 6 se observa que el modelo tiene más dificultad en clasificar la clase mayoritaria (positiva) que la minoritaria, es decir, las muestras de la clase positiva se clasificaron incorrectamente aproximadamente en un 40%.

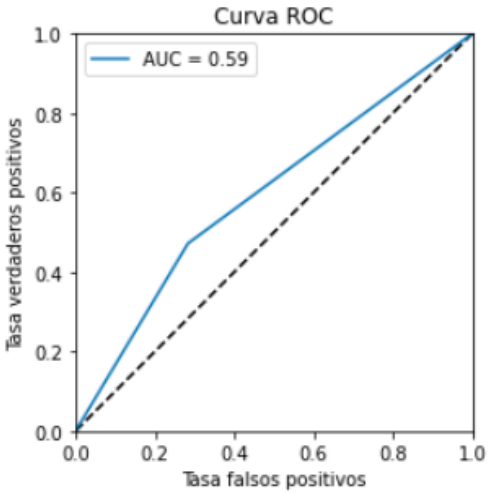

Figura 7. Curva ROC y AUC de la RNA

En la figura 7 se puede observar la curva ROC del mejor modelo encontrado. A partir de ésta se confirma lo observado en la matriz de confusión con una curva apenas superior a la diagonal -que establece el punto de referencia para un sistema que predice de manera aleatoria-, con un AUC que alcanza el 0.60.

A continuación se presentarán los resultados con respecto al problema predictivo 2. En la tabla 5 se muestran los resultados de las mejores configuraciones encontradas para cada uno de los modelos evaluados.

Tabla 5. Resultados clasificación automática si requiere de revisión o no

Modelo	**ACC**		**BACC**		**F1**		**AUC**	
	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A
XGBoost	0.89	0.94	0.8	0.61	0.94	0.97	0.80	0.79
Random Forest	0.9	0.94	0.81	0.54	0.94	0.97	0.79	0.80
Redes Neuronales	0.78	0.94	0.79	0.66	0.95	0.97	0.79	0.761
SVM	0.88	0.94	0.79	0.63	0.94	0.97	0.81	0.73

Como se analizó en la sección 3.3 y en la figura 4, el desbalance en este caso es significativo por lo que cobra mayor importancia el uso de las estrategias para compensar el desbalance. Después de analizar la tabla 5 se concluye que el mejor modelo es el RF con un 81% en la métrica BACC utilizando SMOTE al 40%. Es importante resaltar que aunque los modelos sin SMOTE presentan mayor valor de F1, el BACC disminuye. También es importante recordar que aunque se presenta la métrica de ACC, ésta no debe ser tomada como punto de referencia debido a que presenta valores sesgados a favor de la clase mayoritaria.

Los resultados alcanzados por el RF en la tabla 5, fueron obtenidos con los siguientes hiper parámetros: máxima profundidad de 4, máximo número de características por nodo de 16, el número de estimadores de 120. En la figura 8 se muestra la matriz de confusión correspondiente al RF con el mejor resultado.

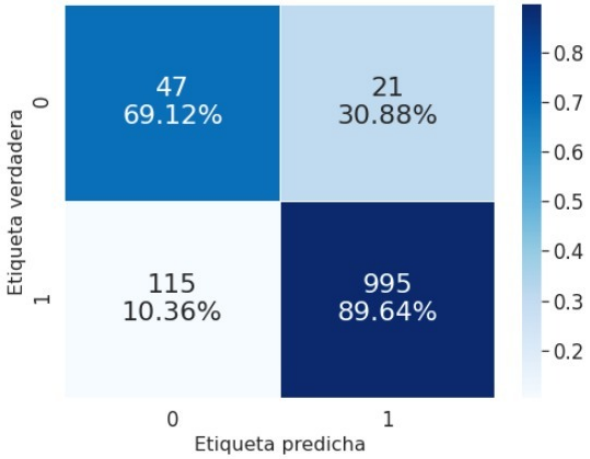

Figura 8. Matriz de confusión del modelo RF

En la figura 8 se puede observar que la clase positiva se predice de manera correcta en aproximadamente un 90%, es decir, el modelo presenta un porcentaje de falsos negativos del 10%, representando una mejora significativa con respecto a los resultados obtenidos en el experimento 1.
En cuanto a la clase negativa se muestra que por cada 68 prescripciones que no requieren revisión, el modelo se equivoca en 21, esto es menos de la mitad, de modo que es una mejora con respecto a la anterior etiqueta, teniendo en cuenta que esta clase tiene significativamente menos muestras que la clase positiva para el entrenamiento del modelo.

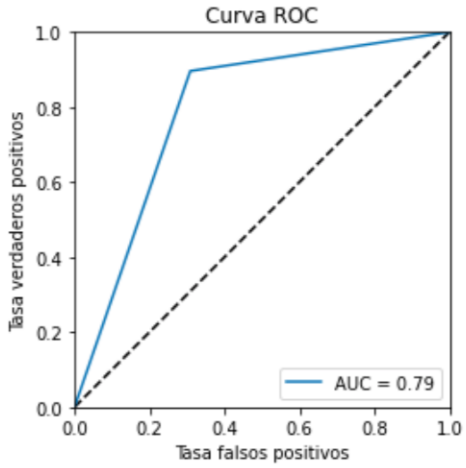

Figura 9. Curva ROC y AUC del RF en el problema 2

En la figura 9 se presenta la curva ROC y el correspondiente AUC para el mejor modelo encontrado en el problema 2. Se observa una mejora con respecto al problema anterior, aunque refleja también que la distribución de puntuaciones de las clases positiva y negativa no es Gausiana ya que el codo de la curva se encuentra cercano a 0.9 para la tasa de verdaderos positivos y a 0.3 para la tasa de falsos positivos, con un sesgo claro hacia la clase positiva.

En la tabla 6 se presentan los resultados obtenidos para los diferentes modelos evaluados en el problema 3. De acuerdo con la tabla, el mejor modelo en este caso es nuevamente el RF que presenta la menor pérdida según la métrica HAL de 15%. Sin embargo es necesario tener en cuenta que la métrica HAL no tiene en cuenta el desbalance de las etiquetas. Para el entrenamiento de este modelo se usó un sobremuestreo en la etiqueta MC admin. de 1100%, MC dosis y/o tiempo de 50% y para antibiótico adecuado de 9%.

Tabla 6. Resultados de clasificación automática del problema 3

Modelo	**HAL**		**zero-loss**		**RAL**		**ACC**	
	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A	SMOTE	N/A
XGBoost	0.20	0.157	0.49	0.45	0.30	0.44	0.511	0.56
Random Forest	0.15	0.15	0.43	0.44	0.37	0.41	0.56	0.56
Redes Neuronales	0.15	0.15	0.44	0.44	0.42	0.44	0.56	0.56
SVM	0.16	0.16	0.46	0.46	0.45	0.45	0.54	0.54

Los hiper parámetros del RF con los cuales se obtuvieron los resultados presentados en la tabla 6 son los siguientes: máxima profundidad de 14, máximo número de características de 16, un número de estimadores base de 200. Las figuras 10, 11 y 12 presentan las matrices de confusión para cada una de las etiquetas de salida, en las cuales se puede observar el efecto del problema

de desbalance, el cual afecta el desempeño para todas las etiquetas, siendo *MC Admin* la más crítica.

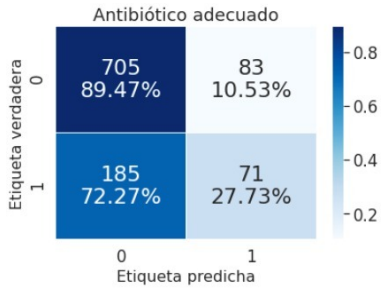

Figura 10. Matriz de confusión para la etiqueta tratamiento adecuado

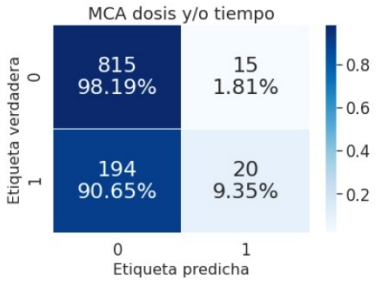

Figura 11. Matriz de confusión para la etiqueta MC dosis y/o tiempo

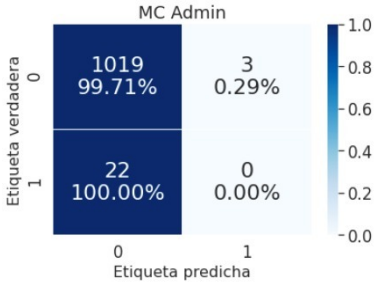

Figura 12. Matriz de confusión para la etiqueta MC Admin

A partir de las figuras 10, 11 y 12 se concluye que, en general, el sistema no acertó en más del 50% ninguna de las clases, es decir que diferenciar de manera automática el tipo de error en la prescripción, es aún difícil a partir de las variables incluidas en la BD. A continuación en la fig. 13 se representan las curvas ROC y el AUC del RF para cada una de las etiquetas en el problema 3.

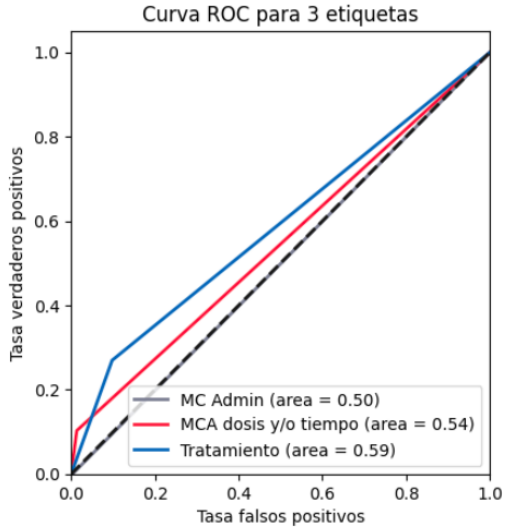

Figura 13. Curva ROC y AUC del modelo RF en el problema 3 para las tres etiquetas evaluadas.

Como se esperaba, las curvas ROC en la figura 13 no muestran desempeños sobresalientes para ninguna de las etiquetas. Se observa además que en este caso, las estrategias para compensar el desbalance tuvieron menos éxito que lo observado en el problema 2.

Para terminar, en la tabla 7 se presenta un resumen de los mejores modelos y métricas de desempeño de los tres problemas predictivos abordados.

Tabla 7. Resumen de los mejores resultados para los tres problemas predictivos

Problema	Modelo	ACC	BACC	HAL
1	RNA	0.61	0.61	N/A
2	RF	0.90	0.81	N/A
3	RF	0.56	N/A	0.15

5.3. Análisis de importancia de variables

Teniendo en cuenta la importancia de determinar las variables de mayor influencia dentro del modelo de clasificación, en esta sección se presentan los resultados utilizando la técnica de los SHAP values. Para esta fase, sólo se utilizó el modelo RF del segundo problema, debido a que fue el modelo con los mejores resultados, pero además el único que alcanzó medidas de BACC superiores al 80%. En los demás casos, realizar un análisis de importancia de variables no tendría sentido, ya que la confiabilidad sobre los hallazgos sería mínima teniendo en cuenta que el nivel de acierto en la clasificación de los modelos es muy bajo.

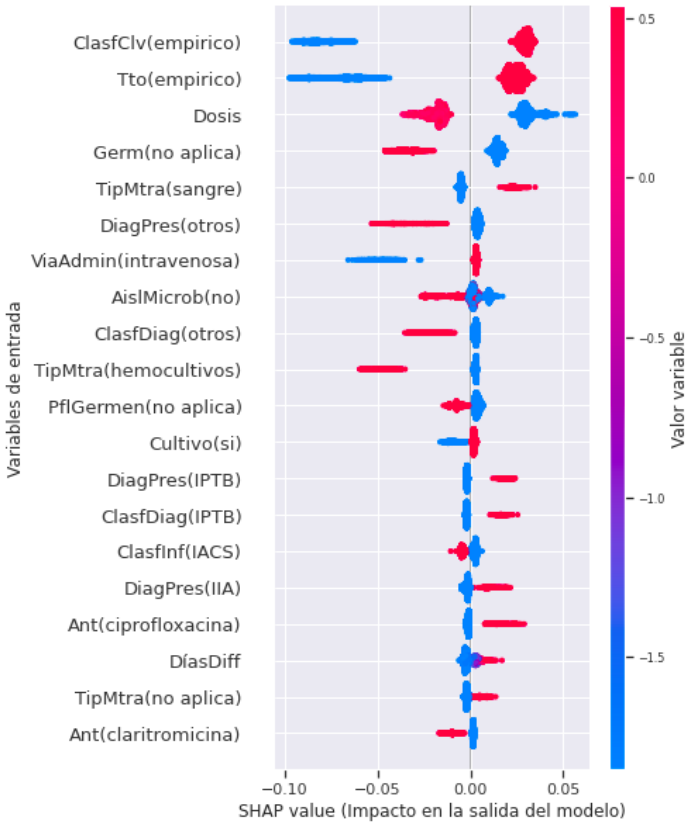

Figura 14. Gráfica que representa el impacto de las variables de entrada según los SHAP Values. IPTB significa Infección de piel y tejidos blandos, IACS Infección asociada al cuidado de la salud, IIA es Infección Intraabdominal.

En la fig. 14 se observa la importancia de las variables para todo el conjunto de entrenamiento, de acuerdo con el análisis proporcionado por los SHAP values. Para comprender y analizar la gráfica se debe tener en cuenta lo siguiente:

1. En el eje “y” se reúnen todas las variables por orden de importancia siendo la primera la que más contribuye y la última la de menor, o incluso ninguna, contribución a la predicción. Teniendo en cuenta que muchas variables tienen nombres largos, la gráfica presenta las variables con nombres resumidos; la equivalencia con el nombre completo de las variables puede ser consultada en la tabla 2.
2. En el eje “x” están los SHAP values, los valores mayores a 0 contribuyen a predecir la clase positiva y los valores menores a cero corresponden a variables que apoyan la decisión de la clase negativa.
3. Cada punto circular representa una muestra del conjunto de test y el color denota su valor de salida entre 0 y 1. Cabe mencionar que para las variables que son de tipo categóricas (ver tabla 2), se muestra entre paréntesis la categoría que está siendo analizada en la gráfica.

De acuerdo con el resultado, las 3 variables con mayor capacidad de discriminación son clasificación cultivo (Clasf. Clv), tratamiento (Tto.) y dosis, las dos primeras tienen una correlación con la clase positiva. Los casos en los que las variables son más determinantes es cuándo a las prescripciones se les realizó un procedimiento empírico, es decir, si se les hizo un tratamiento sin la información completa y/o definitiva sobre la patología del paciente; este hecho contribuye, en gran medida, a favor de la clase positiva para indicar que es necesaria la revisión manual.

Por su parte, de acuerdo con el análisis las variables que menos contribuyeron a la clasificación de las prescripciones son algunas categorías de la variable *diagnóstico quien prescribe* y varios tipos *antibióticos*, además de la edad y el sexo del paciente.

Como se comentó en la sección 4.2, los SHAP values pueden también usarse para determinar las variables más importantes para la predicción de una muestra. Sin embargo, en este caso seleccionamos muestras en las cuales el modelo hubiera realizado predicciones incorrectas, para mostrar las capacidades de la técnica y evidenciar las variables que impulsaron al modelo hacia la clase incorrecta, en este caso, las variables que están a la izquierda y cuáles intentan direccionar la predicción hacia la clase indicada (las de la derecha). La figura 15 muestra el resultado del análisis para una muestra de clase positiva, que el modelo clasificó como negativa. El color asociado a las variables denota el sentido del impacto de las variables en la decisión, ordenadas por su importancia. La gráfica muestra también la probabilidad de asignación de la muestra a la clase negativa (resaltada en negrita), además del valor base (en inglés base value) que corresponde al valor de umbral obtenido promediando las probabilidades de salida en el conjunto de entrenamiento.

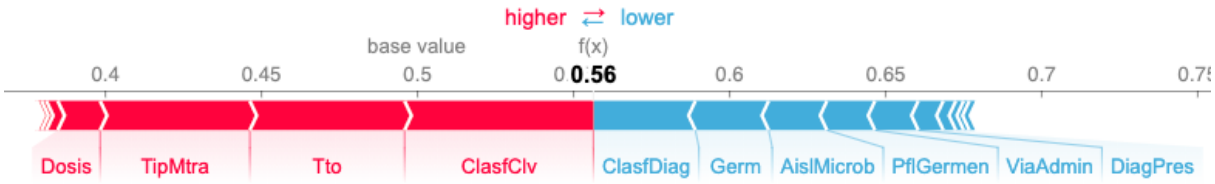

Fig 15. Análisis de los SHAP values en la predicción errónea de una muestra de la clase negativa en el problema 2.

En este caso el modelo arrojó que hay un 56% de probabilidad de que la prescripción no necesita de revisión, debido principalmente al hecho de que la variable clasificación cultivo y tratamiento son de tipo *dirigido*, además de que el tipo de muestra fue *hemocultivos*. Por otro lado, las intervenciones del médico que prescribe el antibiótico como son el diagnóstico final y la vía de administración aportaron a que salida fuera la correcta, sin embargo, el peso de éstas no fue suficiente.

La figura 16 muestra el resultado de un ejercicio similar al anterior, pero para una muestra de la clase negativa. El modelo predice que la muestra necesita revisión con una probabilidad del 70%, apoyado principalmente en las variables clasificación cultivo y tratamiento, que fueron las más influyentes en el análisis global. Es posible que el hecho de que el procedimiento realizado en este caso haya sido empírico y de acuerdo con los patrones identificados por el modelo durante el entrenamiento, dicho procedimiento tiene más probabilidad de que la prescripción requiera revisión.

Figura 16. Predicción errónea de clase negativa para el problema 2

Análisis similares pueden llevarse a cabo para cada una de las muestras, de tal manera que se identifique por parte del equipo de DI, ajustes en los procedimientos de definición de variables, que mejoren la calidad de la información usada para el entrenamiento del modelo.

6. Conclusiones

Se presentó un análisis de la capacidad de predicción que tienen modelos convencionales de ML, para la detección de prescripciones de antibióticos que requieren modificaciones, en el contexto del programa de prevención y control de la proliferación de bacterias resistentes a los antibióticos llevado a cabo por el departamento de infectología de la clínica León XIII y la IPS Universitaria.

Se plantean tres problemas predictivos, dos biclase y otro de múltiples etiquetas, donde en cada problema se evidenció un desbalance notable en la base de datos compartida. Aunque en el segundo de los tres problemas planteados el desbalance fue mayor, con un 94% de las muestras correspondientes a prescripciones que requirieron algún tipo de modificación, el resultado obtenido fue de 81% de BACC y un F1 de 0.94, usando durante el entrenamiento un estrategia de sobre-muestreo. El desbalance presente en la base de datos a favor de la clase positiva (prescripciones que requieren modificaciones), implica que un sistema de ML que alcance tasas de detección cercanas al 100%, podría reducir apenas en un 6% la cantidad de registros que deben ser analizados por el DI. Es necesario verificar con el equipo del DI, si la relación de muestras encontrada en la BD obedece a un subregistro de las prescripciones que no requieren modificaciones, o si el comportamiento estadístico del fenómeno es precisamente el evidenciado en la BD.

Los niveles de acierto observados por los modelos evaluados, no hacen viable el despliegue de este tipo de modelos como solución al problema de análisis y filtración preliminar de los registros de prescripciones. Es necesario determinar si pueden incluirse nuevas fuentes de variables, como información de la anamnesis de la histórica clínica del paciente e información de ayudas diagnósticas. No se descarta que sea necesario adicionar otras variables de interés que los expertos analizan durante el proceso de revisión de las prescripciones, pero que no quedan consignadas en la base de datos.

Se llevo a cabo un análisis de importancia de variables usando la técnica de SHAP values, lo que permitió evidenciar las variables con mayor capacidad de discriminación según el mejor modelo encontrado. La técnica tiene gran potencial para apoyar el proceso de refinamiento de la base de datos, teniendo en cuenta que puede ser usada para identificar las variables relevantes para cada muestra de manera individual. En caso de que se puedan adelantar procesos que conduzcan a la obtención de modelos con mayor exactitud, los SHAP values permitirán guiar a los profesionales de DI para el análisis de prescripciones con alta probabilidad de requerir un ajuste, identificando las variables que determinaron dicho resultado por parte de los modelos.

En este mismo escenario, el análisis de variables también podría permitir identificar las variables que no realizan un aporte relevante al proceso de clasificación y que por lo tanto podrían ser eliminadas del proceso de caracterización, disminuyendo la carga computacional de un posible sistema de información que se desarrolle para desplegar una solución como la aquí analizada. A este respecto, los análisis preliminares indicaron que la variable *diagnóstico quien prescribe* y antibiótico presentan varias categorías con poca capacidad discriminante, se sugiere por lo tanto usar únicamente las agrupaciones definidas durante la primera etapa del proceso.

Bibliografía

[1] Chen, T., and Guestrin, C. “Xgboost: A Scalable Tree Boosting System,” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, Aug. 13–17. 2016

[2] Breiman, L. “Random Forests”. *Machine Learning.* Kluwer Academic Publishers vol 45, pp 5–32. 2001

[3] Bishop, Ch. “Neural Networks For Pattern Recognition”. USA: *Oxford University Press, Inc.*, pp 116-150. 2005

[4] Evgeniou T., Pontil M. “Support Vector Machines: Theory and Applications”. In: Paliouras G., Karkaletsis V., Spyropoulos C.D. (eds) Machine Learning and Its Applications. ACAI 1999. Lecture Notes in Computer Science, vol 2049. 2001

[5] Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions." *31st Conference on Neural Information Processing Systems (NIPS 2017),* Long Beach, CA, USA. 2017.

[6] Zeng, Xinchuan & Martinez, Tony. “Distribution-Balanced Stratified Cross-Validation for Accuracy Estimation”.*Journal of Experimental & Theoretical Artificial Intelligence*. vol. 12, no. 1, pp. 1-12, 2000.

[7] Szymański, Piotr y Kajdanowicz, Tomasz. “A Network Perspective on Stratification of Multi-Label Data". *In First International Workshop on Learning with Imbalanced Domains: Theory and Applications, LIDTA 2017, 22 September 2017, ECML-PKDD, Proceedings of Machine Learning Research*, Skopje, Macedonia, vol. 74, pp 22–35, 2017.

[8] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall y W. Philip Kegelmeyer. “SMOTE: Synthetic Minority Over-sampling Technique”. *Journal of Artificial Intelligence Research 16*, pp *321–357*. 2002

[9] Ceren Gök, Elif y Onur Olgun, Mehmet. “SMOTE-NC and gradient boosting imputation based random forest classifier for predicting severity level of covid-19 patients with blood samples”. *Neural Computing and Applications*, vol. 33, pp 1-15, 2021

[10] Haibo He, Yang Bai, E. A. Garcia y Shutao Li, “ADASYN: Adaptive synthetic sampling approach for imbalanced learning”. *IEEE International Joint Conference on Neural Networks.* Hong Kong, China. 2008

[11] R. Venkatesan and M. J. Er, "Multi-label classification method based on extreme learning machines," 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), 2014, pp. 619-624

[12] A. Kanehira and T. Harada, "Multi-label Ranking from Positive and Unlabeled Data," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA, pp. 5138-5146. 2016

[13] Kohavi, Ron y Wolpert, David. “Bias Plus Variance Decomposition for Zero-One Loss Functions”. *Machine Learning: Proceedings of the Thirteenth International Conference*. San Francisco,CA, USA, pp 275–283. 1997.