Técnicas de Ensamble Aplicadas a un Conjunto de Datos Perteneciente a Pacientes de Leishmaniasis Cutánea para Predecir la Efectividad del Tratamiento Glucantime

K. Camacho Calderon[1]

[1] *Pontificia Universidad Javeriana Cali, Facultad de Ingeniería, Ingeniería de Sistemas y Computación, Cali Colombia.*

Resumen

El glucantime es uno de los medicamentos usados para tratar la leishmaniasis cutánea, la cual ha sido una enfermedad reemergente en Colombia. La forma de administrar este medicamento en algunas ocasiones ha causado mucho dolor y el hecho de usarlo trae la posibilidad de causar efectos colaterales. El anterior panorama ha sido de inspiración para desarrollar modelos de aprendizaje automático que permitan predecir el desenlace terapéutico del tratamiento glucantime. Este artículo describe la preparación del conjunto de datos, la construcción de los modelos usando técnicas de ensamble y las evaluaciones de los mismos, donde se muestran los resultados obtenidos, los cuales en este experimento no fueron positivos dado que ningún modelo presentó un desempeño que permita confiar en sus predicciones, por lo que también se discute en este artículo que la cantidad de datos no fue suficiente para que los modelos establecieran hipótesis fuertes.

Palabras claves: Glucantime, leishmaniasis cutánea, modelos de aprendizaje automático y técnicas de ensamble.

1. Introducción

El glucantime el cual es uno de los medicamentos usados para tratar la leishmaniasis cutánea, es administrado a través de inyecciones, lo cual resulta doloroso para muchos pacientes. El uso de este medicamento puede causar efectos colaterales y no siempre es efectivo. Es así como dicha falta de efectividad del tratamiento y el poco interés hacia el estudio de esta enfermedad ha llevado a que el personal de salud de las zonas endémicas tenga poca experiencia sobre los diagnósticos.

El anterior panorama inspiró a desarrollar el experimento en el que se construyeron modelos de aprendizaje automático que permiten predecir cuándo el tratamiento glucantime para un paciente funcionará y cuándo no, dadas ciertas características sociodemográficas. El conjunto de datos fue suministrado por CIDEIM, el cual es un centro de investigación, desarrollo tecnológico y formación de recurso humano en el campo de la salud.

Este artículo describe el proceso que se llevó a cabo para la construcción de los modelos. El proceso se conformó por las siguientes tareas: Preparación del conjunto de datos, construcción de los modelos usando técnicas de ensamble y las evaluaciones de los mismos. Adicionalmente se discuten los resultados y se mencionan algunas conclusiones que se lograron obtener luego de realizar el experimento.

2. Fundamentación teórica

Como punto de partida es importante la definición de los siguientes conceptos:

- **Modelos de aprendizaje automático:** Son la salida de información que se genera cuando se entrena un algoritmo de aprendizaje automático con datos. Este modelo permite reconocer determinados tipos de patrones y entregar predicciones sobre estos[1].
- **Técnicas de ensamble:** Son técnicas que permiten la construcción de modelos a través de múltiples algoritmos de aprendizaje. Cada algoritmo produce una predicción distinta, dichas predicciones se combinan para obtener una única predicción[2].
- **Técnica de bagging:** El principal objetivo de bagging es sacar provecho de forma paralela de los algoritmos simples, reduciendo así el error de las predicciones al combinar los resultados de los modelos construidos.[2]
- **Técnica de boosting:** En esta técnica la idea es que cada modelo intente arreglar los errores de los modelos anteriores[2]. Es así, que saca provecho al utilizar algoritmos simples de forma secuencial para mejorar la predicción final. Los siguientes son algoritmos de boosting conocidos por tener éxito en problemas de clasificación binaria:

  *__Gradient Boosting Classifier:__ Utiliza árboles de decisión de forma secuencial para producir predicciones precisas. Para determinar los parámetros que usará cada árbol de decisión se usa el algoritmo de descenso de gradiente que a través de derivadas tiene como objetivo minimizar la función de pérdida.

  *__AdaBoost Classifier:__ Generalmente usa árboles de decisión y la estrategia es aplicar el estimador base n veces al conjunto de entrenamiento, de tal forma que en cada iteración tome en cuenta aquellos casos que dicho algoritmo base ha clasificado incorrectamente, aumentando los pesos de los ejemplos mal clasificados.

- **Técnica de stacking:** Stacking usa la predicción de varios modelos diferentes de forma secuencial y paralela, de tal forma, que dichas predicciones en conjunto con los datos originales permitan crear modelos mucho más completos[2].
- **Leishmaniasis cutánea:** Es una enfermedad cutánea causada por un parásito del género leishmania. Los síntomas incluyen lesiones en la piel, que se desarrollan durante varias semanas o meses después de la exposición. Estas lesiones pueden llegar a ser muy grandes y dar lugar a llagas con un borde elevado, que puede estar cubierto de escamas o de una costra.[3]

3. Resultados

Inicialmente se realizó el proceso de limpieza de datos. El resultado de esta limpieza es un conjunto de datos con las siguientes características:

- Contiene 18 registros y 10 atributos.
- Los 10 atributos representan la siguiente información para cada uno de los pacientes: Género, etnia a la que pertenece, edad, el departamento en el que contrajo la enfermedad, el número de lesiones cutáneas activas, el tiempo de evolución de la primera lesión, el área total de la lesión, la dosis diaria de glucantime y el estado final del paciente, si curó o no lo hizo.
- Contiene 11 registros de personas que curaron definitivamente y 7 registros de personas que tuvieron falla terapéutica.

Luego, se realizó el proceso de construcción de modelos a partir de las técnicas de ensamble y evaluación de los mismos. Las técnicas seleccionadas fueron: Bagging, boosting y stacking.

El anterior proceso se dividió en tres pasos:

1. Selección de parámetros y selección de los posibles valores que podrían tomar según la documentación.
2. Selección de los valores que mejor se ajustaron a los parámetros de dichos algoritmos usando una búsqueda exhaustiva combinada con la técnica de validación cruzada de tres iteraciones.
3. Construcción de los modelos y evaluación de los mismos usando validación cruzada de tres iteraciones.

Después de desarrollar cuidadosamente los pasos 1 y 2, se lograron construir y evaluar los siguientes modelos:

Modelo de Bagging Classifier: El modelo base es K vecinos más cercanos.

Cuadro 1: Valores seleccionados para los parámetros del modelo de Bagging Classifier

Nombre del parámetro	**Valor seleccionado**
n_neighbors	11
weights	distance
algorithm	ball_tree
leaf_size	2
p	4
n_estimators	3
max_samples	12
max_features	3

Modelo de Gradient Boosting Classifier: El modelo base es un árbol de decisión.

Cuadro 2: Valores de los parámetros de Gradient Boosting Classifier

Nombre del parámetro	Valor seleccionado
loss	exponential
learning_rate	0.5
n_estimators	3
criterion	mse
min_samples_split	10
min_samples_leaf	4
max_depth	5
max_features	3

Modelo AdaBoost Classifier: El modelo base es un árbol de decisión.

Cuadro 3: Valores seleccionados para los parámetros del modelo AdaBoost Classifier

Nombre del parámetro	Valor seleccionado
criterion	entropy
splitter	best
max_depth	7
min_samples_split	8
min_samples_leaf	1
max_features	5
n_estimators	5
learning_rate	0.01

Modelo Stacking Classifier: Los modelos del primer nivel son: Máquinas de vectores de soporte, K vecinos más cercanos y árbol de decisión. El modelo usado para el segundo nivel es: Regresión logística.

Cuadro 4: Valores de los parámetros del algoritmo de Stacking Classifier

Nombre del parámetro	Valor seleccionado
C	1.0
kernel	linear
gamma	0.0001
degree	2
n_neighbors	3
weights	distance
algorithm	ball_tree
leaf_size	2
p	9
criterion	gini
splitter	best
max_depth	9
min_samples_split	6
min_samples_leaf	1
max_features	5
solver	newton-cg
cv	3
passthrough	True

Se evaluó cada uno de los modelos construidos usando validación cruzada de tres iteraciones, y el desempeño que mejor alcanzaron los modelos durante cada una de las tres iteraciones mencionadas fue:

Cuadro 5: Resultados de la evaluación de los modelos de ensamble

Modelo de ensamble	**F1 score**
Bagging Classifier	0.13
Gradient Boosting Classifier	0.67
AdaBoost Classifier	0.47
Stacking Classifier	0.27

Discusión y Conclusiones

Se logró seleccionar la representación de la información para facilitar el desempeño de los algoritmos que usan las técnicas de ensamble, luego se realizó la estimación de los parámetros y sus valores para cada uno de los algoritmos, lo que permitió finalmente contruir modelos de ensamble para predecir el desenlace terapéutico del tratamiento de glucantime para la leishmaniasis cutánea.

Luego, se realizaron las evaluaciones de los modelos construidos las cuales permitieron inferir que ningún modelo presentó un desempeño que permita confiar en sus predicciones, esto dada la cantidad de datos que no permitió construir modelos diversos. El modelo que presentó el mejor desempeño fue Gradient Boosting Classifier, sin embargo, la métrica indica que aún le falta por establecer hipótesis un poco más fuertes para poder entregar predicciones confiables.

A pesar de que los resultados no fueron favorables, se logró concluir que los algoritmos de aprendizaje automático juegan un papel muy importante dentro de este tipo de investigaciones, y que las técnicas de ensamble pueden resultar estrategias bastantes interesantes con una mayor cantidad de datos.

Referencias

[1] https://docs.microsoft.com/es-es/windows/ai/windows-ml/what-is-a-machine-learning-model.

[2] https://www.iartificial.net/ensembles-voting-bagging-boosting-stacking/.

[3] MJ Hidalgo Solís, KF Víquez Redondo, SM Barrantes Valverde. ”Leishmaniasis cutánea”. Rev.méd.sinerg. 1 de mayo de 2021;6(5):e674.