Inteligencia Artificial y Aprendizaje Automático

Actividad Semanas 5 y 6: Riesgo Crediticio

Maestría en Inteligencia Artificial Aplicada — Tecnológico de Monterrey
Prof. Luis Eduardo Falcón Morales

Nombres: _____ Matrículas: _____

Esta Tarea deberá resolverse en equipos.

Esta actividad se complementa con el archivo "**MNA_IAyAA_semana_5_y_6_Actividad _sept_2024.ipynb**" que se encuentra en Canvas y donde deberán ir respondiendo los ejercicios. Al final deberán entregar la liga de GitHub donde se encuentra el archivo JupyterNotebook con las respuestas y el nombre de los dos miembros del equipo.

El asignar un crédito conlleva un riesgo para el prestamista en caso de que el deudor no pague al final la cantidad asignada, o bien, al equivocarnos en negarle el préstamo a alguien que sí era confiable. Durante décadas se ha tratado de resolver dicho problema desde muchas áreas del conocimiento y en particular las técnicas de Aprendizaje Automático (Machine Learning) han brindado y siguen proporcionando nuevas formas de enfrentar estos problemas.

Existen pocas bases de datos abiertas bien documentadas sobre este problema. Una de ellas son los datos de la página de la UCI llamada "South_German_Credit" y sobre la cual se ha hecho mucha investigación en torno a la asignación de créditos. En esta actividad trabajarás con los datos del archivo "**SouthGermanCredit.asc**", el cual se encuentra dentro del archivo **south+german+credit.zip** que puedes descargar de la liga : https://archive.ics.uci.edu/dataset/522/south+german+credit

En esta actividad nos proponemos tratar de obtener los mejores resultados que se reportan en la Tabla_12 del siguiente artículo de la IEEE:

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9239944

En este artículo se comenta que los datos que utilizaron no son exactamente que los que se encuentran en la UCI. Difieren un poco por cuestiones de privacidad. Sin embargo, por la selección aleatoria se espera obtener resultados análogos en la actividad de esta semana.

Como comentario sobre los datos que utilizaremos, estos datos llamados "South-Geraman-Credit" son una actualización mejorada de otros previos que se estuvieron usando durante décadas para investigación, pero como estaban en idioma alemán, no se habían percatado de varios errores que se habían generado al codificar las variables.

1. Descarga los datos, los cuales nos llevan a un arreglo de 1000 registros y 21 variables. Cambia los títulos de las columnas al nombre en inglés (originalmente están en alemán). La información la puedes encontrar en cualquiera de las ligas dadas arriba.
2. Contrario a lo que sucede en analítica de datos, la clase mayoritaria de los buenos clientes están etiquetados con el valor de 1 y los malos clientes con el valor de 0. Como este no es el proceder dentro del área de ciencia de datos, aplica alguna transformación de manera que la clase negativa (mayoritaria) de los buenos clientes estén etiquetados con el valor de 0 y los malos clientes o clase positiva (minoritaria), estén etiquetados con el valor de 1.

3. El propósito de esta actividad es obtener un modelo que iguale o mejore los modelos del artículo de la IEEE. Así que haremos una partición solamente en dos partes: el conjunto de Entrenamiento (Train) y el de Prueba (Test), con las mismas características que se siguieron en el artículo.
    a. Para ello realiza una partición en Entrenamiento y Prueba con los mismos porcentajes que se utilizaron en el artículo. Llama a las variables Xtrain, ytrain, Xtest, ytest.
    b. Con base al porcentaje de los niveles de la variable de salida ¿podemos decir que tenemos un problema de datos desbalanceado? ¿Por qué?
4. Utiliza la información de la Tabla_3 del artículo para identificar y definir las variables de entrada en numéricas (quantitative), ordinales (ordinal) y las nominales (categorical, binary).
5. Utiliza las clases Pipeline() y ColumnTransformer() de Sklearn para definir y conjuntar las siguientes transformaciones:
    a. A las variables numéricas aplica la misma transformación de normalización que se comenta en el artículo que se aplicó.
    b. A las variables nominales aplicar la transformación One-Hot-Encoding con k-1 niveles.
    c. A las variables ordinales aplicar la transformación OrdinalEncoder.
6. Como vamos a utilizar validación cruzada, concatena los conjuntos de entrenamiento y validación en un nuevo conjunto llamado traintest, que tendrá el mismo número de columnas, pero con el total de renglones la suma de ambos.
7. En la Tabla_12 del artículo se muestran los mejores resultados de uno de sus mejores modelos encontrados para el caso de los datos South-German. Considerando la métrica G-mean como la métrica que mida el desempeño del modelo buscado, entonces el último renglón de dicha tabla nos estaría dando los desempeños obtenidos con su mejor modelo de todas las métricas que utilizaron, a saber: Accuracy, Precision, Recall, F1, ROC-AUC, G-mean. Observa que todas son valores arriba del 80% y la ROC_AUC de 0.9. En este ejercicio deberás obtener el mejor modelo (Logística, kNN, DecisionTree, RandomForest, XGBoost, MLP, SVM) con la mejor técnica de submuestreo y/o sobremuestreo, cuyo valor de G-mean sea el mejor posible. Considera los valores del artículo como los valores a alcanzar. Reporta tu mejor resultado a continuación:

    Mejor modelo con los valores de hiperparámetros encontrado:

    Técnica de submuestreo o sobremuestreo utilizada (en caso de que se haya utilizado):

    Reporte de valores de métricas:

Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	ROC_AUC	G-mean

NOTA: Puedes consultar la siguiente liga para las diferentes técnicas de submuestreo y/o subremuestreo:

https://imbalanced-learn.org/stable/references/over_sampling.html

8. Incluye tus conclusiones finales de la actividad.