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 parejas.

Esta actividad se complementa con el archivo "**MNA_IAyAA_semana_5_y_6_Actividad _abril_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 ese mismo archivo zip se encuentran los archivos **codetable.txt** y **read_SouthGermanCredit.R**, donde puedes encontrar información más detallada sobre el significado y tipo de cada variable.

En la página de Kaggle también puedes encontrar información adicional de estos datos:

https://www.kaggle.com/competitions/south-german-credit-prediction/overview

Pero sobre todo puedes apoyarte en el estudio publicado en el siguiente artículo de la IEEE:

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

Estos datos son una actualización de unos 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.

Parte I: Partición, análisis y pre-procesamiento de los datos.

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 para invertir dichos valores, de manera que en lo sucesivo la clase negativa y mayoritaria de los buenos clientes estén etiquetados con el valor de 0 y los malos clientes o clase positiva y minoritaria, estén etiquetados con el valor de 1.
3. Realiza una partición de los datos en los conjuntos de entrenamiento, validación y prueba, del 70%, 15% y 15%, respectivamente. Con base al porcentaje de los niveles de la variable de salida ¿podemos decir que tenemos un problema de datos desbalanceado? ¿Por qué?
4. El tipo de variable en que se puede clasificar un factor depende en ocasiones del tratamiento que le da el analista. En nuestro caso, siguiendo la información dada en las referencias de la base de datos South-German-Credit, las clasificamos de acuerdo a como se indica en el Jupyter-Notebook de esta actividad. A partir de las referencias dadas y de la información que puedas obtener de los datos, indica el significado de cada una de las 21 variables. En particular, para todas las variables categóricas deberás indicar el número de niveles que tiene cada una.
5. Utilizando el conjunto de entrenamiento de las variables numéricas solamente, realiza un análisis descriptivo sobre este conjunto de datos e indica el tipo de transformaciones que sería conveniente aplicar. NOTA: Utilizaremos la clase Pipeline de Sklearn para aplicar dichas transformaciones en un siguiente ejercicio, por lo que aquí solamente tienes que mencionar las transformaciones que has decidido aplicar a cada variable numérica.

Parte II: Modelos de aprendizaje automático con los conjuntos originales de la partición realizada.

6. Utiliza las clases Pipeline() y ColumnTransformer() de Sklearn para definir y conjuntar las siguientes transformaciones:
    a. A las variables numéricas aplica las transformaciones que hayas decidido en el ejercicio 5.
    b. A las variables nominales y binarias aplicar la transformación One-Hot-Encoding con k-1 niveles.
    c. El resto de las variables dejarlas por el momento sin transformar.
7. Como vamos a utilizar validación cruzada, concatena los conjuntos de entrenamiento y validación en un nuevo conjunto llamado trainval, que tendrá el mismo número de columnas, pero con el total de renglones la suma de ambos.
8. En este ejercicio deberás encontrar los mejores hiperparámetros de cada modelo. Recuerda que debes buscar que no esté sobreentrenado (overfitting) o subentrenado (undefitting) cada uno de los modelos. En este ejercicio diremos que un modelo está sobreentrenado, si con respecto a la métrica de la exactitud (accuracy), la diferencia entre el conjunto de entrenamiento y el de validación es mayor al 3%.
9. De acuerdo a la información dada en las referencias de los datos South-German-Credit, se tiene una matriz de costo que pondera diferente los Falsos Positivos y los Falsos Negativos. Ver: https://archive.ics.uci.edu/dataset/144/statlog+german+credit+data Tomando en cuenta dicha información, contesta las siguientes preguntas con base al problema:

a. ¿Qué error se considera que es el más costoso por parte del banco?

b. ¿Cuál (o cuáles) consideras que sería entonces la métrica (o métricas) a considerar como la más importante para monitorear el desempeño de los modelos?

   NOTA: Existe muchas más métricas que puedes ir investigando en la siguiente página de Sklearn: https://scikit-learn.org/stable/modules/model_evaluation.html

10. Obtener un diagrama de caja y bigotes (boxplot) múltiple de todos los modelos utilizando los resultados obtenidos con la métrica de la exactitud (accuracy) y el conjunto de validación. Es decir, en un mismo gráfico deben estar los siete diagramas de caja. Incluye tus conclusiones al respecto, en particular indica cuáles consideras son los mejores modelos obtenidos.

Parte III: Modelos con técnicas para clases no balanceadas

11. De manera análoga a lo realizado en el ejercicio 8, agrega ahora una técnica de sobremuestreo y/o submuestreo para clases no balanceadas que consideres adecuada para entrenar y desplegar nuevamente todas las métricas y modelos que se utilizaron en la Parte II.

    NOTA: Solo debes incluir la mejor que encuentres. Al menos debes probar los modelos SMOTE y SMOTEENN, pero puedes consultar la página de Sklearn y seleccionar alguna otra técnica que consideres adecuada: https://imbalanced-learn.org/stable/references/over_sampling.html . Igualmente te puedes apoyar en los resultados presentados en la investigación del artículo de la IEEE que se incluye en las referencias dadas al inicio de este documento para considerar algunas de las técnicas de submuestreo o sobremuestreo que dichos autores encontraron como las que les dieron los mejores resultados.

12. Obtener nuevamente el diagrama de caja y bigotes múltiple, pero utilizando ahora la métrica que consideraste en el ejercicio 9 es la mejor para medir el desempeño de los modelos.

Parte IV: Mejor modelo

13. Con base a la información obtenida hasta ahora, selecciona y justifica cuál consideras es tu mejor modelo y con cuál métrica.

14. Con el mejor modelo seleccionado y sus mejores hiperparámetros, utiliza ahora (por primera vez en la actividad) el conjunto de prueba (test set) para:

    a. Mejor modelo con los mejores hiperparámetros y conjunto de prueba.

    b. Obtener la matriz de confusión y la salida del classification_report().

    c. Realiza un análisis de importancia de variables (feature importance) de este mejor modelo e incluye tus conclusiones al respecto.

15. Incluye tus conclusiones finales de la actividad.