Las actividades que se mencionan se enmarcan en lo que es una continuación y ampliación del siguiente trabajo:

> ***Predictor de deserción universitaria***
>
> ***Giselle V. Romero, Joaquín S. Toranzo Calderón, Sebastián E. Jaremczuk, Juan C. Gómez, Claudio Verrastro***

Luego de explorar los dataset disponibles y analizar los procedimientos y resultados obtenidos en la investigación previa del GIAR, se abocó a encontrar nuevas aristas que incluir en el abordaje de la temática, para tratar de complementar y ampliar el análisis realizado hasta el momento.

De esta manera, es que haciendo uso del mismo set de datos se lograron construir dos nuevas variables las cuales se incorporaron al análisis en pos de intentar enriquecerlo.

La primera de estas fue la denominada como "*Distancia*", la cual representa la distancia en kilómetros que existe entre el domicilio declarado del alumno (dato que se construye a partir de los datos presentes en el dataset "*Datos-Alumnos-SIGA*") y la sede de la facultad. Dado que la carrera Ingeniería en Sistemas se cursa en ambas sedes de la FRBA, se decidió fijar como locación de la misma el punto medio entre ambas sedes. El mismo se ubicó en Av. Directorio 1150, CABA.

Para aquellos alumnos que tuviesen domicilio en CABA, pero se desconociera el barrio de dicha ciudad en el que residían se decidió no desestimar el registro y dar un tratamiento particular. La Ciudad Autónoma de Buenos Aires puede inscribirse en un círculo de 10 km de radio. La sede Medrano de la UTN FRBA se encuentra en la zona de mayor densidad poblacional de la Ciudad Autónoma de Buenos Aires. El punto medio entre ambas sedes se encuentra a solo 1,7 km del centro geométrico de la ciudad y dentro de la comuna 6, una de las cuatro comunas más densamente pobladas de la capital del país. Estos datos nos permitieron pensar que los alumnos que viven en CABA tienen mayor probabilidad de encontrarse en las zonas aledañas a la universidad y no en las periferias de la ciudad. Al calcular la media de la distancia de aquellos registros en los que sí se contaba con información completa respecto a en qué parte de CABA residían los alumnos, se consideró que la mencionada presunción era acertada. La media para aquellos alumnos que habitaban en CABA era de 5,1 km. Se decidió entonces que aquellos registros en los que solo se tenía la información de que vivían en CABA, pero se desconocía el barrio, tuvieran como *"Distancia"* la media previamente calculada: 5,1 km.

Por otra parte, se notó que existieron casos en los que la variable "*Distancia*" tomó un valor por encima de los 75 km. Estos representan un 5,32% respecto del total de observaciones (4558). Dado que se cree poco probable que un alumno viva a más de 75 km lineales de la universidad y se desplace a ella cotidianamente (durante el período analizado no habían cursadas virtuales), se consideró que esos datos no estaban actualizados. Es habitual que alumnos originarios de ciudades no comprendidas por el AMBA se desplacen hacia la capital del país durante la época de clases y habiten allí. Por ello, se decidió considerar que estos alumnos (con "*Distancia*" > 75) habitaron CABA durante el período de clases, pero no actualizaron su domicilio. Se les asignó, por ende, como distancia a la universidad la media previamente calculada para quienes habitaban en la capital del país: 5,1 km.

La otra variable que se creó e incorporó al análisis fue la denominada "*grupo_ingreso_nivel1*". Esta le asigna a cada estudiante un grupo de ingreso basándonos en la cursada de la materia integradora de primer nivel "*Sistemas y Organizaciones*" durante su año de ingreso.

Aquellos grupos de ingreso que contaron con menos de diez alumnos registrados en todo el dataset fueron descartados, dado que se consideró demasiado pequeño como para ser representativo.

Esta variable inicialmente tenía formato no numérico y estaba formada por el Ciclo Lectivo de Cursada, el Curso, la Modalidad y el Turno. Por ejemplo, "*2010 S1022 Anual Mañana*" hacía referencia a un grupo particular de ingreso. Dado que los modelos de machine learning a utilizar requieren que los datos sean de tipo numéricos, se debió transformar el dato. Se identificó a cada grupo con un número entero particular.

Se construyó de esta manera un set de datos que unificaba la información disponible en las tres tablas iniciales. El mismo incluía todas las variables analizadas por la investigación previa más las dos nuevas mencionadas: "*Distancia*" y "*grupo_ingreso_nivel1*". Para la construcción de las variables incluídas en el estudio precedente se siguieron los mismos pasos indicados en el siguiente repositorio: Análisis deserción en ingeniería

Luego, se procedió con el análisis estadístico del set de datos obtenido al momento, el cual contaba con 4558 registros, cada uno de ellos asociado a un alumno particular, y 29 features. Existían aún registros con valores no asignados.

Las variables que presentaban valores nulos fueron solo:

- Distancia: 1,08%
- EsTecnico: 13,32%
- grupo_ingreso_nivel1: 25,01%

Se decidió remover todos los registros que tenían valores nulos y aquellos que en la campo "Cantidad de veces recursada regular" tuvieran un valor por encima de 30.

Al descartar los registros que en "*grupo_ingreso_nivel1*" no tenían un valor asignado, estamos descartando necesariamente a quienes hayan ingresado a la FRBA luego de haber comenzado sus estudios en otra institución. Por ejemplo, un alumno que asistió a la UBA y allí completó y aprobó todas las asignaturas de primer nivel, decide cambiar de casa de estudios y continuar su carrera en la UTN. En este caso, el alumno durante su primer año en la FRBA estaría cursando materias de segundo nivel, por ende, no se le podría asignar un *grupo_ingreso_nivel1* ya que no cursó la materia integradora de primer nivel en esta institución. Por ende, al decidir eliminar todos los registros con nulos, se decide que **solo** van a analizarse aquellos registros correspondientes a alumnos que hayan iniciado la cursada de la carrera en la FRBA y se tenga registro de ello.

Al eliminar los registros que tuviesen algún valor nulo, se notó que no quedaron registros disponibles en los que la feature "*Descripción de recursada regular_Recurso n Veces (>5)*" tomara valor distinto de cero. Por ende, se eliminó dicha variable del dataset.

Quedó así, luego de la limpieza, preprocesamiento, transformación, agrupamiento, un dataset con **3.177 registros** y **29 variables**, entre ellas la dependiente que se quiere predecir.

Posteriormente, se procedió a separar los datos disponibles en los conjuntos de entrenamiento (train) y prueba (test). Se dejaron para test la misma proporción de registros que en el trabajo que se cita como fuente: 21,6% (824 registros de un total de 3814 en el trabajo original). El número de registros con el que contó el set de test fue de **687**.

Acto seguido, tal como en el trabajo citado, se procedió a revisar la relación de los registros de estudiantes contra el total y se observó que quienes desertaron representaban un 38% aproximadamente, lo que implicaba un pequeño desbalance en el dataset. Se recurrió también a la técnica de sobremuestreo para balancear los datos de entrenamiento y validación. Su aplicación implicó la duplicación de 582 registros elegidos al azar de la clase desertor. Se obtuvo finalmente un set de train con un total de 3072 registros balanceados. El set de prueba no se modificó, sino que se mantuvo en la proporción original (37%).

Finalmente, siguiendo los pasos del proyecto de 2021, en el caso de las features numéricas, dado que la diferencia de escalas puede generar problemas a la hora de aprender de los datos, se recurrió a la estandarización de las mismas para evitar que tengan rangos muy distintos. Esto es, se transformó a cada una de estas variables de manera tal que tengan media $\mu = 0$ y desvío estándar $\sigma = 1$.

La etapa siguiente consistió en la construcción del modelo de clasificación binario.

Para éste estudio, el valor 0 en la etiqueta está asociado a los alumnos no desertores, mientras que el valor 1 está asociado al caso contrario.

Existen numerosos métodos de clasificación. En este caso, se optó por evaluar los siguientes:

- **Regresión Logística:** clasificador de tipo lineal. Se basa en una regresión lineal precedida de una función de activación o de decisión “sigmoide”, la cual genera un output binario y no continuo como una regresión normal.
- **Support Vector Machines (SVM):** este método se basa en la idea de separar los datos mediante hiperplanos. El nombre del mismo proviene de la utilización de vectores que hacen de soporte para maximizar la separación entre los datos y el hiperplano.

Para cada uno de ellos, se definieron diferentes hiper parámetros para ser utilizados en el proceso de Grid Search. Este consiste en generar una lista de combinaciones posibles para cada método, para que luego en el proceso de validación cruzada (Cross Validation), se compare cada una de ellas y se compruebe cual de todas las combinaciones es la que mayor Train Accuracy promedio genera. De esta manera, se elige el mejor método de clasificación con sus hiper parámetros identificadores para poder realizar las predicciones pertinentes a futuro.

Entonces, en pos de desarrollar un modelo para predecir la variable “*deserto*” se creó un pipeline de Machine Learning.

Para ello, se recurrió a la herramienta *Pipeline* de la librería *scikit-learn*. Esta permite aplicar secuencialmente una lista de transformaciones y un estimador final. En este caso, como estimador final se tuvieron los modelos de clasificación previamente mencionados: SVM y Logistic Regression.

Se definieron los hiper parámetros a combinar para cada modelo de clasificación a comparar. En el caso del SVM, se emplearon los mismos que se analizaron en el trabajo original de 2021.

Posteriormente, se construyó el Grid Search y Cross Validation. El set de datos de train se divide en 5 porciones durante la validación cruzada.

Al realizar el entrenamiento se obtuvo que, dados los hiper parámetros propuestos, el modelo con el mayor accuracy (**83,11%**) fue **Support Vector Machines** con la siguiente combinación de hiper parámetros:

C: 1000 | kernel: ‘rbf' | gamma: 0.001

Dicha combinación de hiper parámetros es la **misma** que arrojó los mejores resultados en la investigación previa.

El PCA es un método que permite reducir la cantidad de dimensiones (features) de un dataset, creando nuevas features llamadas “Componentes Principales” a partir de la descomposición espectral. Esas nuevas dimensiones serán combinaciones lineales de las features originales creadas con el fin de representar la mayor cantidad de variación de los datos.

Con el objetivo de filtrar el ruido y mejorar la eficiencia computacional se decidió aplicar dicho método. Se definió que los autovectores a extraer debían explicar como mínimo un 80% de la variación de los datos.

Se construyó un nuevo pipeline que incluyera al final del preprocesamiento la implementación del PCA (solo para las variables no categóricas). Las transformaciones iniciales y los modelos de clasificación al final del pipeline, así como también los hiper parámetros a combinar, fueron los mismos que en los utilizados en el pipeline inicial.

Luego de aplicar el preprocesamiento que incluyó el PCA, se pasó de tener una matriz con 28 dimensiones a una con 13.

Posteriormente, se construyó el Grid Search y Cross Validation que se utilizaría para seleccionar el mejor modelo y la combinación de hiper parámetros. Durante la validación cruzada el set de datos de train, al igual que antes, se dividió en 5 porciones.

Al realizar el entrenamiento, se obtuvó que, dados los hiper parámetros propuestos, el modelo con el mayor accuracy (**79,77%**) fue **Support Vector Machines** con la siguiente combinación de hiper parámetros:

C: 1000 | kernel: 'rbf' | gamma: 0.01

Ambos modelos lograron una exactitud mayor a la obtenida por los modelos construidos en el proyecto original.

Esto nos invita a pensar que hay margen de mejora y que la incorporación de más y diferente información podría permitir construir una herramienta más robusta para detectar desertores en forma anticipada.

Actualmente, se siguen analizando y explorando la construcción de otros modelos diferentes a los mencionados.

Por otra parte, se está trabajando en la construcción de modelos de **Redes Neuronales**.

La intención es construir uno que posea una estructura igual a la utilizada en la investigación previa y analizar las variaciones.

Paralelamente, se está explorando y haciendo pruebas con diferentes diseños de estructuras y variando los parámetros épocas, paciencia y learning rate. La red con la que eventualmente nos quedaremos será la que ofrezca el mejor resultado.

En el siguiente repositorio de Github se puede encontrar el trabajo hasta aquí realizado:

https://github.com/cabatedag/desercion_universitaria

La matriz de confusión permite evaluar el desempeño de los clasificadores construidos. La misma muestra qué tan bien cada clasificador categoriza las distintas clases.

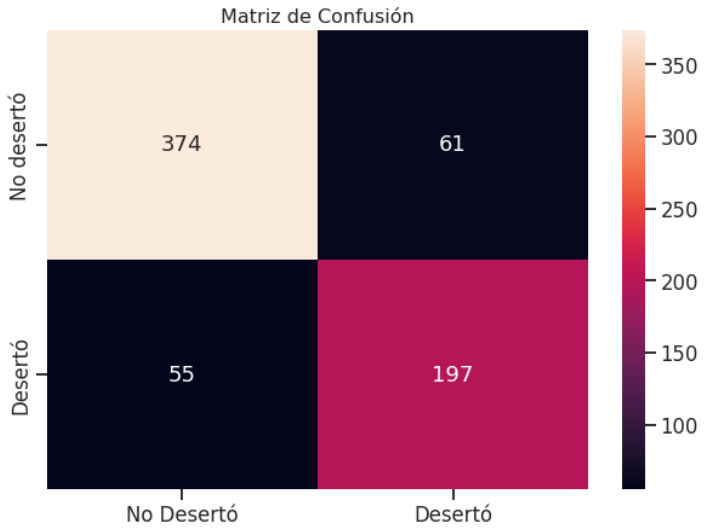

Fig. 1 Matriz de confusión para el primer modelo construido (SVM: C 1000, kernel 'rbf', gamma 0,001).

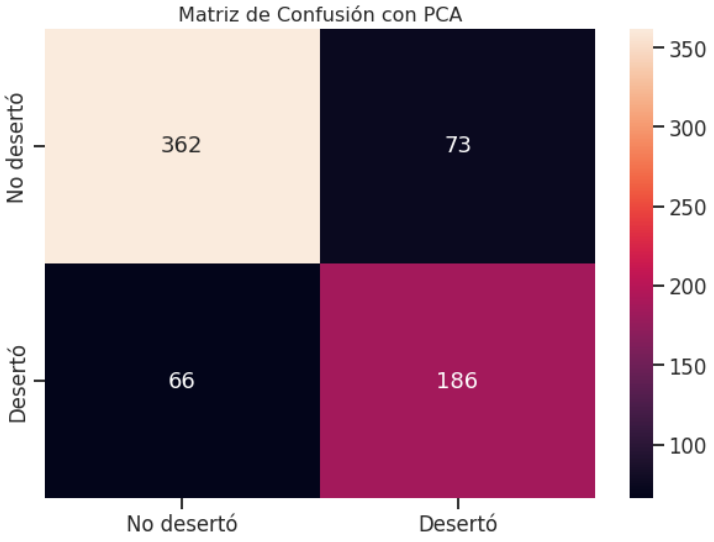

Fig. 2 Matriz de confusión para el segundo modelo construido luego de aplicar el PCA (SVM: C 1000, kernel 'rbf', gamma 0,01).

Se observa que en ambos modelos obtienen valores similares. La exactitud obtenida por el primer clasificador fue **83,11%**, mientras que la del segundo **79,77%**.

Ambos resultados están por encima de los obtenidos en el estudio del año 2021 en el que no se incluyeron las variables "Distancia" y "grupo_ingreso_nivel1"

La implementación del PCA, es decir, la disminución de la dimensionalidad del dataset, permitió mejorar la eficiencia computacional, pero implicó una pequeña merma en el desempeño del modelo que se construyó a posteriori respecto del construido inicialmente.

Se puede también comparar los clasificadores haciendo uso de la curva ROC. Los resultados permiten verificar la reflexión anterior.

Los resultados que se obtienen a partir de cada uno de ellos son considerados buenos.

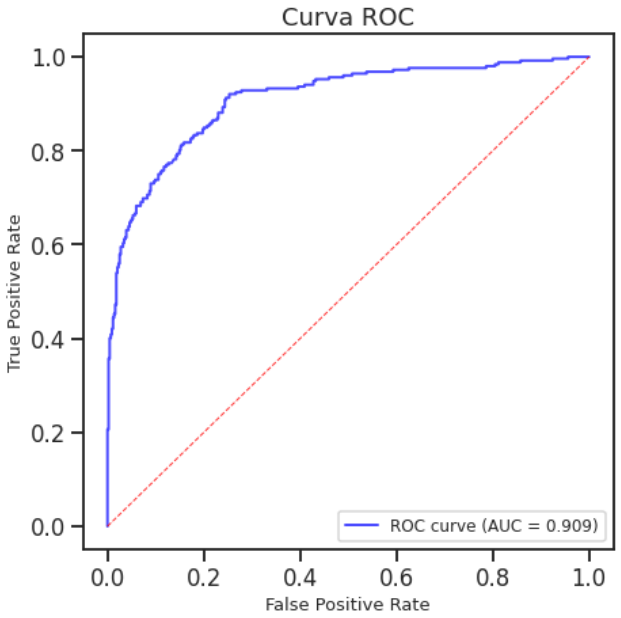

Fig. 1 Curva ROC del primer modelo construido (SVM: C 1000, kernel ‘rbf', gamma 0,001).

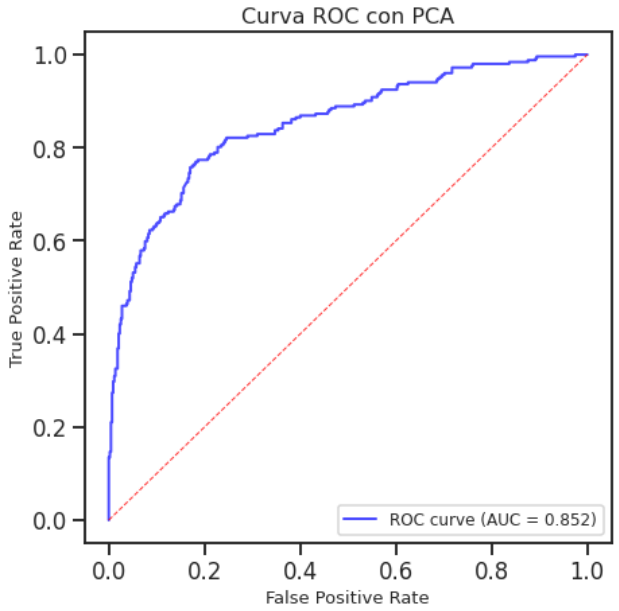

Fig. 2 Curva ROC del segundo modelo construido (SVM: C 1000, kernel ‘rbf', gamma 0,01).

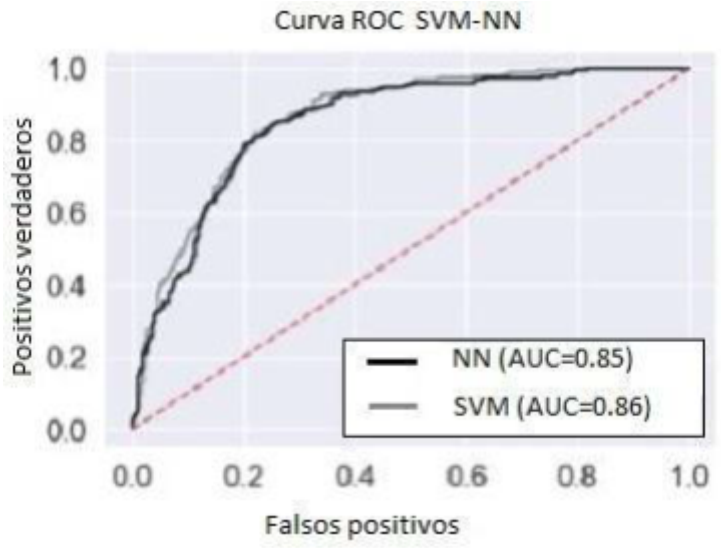

Fig. 3 Curva ROC de los modelos construidos en la investigación de 2021.[1]

Los resultados de exactitud obtenidos para los dos nuevos clasificadores, 89,1% y 89,4% para el primero y el segundo respectivamente, son considerados buenos. Tal como en el trabajo citado, se puede afirmar que la base de datos contiene información suficiente para predecir si un alumno desertará o no.

[1] Romero, G., et al. Predictor de deserción universitaria. Proyecciones-Publicación de investigación y posgrado de la FRBA. Año 19, vol. 1, abril 2021.