**Uso de Algoritmos Genéticos para Definir el Clasificador que Maximice las Métricas de Desempeño**

Jorge Eduardo Enciso Agudelo1, Daniel Figueroa Torres1, Sergio Andrés Rojas Moreno1, Édison Javier Diaz Prieto1, Alexander Luna Ruiz1, Árnol Guerrero Calderón1

1 Sistemas Inteligentes, Maestría en Inteligencia Artificial, Pontificia Universidad Javeriana Bogotá D.C., Colombia

**Resumen.** En este proyecto se quiere usar una herramienta que permita implementar algoritmos genéticos para mejorar el resultado obtenido en la primera entrega del proyecto, donde se pretendía clasificar los diferentes géneros musicales basado en características medidas para cada canción. Se espera que mediante algoritmos genéticos se pueda maximizar las métricas de desempeño del algoritmo, con el fin de obtener mejores resultados a los ya obtenidos con la aplicación de redes neuronales artificiales.

# Motivación y Contextualización del Problema

Dentro de los retos que surgieron de la primera entrega del proyecto de “Sistema de Clasificación de Géneros Musicales para Plataformas de Streaming”, se encontraba el poder encontrar el clasificador más adecuado para para mejorar las métricas de desempeño que se obtuvieron inicialmente. Gracias a los algoritmos genéticos, se puede utilizar una herramienta en Python que permite maximizar dichas métricas de desempeño, por lo tanto, podemos aproximarnos a mejores resultados.

Con esto es posible continuar con el problema inicial de clasificación, pues con mejores métricas se podrá crear un nuevo sistema con la capacidad de segmentar de manera más eficiente los géneros musicales, que actualmente son muy amplios y con características heterogéneas, como en el caso de la música alternativa y la música indie. Claramente, es difícil asegurar que con todos los géneros se va a obtener un resultado muy específico de clasificación, sino que, por el contrario, se quiere llegar a un punto donde se pueda dar un porcentaje de similitud a los diferentes géneros, pues existen muchos géneros basados en otros, por lo que al final, lograr ver esa mezcla sería más eficiente.

## Estado del arte del uso de GA para un problema similar

A partir de la motivación y contextualización del problema descritos en el acápite anterior, se llega a la conclusión que el uso de Algoritmos Genéticos puede resultar beneficioso en cuanto a los resultados que se obtienen, dada la complejidad y magnitud de los datos disponibles. Se decidió hacer uso de esta técnica con referencia en el artículo “Automatic Music Genre Classification Using Hybrid Genetic Algorithms”. Este documento tiene como objetivo desarrollar un sistema de clasificación automática de género musical y se centra en el cálculo de algoritmos que (idealmente) pueden predecir la clase de música a la que pertenece un archivo de música. El sistema propuesto se basa en técnicas de los campos de procesamiento de señales, reconocimiento de patrones y recuperación de información, así como métodos de optimización heurística. Se utilizan mil archivos de música para entrenar y validar el sistema de clasificación. Estos archivos se distribuyen por igual en diez clases. De cada archivo, se extraen ochenta y un (81) características y se utilizan para crear 81 matrices de similitud. Estas 81 matrices de similitud constituyen las instancias de entrenamiento. Durante la fase de entrenamiento, la selección de características se lleva a cabo mediante un algoritmo genético híbrido modificado para mejorar la claridad de discriminación de clases y reducir el costo de cálculo. En este algoritmo, la probabilidad de cruce se reemplaza por un número de par principal que produce nuevas soluciones a través de una lista tabú. Además, se utiliza una mutación adaptativa, una búsqueda exhaustiva local adaptativa y una estrategia de reemplazo adaptativo, dependiendo de si el sistema ha llegado a un máximo local. La búsqueda local exhaustiva se lleva a cabo en los cromosomas vecinos óptimos hasta la solución actual. La función de adecuación del algoritmo genético constituye un clasificador ponderado de vecinos más cercanos. Por lo tanto, la aptitud del cromosoma es proporcional a la precisión del clasificador que crea el cromosoma. Durante la fase de clasificación, las características seleccionadas a través del algoritmo genético crean un clasificador vecino más cercano ajustado que realiza las clasificaciones. De cada nuevo archivo de música pendiente de clasificación, las características seleccionadas se extraen y luego se comparan con las características correspondientes de los archivos de música de la base de datos. El archivo de música se asigna a la clase indicada por los k archivos de música más cercanos (Karkavitsas, George & Tsihrintzis, George, 2011).

### Inteligencia Artificial (IA)

Con el fin de introducir el concepto de Inteligencia Artificial, en adelante (IA), esta puede definirse como “la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano” (Rouhiainen, 2018, p.17), o como interpretase Marvin Minsky: “...diseño de máquinas con el objeto de simular el funcionamiento de la mente humana y con capacidad adaptativa y libertad de actuación” (Minsky, 2010).

A partir de esta definición, y dado que “las diferentes técnicas de IA, tales como los algoritmos genéticos, las redes neuronales, la lógica difusa, las hiper-heurísticas y en general los sistemas de aprendizaje automático (Machine Learning)” (Azar, Tapia, García y Pérez, 2019, p.1), han obtenido resultados importantes en el procesamiento de Big Data y en la resolución de problemas complejos, se propone solucionar el problema de clasificación de música, objeto del presente artículo, a partir de la implementación de Redes Neuronales Artificiales, teniendo en cuenta su capacidad de generalizar información extraída de datos experimentales (Ponce C., 2010).

### Redes Neuronales Artificiales

Se conoce como Red Neuronal Artificial (Por sus siglas en inglés, ANN) a un modelo matemático o computacional basado en el aspecto o estructura funcionales de las redes neuronales biológicas. La ANN se compone de neuronas artificiales (nodos) interconectados entre sí.

Una definición adicional de ANN propone que es “una técnica de aprendizaje automático que es generalmente eficaz para extraer características críticas de conjuntos de datos complejos y derivar una función o modelo que expresa esas características” (Pelchat y Gelowitz, 2019, p. 1).

Las ANN normalmente buscan solucionar problemas no lineales, las cuales generalmente proporcionan salidas continuas y realizan funciones simples. Las ANN tienen entradas de datos las cuales son ensambladas al modelo de acuerdo con sus funciones operativas y producen respuestas teniendo en cuenta su función de activación.

El tipo de modelo más simple de neurona fue propuesto por McCulloch y por Pitts (1943) y sigue siendo el modelo más utilizado en las distintas arquitecturas de las redes neuronales. Desde 1943, se han desarrollado modelos mucho más detallados y realistas, tanto para neuronas como para sistemas más grandes en el cerebro, lo que lleva al campo moderno de la neurociencia computacional. Por otro lado, los investigadores en inteligencia artificial y estadística se interesaron en las propiedades más abstractas de las redes neuronales, como su capacidad para realizar cálculos distribuidos, tolerar entradas ruidosas y aprender. Aunque ahora entendemos que otros tipos de sistemas, incluidas las redes bayesianas, tienen estas propiedades, las redes neuronales siguen siendo una de las formas más populares y efectivas de sistema de aprendizaje y son dignas de estudio por derecho propio (Rusell & Norvig, 2010).

Este modelo se compone por múltiples señales de entradas del ambiente exterior, las cuales estas conectadas a una neurona artificial, cada conexión cuenta con un peso el cual busca similar las conexiones sinápticas de una red neuronal biológica. Este peso busca dar la relevancia que tiene cada entrada para la neurona. El nodo básicamente hace una suma pondera del peso multiplicado por las entradas menos el sesgo, el cual lo que busca es darle el umbral adecuado a la neurona. A este resultado se le aplica una función de activación el cual tiene como objetivo limitar la salida de la neurona dentro de un rango de valores razonable. Por último, el modelo arroja una señal de salida el cual es el valor final producido por la neurona.

Las redes neuronales están diseñadas a partir de una arquitectura que define como se organiza cada neurona entre sí. Existen distintos tipos de arquitecturas, cada una de estas se debe aplicar dependiendo la problemática que se aborde. Entre estas arquitecturas se pueden encontrar: Arquitectura feed-forward de una sola capa, la Arquitectura feed-forward multicapa, la Arquitectura recurrente, entre otros.

En este caso nos enfocaremos en la arquitectura feed-foward multicapa. Esta arquitectura se compone por una capa de entrada la cual se encarga de recibir la información, una o varias capas ocultas dependiendo la complejidad del modelo, esta capa es la encargada de extraer patrones asociados con el proceso o sistema que se está analizando y por último una capa de salida la cual se encargar de dar la salida final de la red, el cual es el procesamiento realizado por los nodos de las anteriores capas. Este tipo de arquitectura normalmente es utilizado para: clasificación de patrones, identificación de sistemas, control de procesos, optimización, robótica, entre otros.

Las ANN se caracterizan porque tienen la capacidad de aprender, el cual consiste en aplicar una serie de pasos ordenados para ajustar el peso y el sesgo de cada neurona, con el propósito de generalizar la solución producida por sus salidas. Existen distintos tipos de aprendizaje entre los cuales esta: el Aprendizaje supervisado, aprendizaje no supervisado, el aprendizaje por refuerzo, entre otros.

Para abordar esta problemática el modelo se acogerá a utilizar el aprendizaje supervisado. Este consiste en aprender de los datos que se le proporcionan al modelo para entrenar. Estos datos deben estar categorizados para que el modelo al momento de entrenar pueda comparar su salida con el resultado esperado y así por medio de la función de perdida y la función de optimización, el modelo pueda ajustar sus parámetros para disminuir el error que existe entre el dato que se predijo con el dato que se esperaba.

### Algoritmos Genéticos

Los algoritmos genéticos son una técnica del área de Sistemas Inteligentes que constituye una abstracción del concepto biológico de evolución natural y tiene numerosas aplicaciones en problemas de optimización. Su funcionamiento está basado en los mecanismos de selección natural, combinando la supervivencia del más apto con un intercambio de información entre miembros de una población de posibles soluciones. Se estima que el uso de algoritmos genéticos puede contribuir a acortar significativamente los tiempos de resolución del problema, ya que se caracterizan por su capacidad de explorar el espacio de búsqueda amplia y eficientemente.

Los algoritmos genéticos fueron desarrollados por John Holland, junto a su equipo de investigación, en la universidad de Michigan en la década de 1970. Éstos combinan las nociones de supervivencia del más apto con un intercambio estructurado y aleatorio de características entre individuos de una población de posibles soluciones, conformando un algoritmo de búsqueda que puede aplicarse para resolver problemas de optimización en diversos campos. Imitando la mecánica de la evolución biológica en la naturaleza, los algoritmos genéticos operan sobre una población compuesta de posibles soluciones al problema. Cada elemento de la población se denomina “cromosoma”. Un cromosoma es el representante, dentro del algoritmo genético, de una posible solución al problema. La forma en que los cromosomas codifican a la solución se denomina “Representación” (ver Figura 1).

Diagrama

Descripción generada automáticamente

*Figura 1. Representación de como cada cromosoma codifica una posible solución al problema*

El algoritmo genético va creando nuevas “generaciones” de esta población, cuyos individuos son cada vez mejores soluciones al problema. La creación de una nueva generación de individuos se produce aplicando a la generación anterior operadores genéticos, adaptados de la genética natural. La Figura 2 representa el esquema de funcionamiento del algoritmo genético. El proceso comienza seleccionando un número de cromosomas para que conformen la población inicial. A continuación, se evalúa la función de adaptación para estos individuos. La función de adaptación da una medida de la aptitud del cromosoma para sobrevivir en su entorno. Debe estar definida de tal forma que los cromosomas que representen mejores soluciones tengan valores más altos de adaptación (Yolis, 2003, p. 39).

Los individuos más aptos se seleccionan en parejas para reproducirse. La reproducción genera nuevos cromosomas que combinan características de ambos padres. Estos nuevos cromosomas reemplazan a los individuos con menores valores de adaptación. A continuación, algunos cromosomas son seleccionados al azar para ser mutados. La mutación consiste en aplicar un cambio aleatorio en su estructura. Luego, los nuevos cromosomas deben incorporarse a la población; estos cromosomas deben

reemplazar a cromosomas ya existentes. Existen diferentes criterios que pueden utilizarse para elegir a los cromosomas que serán reemplazados. El ciclo de selección,

reproducción y mutación se repite hasta que se cumple el criterio de terminación del

algoritmo, momento en el cual el cromosoma mejor adaptado se devuelve como solución (ver Figura 2).

Diagrama

Descripción generada automáticamente

*Figura 2. Esquema de funcionamiento del algoritmo genético*

# En el presente proyecto se decidió referenciar el articulo “Hyper parameter optimization using genetic algorithm on machine learning methods for online news popularity prediction”, donde se busca identificar la noticia con mayor probabilidad de tener popularidad, para que así los consumidores de noticias se vean atraídos. Para ello, en el artículo se explican diferentes métodos de aprendizaje de máquina, donde, sin embargo, se debieron optimizar los hiper parámetros de método aplicado, para determinar cuál es el que tiene la mejor precisión. Para realizar esta optimización, los autores deciden utilizar el método de algoritmo genético con la finalidad de encontrar los mejores hiper parámetros de cada método en el menor tiempo posible. Para esto, primero se debe definir cuales hiper parámetros van a ser modificados para cada modelo. Una vez definidos, luego se define la tasa de cruce, la tasa de mutación, el número de iteraciones, el tamaño de la población y el número de generación. Al correr el método se puedo identificar que le tomaba menor tiempo posible optimizar los hipermer paramatros comparado con grid search.

Gráfico, Gráfico en cascada

Descripción generada automáticamente

# Descripción de la Tarea

Se usa un algoritmo genético para definir el clasificador que maximice las métricas de desempeño del algoritmo. Se parte de un modelo base, el cual fue Naive Bayes con un valor de 0.33 pero el mejor modelo fue gracias al Random Forrest con tfidf en el título de las canciones y un valor de 0.57. Se utiliza la herramienta tpot, exactamente el TPOTClassifier para ejecutar el algoritmo genético haciendo varios intentos con diferentes valores de generaciones y de tamaño de población.

## Visión General

Basado en que actualmente existen muchas canciones en el mundo, existen diferentes características que pueden describir el género. Entre ellas tenemos 'Popularidad', 'bailabilidad', 'energía', 'clave', 'sonoridad', 'acústica', 'instrumentalidad', 'viveza', 'valencia', 'tempo', entre otros. Lo que se puede notar de estas características es que, a pesar de estar relacionadas con la música, no necesariamente se conectan a la hora de describir un género musical porque en la lógica humana, por ejemplo, la popularidad no necesariamente puede definir un género, o la bailabilidad, pues, claramente muchos de estos géneros musicales pueden compartir estas características. En este punto es donde se hace necesario el reto de seleccionar correctamente la información para poder llegar a una clasificación efectiva.

Por otro lado, también vale la pena recalcar que la motivación para realizar este proyecto radica en la importancia de reconocer los estilos propios de los diferentes géneros musicales en los diferentes países. Colombia tiene una cultura musical amplia, diferentes géneros que pueden ser muy genéricos, por ejemplo, la cumbia, pero si comparamos las características que tiene este género en Colombia con el estilo que se puede presentar en Argentina o en México, se presentaran diferencias importantes, lo mismo con un género como la Salsa, pues a pesar de que Colombia tiene varios grupos musicales que manejan el género, no es posible comparar los estilos con países como Cuba, Puerto Rico o Republica Dominicana.

Finalmente, del dataset seleccionado hemos analizado asuntos de similitud en medidas, denotando que un entrenamiento de la red con variables que se comportan de la misma forma no resulta favorable para una correcta clasificación, también se encontró un desbalanceo en las diferentes clases, por lo que se tiene que tomar una decisión importante sobre el número de muestras que hay para cada uno de los géneros o clases que se quieren clasificar, con el objeto de lograr resultados coherentes y favorable

## Restricciones

Las restricciones con la aplicación de los GA en este problema sería que los individuos o soluciones para el modelo lleguen a ser factibles. Es decir que, a través del cruce y la mutación, la factibilidad que se tiene se pueda mantener en todo momento. Adicionalmente, tras la visión general presentada, se pueden notar varias restricciones presentes desde el simple hecho de la selección de los datos de valor del dataset. Inicialmente, se requiere hacer el balanceo de los datos, pues de las diferentes clases se ven unos porcentajes desproporcionados que producen fallas a la hora de querer entrenar la red. Adicionalmente, como parte de un análisis de los diferentes géneros, para los casos de la música alternativa y aquellas canciones clasificadas en el género indie, se ha encontrado que dentro de ese mismo género siempre van a existir subgéneros muy marcados, por lo tanto, la clasificación de un género tan variante resulta compleja en términos de lograr un alto margen de exactitud del modelo, dado que existen otros géneros que presentan características mucho más evidentes.

Adicionalmente, se tiene el caso de los géneros Folk y Country, donde, técnicamente, la música folk es la que emplea instrumentos y formas tradicionales de un territorio para crear nuevos temas musicales, así que, puede haber música folk de cualquier origen y con diferentes características, en tanto reconozca ese origen en los ritmos y los instrumentos que se emplean. Por su parte, la música country es la música folk rural de Estados Unidos, especialmente del centro y oeste del país, es decir, el country representa un subgénero del folk. De lo anterior, se pudo definir que el dataset presenta algunas incongruencias con respecto a la selección de los géneros que se querían clasificar, situación que además hacía parte del reto del hackathon en donde se empleó este set de datos.

## Alcance

El alcance para esta aplicación de GA sería una búsqueda más amplia de algoritmos de machine learning que permitan encontrar unos parámetros que puedan minimizar el error. Dada la limitación de tener un sistema que sea capaz de clasificar géneros tan genéricos como el Folk, Alternativo o Indie, es importante tomar decisiones con respecto al dataset, que realmente permitan tener mayor exactitud en los resultados del modelo planteado. Por lo anterior, realizan varias pruebas con modificaciones leves en la selección de los atributos a tener en cuenta en su entrenamiento y posterior clasificación. Con un conocimiento en el alcance que hubo a nivel del concurso donde originalmente se usó este dataset, se pretende lograr mejores resultados que la media en la exactitud del modelo final, sin embargo, se esperan resultados más favorables con métodos diferentes a las redes neuronales artificiales, como es el caso de los algoritmos genéticos.

# Análisis de Viabilidad de Validación Experimental

A continuación, se describen los métodos empleados para el tratamiento de los datos, preprocesamiento, selección de características, aplicación de algoritmos de machine learning, maximizando la métrica de precisión usando algoritmos genéticos y validación en el proceso de clasificación de los géneros musicales de acuerdo con las características seleccionadas.

## Selección de la base de datos.

Para la selección de la base de datos, se tomó un dataset de un concurso tipo ‘hackatón’ de la página machinehack.com, en asociación con la plataforma de concursos Kaggle, cuyo objetivo es la realización de concursos para la comunidad de científicos de datos, de tal manera que puedan aplicar los conocimientos relacionados a la disciplina de ciencia de datos.

El dataset es un extracto de las canciones más famosas en el año 2020, que se tomó como referencia para el concurso. El dataset de entrenamiento consiste en 17,000 registros que contienen atributos de las canciones. El dataset está compuesto por 14 variables numéricas que describen características específicas de la canción como la instrumentalidad, el tempo, entre otros, y fueron extraídas a priori haciendo uso de software sofisticado de edición de música como Premier Studio. Por otro lado, también se incluyeron variables como popularidad de la canción, la cual fue obtenida directamente de la plataforma que suministró la base de datos. Además de estas variables cuantitativas, se incluyeron 2 variables cualitativas: nombre de la canción y autor, más la variable de respuesta que es el género de las canciones.

## Preprocesamiento de los datos

Para la limpieza de los datos se realizó una exploración, cuyo objetivo consistió en encontrar datos atípicos e identificar otros faltantes. En primera instancia, se identificó alrededor del 5% de datos faltantes en la variable del ranking de popularidad y en la variable de instrumentalidad.

Por otro lado, también se consideró generar un balance de las clases debido a que existe una cantidad desproporcional entre la clase mayoritaria en comparación a la minoritaria.

### Imputación de valores

Debido a que el porcentaje de faltantes no era significativamente alto, se decidió imputar los valores. En este caso se ejecutó un ‘Random Forest’ con entrada de las otras variables para poder predecir los registros faltantes.

### Estandarización de las variables

Con el fin de utilizar algoritmos de Machine Learning y de Deep Learning, se decidió estandarizar todas las variables cuantitativas usando la inversa de una función uniforme, o como bien se conoce, usando el algoritmo Min Max.

## Exploración de variables

Para esta sección se un análisis por variable y entre variables para poder encontrar datos atípicos y relaciones entre variables que puedan mejorar el desempeño de los modelos. En la tabla de correlaciones se puede apreciar que la mayoría de las variables no tiene una correlación lineal. A excepción de la variable de energía y de ruido(loudness) que tienen un coeficiente de Pearson de 0.77.

Chart, histogram

Description automatically generated

*Tabla 1. Correlación entre las variables*

Debido a que este problema es de multi clase y contiene 10 posibles etiquetas, un F1 score de 0.5 es muy optimista, ya que no se está tratando con un modelo binario. Por esta razón, se propuso tomar como línea base un modelo de predicción simple o ingenuo que no tenga gran cantidad de híper parámetros para ajustar. En este caso se optó por un modelo de ingenuo bayes, el cual está basado en probabilidad condicional. Además, se incluyó un modelo más sofisticado para que se tenga una idea del comportamiento de los datos bajo otro modelo.

Calendar

Description automatically generated

Calendar

Description automatically generated with medium confidence  


En las tablas 2 y 3, se aprecia que los resultados de los modelos iniciales o base tienen un F1 Score de 0.31 y 0.47, lo cual implica que los modelos que se generen a partir de técnicas de aprendizaje profundo y más adelante con algoritmos genéticos deben mejorar estos resultados.

## Implementación de los algoritmos genéticos

Después de contar con una línea de base, se decidió hacer una optimización usando algoritmos genéticos como método para optimizar el F1 Score general. De este modo, se implementaron algoritmos genéticos usando la librería de Python TPOT, que permite desarrollar algoritmos genéticos sobre los algoritmos de machine learning en función a la métrica seleccionada.

Para el diseño de algoritmos genéticos, se planteó como función objetivo o de fitness la métrica de desempeño del F1 Score, donde los individuos de cada población serían los algoritmos de machine learning que son más populares en la actualidad. Entre estos se puede encontrar: Random Forest, XGBoost, Maquinas de soporte vectorial, regresiones logísticas, etc. Para la selección de la población inicial, se dejó que la librería seleccionara los algoritmos predefinidos para este, además, para asegurar replicabilidad en los resultados se fijó la semilla de aleatoriedad que selecciona estos algoritmos.

Para realizar el cruce de los individuos, se conoce que la mitad de los algoritmos se cruzarán con un porcentaje de la población total que ya está predefinido, siempre y cuando los algoritmos sean de la misma naturaleza (es decir que sean los mismos algoritmos, pero con diferentes hiper parámetros). La mutación de los algoritmos usa un porcentaje igualmente predefinido, donde cambian los hiper parámetros de los algoritmos seleccionados.

Después de haber realizado la mutación y el cruce de los algoritmos se hará una selección por ruleta, donde los algoritmos que tengan mejor función objetivo tendrán mayor probabilidad de ser seleccionados para siguiente generación. Este proceso se repite la cantidad de generaciones que se haya especificado.

A partir de esto, para el diseño experimental, se consideró como variables independientes y como variables a cambiar las siguientes: (i) número de generaciones, (ii) número de individuos iniciales y (iii) porcentaje de mutación y cruce (los cuales vienen dados en conjunto por el algoritmo). Para realizar los experimentos, se fijarán los valores de las variables, se realizará una búsqueda de grilla y, finalmente, se obtendrá el mejor algoritmo con su respectiva métrica. Como línea base, el algoritmo genético tendrá los siguientes parámetros preconfigurados:

* Numero de generaciones: 5
* Población inicial:10
* Porcentaje de cruce y mutación: 90% y 10%, respectivamente.

|  |  |  |  |
| --- | --- | --- | --- |
| Escenario | Escenario | Modelo | Best F1 Score |
| 1 | Línea Base | Random Forrest. (Predeterminado) | 0.47 |
| 2 | Número de generaciones:30 | XGBoost (max\_depth=4, learning rate=0.1, min\_child\_weight=3) | 0.54 |
| 3 | Número de individuos población:50 | Random Forrest( características máximas = 0.25) | 0.53 |
| 4 | Porcentaje de cruce y mutación : 50% y 50% | Ensamble de algoritmos con XGBoost de diferentes características. | 0.52 |
| 5 | Mejor modelo redes neuronales | Redes neuronales con características del texto | 0.57 |

*Tabla 4. Resumen de los mejores resultados usando algoritmos genéticos.*

En cada ejecución, se utilizó un conjunto de entrenamiento y de prueba con los 17,000 registros, los cuales estaban distribuidos 70% y 30% respectivamente.  Por cada variable independiente, se corrió una búsqueda de grilla con todo el espacio de búsqueda mencionado anteriormente. Para los algoritmos con mejores resultados se calculó la matriz de confusión para compararlo con los modelos de línea base.

Table

Description automatically generated

*Tabla 5. Matriz de confusión del mejor algoritmo*

Como se puede apreciar en la tabla No. 4, el algoritmo de ensambles de Xgboost obtiene los mejores resultados, aunque solo por un punto porcentual frente a los demás y menos 3 puntos porcentuales del mejor algoritmo de redes neuronales con características de texto.

# Conclusiones

Se puede concluir que la calidad de los datos importa más que el algoritmo utilizado para la clasificación de las variables objetivo. Se puede apreciar que todos los algoritmos de machine learning ejecutados con diferentes configuraciones dan resultados similares a los algoritmos genéticos. Es decir, existe un límite en los resultados obtenido por la información procesada. En este caso, ningún algoritmo con las variables iniciales superó el 0.55, lo que sugiere que con las variables actuales este es el límite de efectividad.

En vez de usar modelos de aprendizaje profundo que son denominados cajas negras, se pueden usar algoritmos con una interpretabilidad mayor y obtener resultados muy similares a los del aprendizaje profundo.

En este caso, el mejor algoritmo encontrado fue ejecutado con 30 generaciones y 10 individuos cada una, lo cual puede significar que entre más generaciones los algoritmos encontrarán mejores resultados. Sin embargo, estos resultados están ligados a los datos de entrada y si se aplica el mismo algoritmo genético a una base de datos diferente estos resultados podrán cambiar significativamente.

# Referencias

1. Rouhiainen, L. Inteligencia Artificial. 101 cosas que debes saber hoy sobre nuestro futuro. Editorial Planeta (2018)

2. Minsky, M. La Máquina de las Emociones. Barcelona: Debate. (2010).

3. Ponce C., P. Inteligencia Artificial con Aplicaciones a la Ingeniería. México: Algaomega. (2010).

4. Yolis, E., Algoritmos Genéticos Aplicados A La Categorización Automática De Documentos: Tesis (2003).

5. Karkavitsas, George & Tsihrintzis, George. (2011). Automatic Music Genre Classification Using Hybrid Genetic Algorithms. 10.1007/978-3-642-22158-3\_32.

6. Randal S. Olson, Nathan Bartley, Ryan J. Urbanowicz, and Jason H. Moore (2016). Evaluation of a Tree-based Pipeline Optimization Tool for Automating Data Science. Proceedings of GECCO 2016, pages 485-492.

7. Setyo Wicaksono, A., & Afif Supianto, A. (2018). Hyper Parameter Optimization using Genetic Algorithm on Machine Learning Methods for Online News Popularity Prediction. https://www.semanticscholar.org. https://pdfs.semanticscholar.org/c573/489f37e7f5b66d9155e8c8b66c1dc4d66d2a.pdf