UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS

CAMPO DE CONOCIMIENTO: INTELIGENCIA ARTIFICIAL

DISEÑO DE UN ALGORITMO MULTIOBJETIVO BASADO EN NEUROEVOLUCIÓN PARA LA SELECCIÓN DE GENES Y CLASIFICACIÓN DE MICROARREGLOS

T E S I S

QUE PARA OPTAR EL GRADO DE:

MAESTRO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

P R E S E N T A:

DANIEL GARCÍA NÚÑEZ

TUTOR PRINCIPAL:

DRA. KATYA RODRÍGUEZ VÁZQUEZ

INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS

Ciudad de México, México, mayo 2023

Para los que tengo hoy a mi lado y con quien puedo compartir este logro,
para los que ya no están pero formaron parte de mi vida,
y para los que aún no conozco, pero que quizás encontraron esta tesis
por mera curiosidad.

Agradecimientos

Quiero agradecer especialmente a la Dra. Katya Rodríguez Vázquez, quien me apoyo y guio durante el desarrollo del presente trabajo y en el transcurso de mi maestría. Agradezco que me haya introducido al campo de estudio del cómputo evolutivo y que me haya compartido su pasión por esta área. Su confianza y motivación fueron muy importantes para poder concluir mi tesis y completar otras metas, como la participación en el GECCO 2022.

De igual manera, quiero agradecer al Dr. Carlos Ignacio Hernández Castellanos, quien también me guio para llevar a cabo este proyecto y la participación en el GECCO 2022. Considero que lo que me enseñó durante sus materias ha sido fundamental para a mi formación como estudiante y para mejorar mi investigación.

También, agradezco a mi comité sinodal, incluidos la Dra. Helena Montserrat Gómez Adorno, el Dr. Gibran Fuentes Pineda y al Dr. Edgar Galván López, por su retroalimentación sobre mi tesis.

Agradezco al IIMAS, al Dr. Javier Gómez y a Lulú por el apoyo que me dieron desde mi aplicación al posgrado hasta que lo terminé. También extiendo mi agradecimiento a mis profesores y compañeros que compartieron conmigo de su conocimiento y experiencia.

Adicional a esto, quiero agradecer al CONACYT que me ofreció apoyo económico para poderme enfocar en mi posgrado y sacar el mayor provecho de este.

Asimismo, agradezco al PAEP por el apoyo económico que me proporcionaron para ir a presentar parte de mi proyecto al congreso internacional GECCO 2022 en Boston.

El periodo en que realicé mi maestría se juntó con una pandemia global y estuvo lleno de dificultades e incertidumbre, pero agradezco haber podido enfrentarlo estando cerca de mi familia. Agradezco a mi familia, amigos y amigas por todo el apoyo y motivación que me han brindado. Especialmente, agradezco a mis padres y hermanas por acompañarme y el apoyo incondicional que siempre

me han dado.

Finalmente, también agradezco a mi abuela, por todo su apoyo, lo que me enseñó y porque ella creía en mí. Ser la persona que ella ya creía que yo era es una de mis mayores motivaciones y hoy siento que estoy un paso más cerca de serlo. Pero sobre todo agradezco por todo el cariño que me dio.

Resumen

En el presente proyecto, se presenta un nuevo algoritmo multiobjetivo basado en neuroevolución, el cual toma como base al algoritmo SMS-EMOA junto con la codificación genética de NEAT y los operadores de cruza y mutación presentados en N3O. El algoritmo fue nombrado SMS-MONEAT por sus siglas en inglés "*$\mathscr{S}$ Metric Selection Multi-Objective NEAT*". El diseño de SMS-MONEAT tuvo como principal motivación los microarreglos de ADN, los cuales, son una herramienta para analizar miles de genes de manera simultánea y son comúnmente utilizados para la detección e identificación de múltiples enfermedades, principalmente el cáncer. Sin embargo, los conjuntos de datos existentes tienen un problema de dimensionalidad, contando con decenas o centenas de muestras mientras que cada una tiene miles de genes. Adicionalmente, estos conjuntos de datos tienden a tener un desbalance de muestras entre cada clase. Estas propiedades en los conjuntos de datos dificultan su análisis y pueden ocasionar un sobreajuste al entrenar modelos de clasificación convencionales. Por ello, un proceso de reducción de características es necesario para identificar genes relevantes y generar modelos de clasificación confiables. SMS-MONEAT fue utilizado para generar redes neuronales artificiales para clasificación binaria a la par de realizar un proceso de selección de características. La metodología completa se conformó de un filtro estadístico basado en la prueba H de Kruskal Wallis para la selección de características, seguido por la ejecución de SMS-MONEAT teniendo como objetivos la minimización de las características seleccionadas y el valor de entropía cruzada binaria de los modelos de clasificación generados. También, se incluyó un archivo externo con un procedimiento de especiación basada en los subconjuntos de características seleccionadas para impulsar la diversidad de las soluciones almacenadas. El último paso de la metodología consistió en un procedimiento de selección de soluciones basado en la suma ponderada de pesos con los valores de entropía cruzada y el promedio geométrico evaluados en el conjunto de entrenamiento y en un conjunto de validación.

La metodología propuesta se comparó utilizando SMS-MONEAT y N3O. Adicionalmente, los resultados también se compararon contra una metodología multiobjetivo estándar que consistía en SMS-EMOA y KNN. La comparación se realizó utilizando 20 conjuntos de datos de microarreglos de los principales tipos de cáncer, siendo el colon, hígado, leucemia, mama y próstata. 17 de estos conjuntos fueron tomados de la base de datos CuMiDa y el resto fueron conjuntos de datos de referencia encontrados en la literatura. Los resultados obtenidos

mostraron que SMS-MONEAT es capaz de encontrar soluciones competitivas a N3O respecto a la función de entropía cruzada, pero con una diferencia significativa favorable minimizando el número de características. Por otro lado, al comparar las soluciones obtenidas por SMS-MONEAT y por SMS-EMOA, se demostró una diferencia significativa en la mayoría de los experimentos realizados respecto a la entropía cruzada de las redes neuronales generadas por SMS-MONEAT sobre los modelos entrenados en la otra metodología. Por lo tanto, al comparar los algoritmos utilizando el indicador del hipervolumen con el número de características seleccionadas y la entropía cruzada de los modelos de clasificación, SMS-MONEAT mostró un desempeño superior ante N3O y la otra metodología multiobjetivo.

Abstract

This project introduces a new neuroevolution-based multiobjective algorithm, which uses the SMS-EMOA framework along with the NEAT genetic codification and N3O crossover and mutation operators. The algorithm was named $\mathscr{S}$ Metric Selection Multiobjective NEAT (SMS-MONEAT). The primary motivation behind SMS-MONEAT was the ADN microarrays, a tool used to analyze thousands of genes simultaneously and commonly used for multiple disease detection and identification, mainly cancer. However, existing datasets have a dimensionality problem since they have tens or hundreds of samples, each consisting of thousands of genes. Moreover, these datasets tend to have an unbalanced number of samples for each class. These properties may hinder their analysis and cause overfitting when training commonly used classification models. Hence, a feature reduction process is required to identify relevant genes and build reliable classification models. SMS-MONEAT was used to generate artificial neural networks for binary classification while performing a feature selection process. The full methodology was composed of a statistical filter based on the Kruskal Wallis H test for feature selection, followed by the SMS-MONEAT execution, which attempts to minimize the number of selected features and the binary cross entropy from the classification models built. Also, an external archive was included with a speciation procedure based on the feature subset chosen to enhance the stored solutions' diversity. The methodology's last step consisted of a solution selection procedure based on a weighted sum of the cross entropy and geometric mean values evaluating the solutions on the training and validation datasets.

The proposed methodology was compared using SMS-MONEAT and N3O. Moreover, it was compared against a standard methodology using SMS-EMOA and KNN. The comparison took 20 microarray datasets from the main cancer types: colorectal, liver, leukemia, breast, and prostate. 17 of them were gathered from the CuMiDa database; the rest were benchmark datasets found in the literature. The results showed that SMS-MONEAT could find competitive solutions against N3O regarding the cross-entropy function but with a favorable significant difference in minimizing the number of features selected. Furthermore, when comparing SMS-MONEAT solutions with the ones from SMS-EMOA, a significant difference was shown in most of the performed experiments regarding the cross entropy of the generated neural networks from SMS-MONEAT over the classification models trained on the other methodology. Thus, when

comparing the algorithms using the hypervolume indicator with the number of selected features and the classification models' cross-entropy, SMS-MONEAT showed a higher performance against N3O and the other multiobjective methodology.

Índice general

Abreviaturas

ACO Ant Colony Optimization.

ANN Artificial Neural Network.

CNN Convolutional Neural Network.

CuMiDa Curated Microarray Database.

DT Decision Tree.

FS-NEAT Feature Selective NEAT.

GBDT Gradient Boosting Decisition Tree.

GECCO Genetic and Evolutionary Computation Conference.

ICA Independent Component Analysis.

kNN k-Nearest Neighbors.

KW Kruskal Wallis.

MM-NEAT Modular Multi-Objective NEAT.

mNEAT Multi-Objective NEAT.

mNEAT-IB Multi-Objective NEAT-Indicator Based.

MOCell Cellular Genetic Algorithm for Multiobjective Optimization.

MOCHC Multiobjective Cross-generational elitist selection, Heterogeneous recombination, and Cataclysmic mutation.

MoGA Multo-Objective Genetic Algorithm.

MW Mann-Whitney.

N3O FS-NEAT New 3 Operators.

NEAT Neuroevolution of Aumenting Topologies.

NEAT-MODS NEAT Multi-Objective Diversified Species.

NEAT-PS NEAT Pareto-Strengh.

NSGA-II Non-dominated Sorting Genetic Algorithm II.

PCA Principal Component Analysis.

PCE Projective Clustering Ensemble.

PSO Particle Swarm Optimization.

RF Random Forest.

RFE Recursive Feature Elimination.

RLR Randomized Logistic Regression.

SMS-MONEAT $\mathscr{S}$ Metric Selection Multi-Objective NEAT.

SMS-EMOA $\mathscr{S}$ Metric Selection Evolutionary Multi-Objective Algorithm.

SPEA2 Strengh Pareto Evolutionary Algorithm 2.

SUNA Spectrum-diverse Unified Neuroevolution Arquitecture.

SVM Support Vector Machines.

Índice de figuras

Índice de tablas

Lista de algoritmos

Capítulo 1

Introducción

One must experiment with teaching one such [learning] machine and see how well it learns. One can then try another and see if it is better or worse. There is an obvious connection between this process and evolution...

Alan Turing

Quizás se pregunten por qué empecé con esa frase fuera de contexto de Alan Turing, habiendo tantas otras frases de él a elegir que pudiera dar al lector un sentimiento menor de confusión y uno mayor de motivación para que decidiera continuar leyendo esta tesis un poco más. Pero para mí, que él haya escrito eso hace más de 70 años es sorprendente, y quiero empezar intentando compartir un poco de mi emoción al respecto. En este entonces, Alan Turing hablaba sobre cómo pensaba que sería el futuro de las computadoras, si algún día serían capaces de pensar o aprender, o si por lo menos era un tema que valiera la pena discutir. Y dentro de las ideas que expresó, mencionó que si se quisiera simular a la mente de un adulto podría ser más sencillo simular la de un niño y educarlo hasta obtener algo más cercano a la de un adulto, e hizo esta analogía entre el proceso de enseñanza y el de la evolución biológica, sobre cómo una máquina podría aprender a través de cambios similares a los que ocurren en la mutación genética, y que incluir un juez para determinar al más apto asemejaría a la selección natural. No obstante, esperando que el proceso fuera más rápido para las máquinas que lo que fue para nosotros. Es posible que sea una visión muy simplista tanto de la enseñanza como de la evolución, y quizás yo, como autor de este trabajo, me esfuerzo en aceptar las similitudes que él quiso señalar. Y es porque el trabajo que se presenta aquí trata de eso, de un algoritmo inspirado en la evolución que permite transformar la estructura de modelos computacionales (inspirados en el cerebro humano) para que mejoren en una cierta tarea. Es entonces que la analogía se vuelve una herramienta, cuando se puede aprovechar de las similitudes para alcanzar una meta por medio de otra, cuando se le puede «enseñar» a una máquina por medio de la evolución.

1.1. Microarreglos de ADN

En 1984, Dulbecco argumentó que el conocer la secuenciación del genoma humano facilitaría nuestro entendimiento sobre el cáncer, iniciando así la idea del proyecto del genoma humano [1]. Este proyecto se lanzó en 1990 y se completó en 2003 logrando la primera secuenciación del genoma humano [2], cambiando por completo áreas de investigación como la biología e impulsó avances en medicina, identificando genes relacionados a ciertas enfermedades y permitiendo mejores diagnósticos y tratamientos. En la actualidad, la referencia del genoma humano fue lanzada por el Consorcio de Referencia del Genoma en 2013 y ha sido constantemente mejorado en las últimas dos décadas [3].

Los microarreglos o chips de ADN son una herramienta que permiten el análisis simultáneo del nivel de expresión genética en muestras celulares o de tejido. Estos son contenedores, hechos de plástico, sílice o vidrio, a los cuales se adhieren material genético. Se puede incluir hasta 40,000 fragmentos de ADN distintos por cada centímetro cuadrado de dicha superficie [4]. Existen dos tipos de microarreglos de ADN: los ergonómicos, que permiten detectar ausencia o exceso de genes o detectar alguna mutación; y las transcriptómicas que miden niveles de ARN mensajeros [5].

Para construir los microarreglos se obtienen muestras de ADN de pacientes y sujetos control, según el estudio que se quiera realizar. Las moléculas se marcan utilizando fluoroforos (un compuesto químico fluorescente) para etiquetarlas y se hibridan con una cadena de ADN complementaria que se encuentra en la superficie del microarreglo. Por medio de un láser se obtiene una matriz de puntos de colores y se puede determinar de manera cuantitativa el nivel de expresión que representa cada gen [5]. Finalmente, se obtiene una imagen de cada microarreglo la cuales se procesan para construir un conjunto de datos. La Figura 1.1 muestra el proceso general para obtener un conjunto de datos de microarreglos.

La ventaja de utilizar microarreglos para el análisis de células y el estudio de diferentes enfermedades es que permite analizar miles de genes de manera simultánea. Por otro lado, y además irónicamente, la desventaja es la enorme cantidad de información que se debe analizar, sobre todo si se compara con la pequeña cantidad de muestras que contienen los conjuntos de datos.

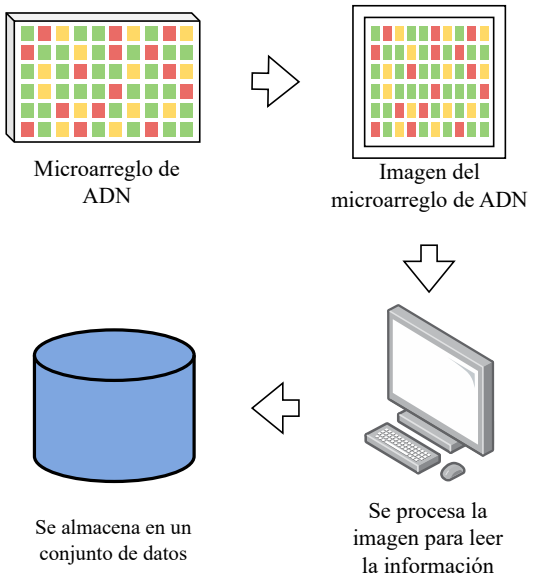


Figura 1.1: Proceso para obtener una muestra de un conjunto de datos de microarreglos de ADN.

1.2. Definición del problema

Los microarreglos contienen información del nivel de expresión de miles o hasta millones de genes. Sin embargo, los conjuntos de datos de microarreglos existentes cuentan únicamente con decenas o centenas de muestras. Además de esto, estos conjuntos tienden a tener un número desbalanceado de muestras para cada una de sus clases. Dichas propiedades en los conjuntos de datos pueden ocasionar un sobreajuste al entrenar modelos computacionales de clasificación usados convencionalmente. Adicional a esto, la mayoría de los genes que se encuentran en una muestra de ADN no son relevantes para la clasificación entre las clases de estudio [6], por lo que mucha información dentro del conjunto de datos se puede considerar como ruido.

Para el análisis de microarreglos comúnmente se realiza un proceso de reducción de dimensionalidad con el objetivo de eliminar ruido e información redundante del conjunto de datos. Además, identificar genes relevantes permitiría generar modelos de clasificación más confiables y a un menor costo computacional. Sin embargo, un microarreglo puede contener miles de genes y, por lo tanto, la cantidad de posibles combinaciones entre ellos es exponencial generando un espacio de búsqueda inmenso. Aunado a eso, para evaluar cada combinación de genes se debe entrenar un modelo de clasificación. Hay metodologías que lo entrenan una vez, otras que prueban diferentes subconjuntos de genes y cada una requiere entrenar su propio modelo de clasificación, o incluso hay quienes

evalúan utilizando múltiples modelos o realizan un proceso de validación cruzada, pero independientemente de la metodología, dicha evaluación tiende a ser la parte de mayor complejidad computacional y hace inviable revisar cada una de las posibles combinaciones. Finalmente, al no poder revisar cada combinación, es imposible tener certeza de encontrar la mejor posible, sin embargo, es suficiente con encontrar un subconjunto que permita al modelo de clasificación diferenciar de manera acertada entre las clases.

1.3. Justificación

Una de las principales aplicaciones de los microarreglos es la detección y diagnóstico de enfermedades, principalmente el cáncer. El cáncer es la principal causa de muerte en el mundo. De acuerdo con información de la Organización Mundial de la Salud (OMS) en el 2020 [7], se atribuyeron casi 10 millones de defunciones atribuidas a esta enfermedad, siendo algunos de los tipos de cáncer más comunes el de mama, pulmón, colon y recto y próstata. Muchos casos de cáncer se pueden curar si se tratan a tiempo y eficazmente.

Construir metodologías para el análisis de microarreglos podría ser fundamental para la detección temprana de cáncer y de otras enfermedades genéticas. Además, identificar genes relacionados a estas enfermedades permitiría un mayor entendimiento sobre las mismas y poder generar modelos de clasificación más confiables.

La metodología propuesta en este proyecto para la selección de genes y clasificación de microarreglos empieza con una reducción de características utilizando un filtro estadístico, descartando posibles características no relevantes. Después se aplica un nuevo algoritmo, que tiene dos propósitos: el primero es encontrar un subconjunto mínimo de características y generar modelos de clasificación, ambos realizados de manera simultánea. Los modelos de clasificación son redes neuronales artificiales cuya topología y pesos de sus conexiones son generadas automáticamente por el algoritmo, de este modo, se evita el esfuerzo manual de diseñar una topología óptima de la red.

1.4. Objetivos

El objetivo principal de esta tesis es el diseño e implementación de un nuevo algoritmo multiobjetivo basado en neuroevolución para la reducción de características y optimización de ANNs para clasificación binaria. El diseño de la metodología presentada toma como base al problema de análisis de conjuntos de datos de microarreglos de ADN, con el objetivo de encontrar el subconjunto mínimo de genes significativos para diferenciar entre las clases y construir modelos que puedan clasificarlos de manera precisa. El nuevo algoritmo toma como marco de referencia al algoritmo evolutivo multiobjetivo basado en selección por la métrica $\mathscr{S}$ (SMS-EMOA por sus siglas en inglés "$\mathscr{S}$ Metric Selection Evolutionary Multi-Objective Algorithm") y utiliza la codificación genética y opera-

dores de cruza y mutación de una variante del algoritmo de neuroevolución de topologías aumentadas (NEAT por "Neuroevolution of Aumenting Topologies") la cual fue nombrada N3O. El algoritmo fue nombrado SMS-MONEAT por sus siglas en inglés "*$\mathscr{S}$ Metric Selection Multi-Objective NEAT*".

Con mayor detalle, se enlista a continuación los objetivos particulares de manera cronológica en la que se realizaron:

- Diseñar e implementar el algoritmo SMS-MONEAT utilizando el lenguaje de programación Python 3 y bibliotecas de código abierto.
- Diseñar una metodología de especiación para mantener diversidad en las soluciones generadas por SMS-MONEAT mediante la inclusión de un archivo externo.
- Optimizar los hiperparámetros relacionados a los operadores de cruza y mutación de SMS-MONEAT utilizando *iRace* [8].
- Validar el funcionamiento de SMS-MONEAT en conjuntos de datos de referencia de microarreglos de cáncer.
- Comparar el desempeño de las siguientes metodologías para la selección de genes y clasificación de microarreglos de cáncer:
  - La metodología propuesta utilizando SMS-MONEAT.
  - La metodología propuesta utilizando SMS-MONEAT junto con el archivo externo de especiación.
  - La metodología propuesta utilizando N3O.
  - Una metodología estándar multiobjetivo utilizando SMS-EMOA y el algoritmo de k vecinos más cercanos (kNN por "k-Nearest Neighbors").

1.5. Contribuciones

Durante este trabajo se realizaron las siguientes contribuciones:

- Diseño de SMS-MONEAT, un algoritmo multiobjetivo basado en neuroevolución, que combina el marco de trabajo de SMS-EMOA y la codificación y operadores evolutivos de N3O.
- Diseño de una metodología de especiación para mantener diversidad en las características seleccionadas utilizando un archivo externo.
- Diseño e implementación de una metodología para la selección de genes y clasificación de microarreglos de cáncer que se compone por los siguientes componentes:
  - Una metodología de selección de características basada en un filtro estadístico (prueba H de Kruskal Wallis).

    - Una metodología de selección de características envolvente basada en neuroevolución (SMS-MONEAT).
    - Un archivo externo con una metodología de especiación.
    - Una metodología para seleccionar soluciones utilizando un conjunto de validación.

- Comparación entre SMS-MONEAT, N3O y SMS-EMOA/kNN para la tarea de selección de genes y clasificación de microarreglos en 20 conjuntos de datos de diversos tipos de cáncer.
- Contribución en la conferencia internacional "Genetic and Evolutionary Computation Conference" (GECCO) 2022: García-Núñez D., Rodríguez-Vázquez K., Hernández C. Neuroevolution based multi-objective algorithm for gene selection and microarray classification.

1.6. Estructura de esta tesis

En el siguiente capítulo, se describen las bases teóricas que fundamentaron esta investigación y el algoritmo desarrollado. El Capítulo 3, expone trabajos previos relacionados con el problema en cuestión y que, algunos de ellos, fueron la inspiración detrás de la presente tesis. La metodología realizada se presenta en el Capítulo 4, desde el preprocesamiento de los conjuntos de datos, el nuevo algoritmo diseñado y la selección de soluciones utilizadas. Posteriormente, en el Capítulo 5, se detallan los experimentos realizados junto con los conjuntos de datos utilizados, mientras que el análisis de los resultados se presenta en el Capítulo 6. Finalmente, en el Capítulo 7 se exponen las conclusiones obtenidas durante este proyecto e ideas que se proponen como trabajo a futuro para continuar esta investigación.

Capítulo 2

Marco teórico

En este capítulo, se presentan conceptos básicos requeridos para entender esta tesis. Primero se describe el proceso de reducción de características (Sección 2.1), el cual está relacionado con el problema a resolver de esta tesis. Después, se da una breve introducción a las redes neuronales artificiales (Sección 2.2) y al cómputo evolutivo (Sección 2.3), lo cual permite describir posteriormente el tema de neuroevolución (Sección 2.4). Luego se presenta al algoritmo de NEAT (Sección 2.5) y a sus variantes para selección de características (Sección 2.6). También se explican conceptos sobre optimización multiobjetivo (Sección 2.7) y sobre el algoritmo de SMS-EMOA (Sección 2.8).

2.1. Reducción de características

La reducción de características en un conjunto de datos es un proceso en el cual se decrementa el número de variables, pero manteniendo la información relevante de cada muestra. El propósito de este proceso es eliminar información redundante o irrelevante del conjunto de datos, por lo que resulta importante previo a entrenar modelos predictivos. Existen dos tipos principales de métodos de reducción de características: la primera es la extracción de nuevas características las cuales representen la información más relevante en un menor número de dimensiones; y la segunda es la selección de características, la cual busca minimizar el subconjunto de las características originales que mantengan la información relevante del conjunto de datos, y en este trabajo, nos enfocaremos en esta segunda metodología.

En específico, nos enfocaremos en dos metodologías para selección de características: una basada en filtro y otra envolvente. La formulación de ambas se describirá a continuación.

2.1.1. Método de filtro para la selección de características

El método de filtro para selección de características utiliza una función para evaluar cada característica por separado y de acuerdo a un umbral definido seleccionar o remover dicha característica. Se puede describir como:

$$Q \subset M, f(Xq) \leq \delta \forall q \in Q, \tag{2.1}$$

donde Q es el subconjunto del total de características M, f es la función de filtro y Xq es vector del conjunto de datos para la característica q. Q está formado por aquellas características que al evaluar $f(Xq)$ tengan un valor menor o igual al umbral δ. Un ejemplo de esta metodología es el uso de pruebas estadísticas y el valor p obtenido para evaluar la relevancia de cada característica.

2.1.2. Método envolvente para la selección de características

Existen distintos modelos predictivos, en esta tesis en específico nos enfocaremos en aquellos que se entrenan mediante aprendizaje supervisado, en los cuales se utiliza un conjunto de datos con muestras etiquetadas y se busca optimizar los parámetros de modelo minimizando la diferencia entre valores predichos por el modelo y las etiquetas reales.

$$f(x) = y', \tag{2.2}$$

donde f es un modelo predictivo, X es un conjunto de datos y y' es el vector de valores predichos por el modelo. Para evaluar el modelo se utiliza una función de error $g(y, y')$ donde y se refiere a los valores reales de la variable objetivo a predecir. El conjunto de datos X se compone por n muestras y un conjunto de características M. Por lo que la tarea de selección de características se puede representar como:

$$Q \subset M : \tag{2.3a}$$
$$f(X_Q) = y'_Q, \tag{2.3b}$$
$$\text{mín}\,\{g(y, y'_Q)\}, \tag{2.3c}$$

donde Q sería el subconjunto de M que minimiza la función de error g. Por lo que, Q se compone por aquellas características que al entrenar al modelo f, se minimiza el resultado de la función de error g.

2.2. Redes neuronales artificiales

Es común que la ciencia busque entender comportamientos que se observan en la naturaleza, y en ocasiones, se utiliza este conocimiento para construir herramientas que mimeticen dicho comportamiento con el fin de que sean más

eficientes. El cerebro humano tiene capacidades que aún sobrepasan a las de las computadoras, este tiene una gran habilidad para adaptarse a múltiples ambientes o tareas y, sobre todo, tiene la habilidad de aprender de la experiencia. Entendiendo el potencial del cerebro humano, hace sentido que en el campo de ciencia de la computación se le haya tomado como inspiración para buscar generar modelos cada vez más eficientes y robustos.

Las redes neuronales artificiales (ANNs por "Artificial Neural Networks") son modelos computacionales de aprendizaje automatizado, las cuales se inspiraron en los primeros modelos de procesos sensoriales del cerebro. En 1943, McCulloch y Pitts [9] modelaron una neurona como un interruptor, el cual se activaba o no, dependiendo de la señal de entrada que recibía de otras neuronas. Cada una de las señales de entrada, tenía un peso asociado por el cual se multiplicaba, dicho peso hacía referencia a la sinapsis, es decir, la región de contacto entre dos neuronas biológicas. Años después, Rosenblatt [10] tomó lo anterior como base y presentó el modelo del "perceptrón" y mostró su capacidad de aprender características de ejemplos de entrada. Sin embargo, uno de los puntos de inflexión en la historia de este campo fue hasta 1986, cuando se presentó el método de retropropagación con el cual se podía entrenar ANNs más complejas que a su vez pudieran realizar tareas más complejas [11]. La retropropagación es un algoritmo que calcula el gradiente de la función de pérdida y mediante el método del gradiente descendente (o alguna variación de esta) optimiza el valor de los pesos de las conexiones de las neuronas, y hasta la actualidad es el método más popular para entrenar ANN. No obstante, tardaron algunos años más en el desarrollo de hardware para que las computadoras pudieran ejecutar este método de manera más eficiente y de este modo, explotar el interés en el área de aprendizaje profundo a nivel mundial.

En la actualidad, existe una enorme variedad de arquitecturas de ANNs que a su vez cubren un amplio campo de aplicaciones en diversas áreas de la ciencia, y en algunos casos, presentan ser soluciones del estado del arte.

2.3. Cómputo evolutivo

Los algoritmos evolutivos son métodos de optimización estocásticos bioinspirados que se basan en principios de la evolución Darwiniana [12]. Son algoritmos basados en poblaciones en donde cada individuo representa un punto del espacio de búsqueda los cuales, van cambiando a lo largo de múltiples generaciones con el propósito de encontrar un valor óptimo o cercano al óptimo. Similar a la selección natural, el individuo más apto tendrá una mayor probabilidad de heredar sus características a la siguiente generación, y se explora el espacio de búsqueda mediante funciones que combinan las características de individuos o que las mutan mediante una perturbación aleatoria.

Hay 4 metodologías de referencia en esta área: algoritmos genéticos, estrategias evolutivas, programación genética y programación evolutiva; sin embargo, su flexibilidad permite combinar elementos de estas metodologías dando paso a un marco de trabajo general de algoritmos evolutivos [13]. Uno de los conceptos

en el cómputo evolutivo, es la codificación genética de los individuos, en la que cada variable de decisión representa un gen, y el conjunto de las variables al genoma. Existen distintas codificaciones y su propósito es representar al conjunto de variables de modo que puedan ser aplicados los operadores de cruza de individuos y mutación. Siendo que esta codificación representa al genoma, la representación original de las variables representaría al fenotipo. Por ejemplo, para un problema con una variable en el espacio de $\mathbb{R}$, se podría codificar a una representación binaria y utilizar operadores apropiados para esta representación; en este caso, el fenotipo sería el valor de la variable en $\mathbb{R}$ y el genotipo sería el arreglo de todos los bits.

Los principales operadores utilizados en algoritmos evolutivos se enlistan a continuación:

- La **selección** es el proceso con el cual se elige uno o más individuos de acuerdo con su aptitud para generar un nuevo individuo. La manera de elegir puede ser determinística o probabilística de acuerdo con la metodología utilizada, siendo la segunda la más común. Al seleccionar a los individuos de acuerdo con su aptitud es la manera en que la población puede converger en un óptimo, sin embargo, utilizar una metodología probabilística permite una mayor exploración del espacio de búsqueda mientras que una determinística una convergencia más rápida.

- La **cruza** es el proceso en que dos individuos (previamente seleccionados por el operador anterior) intercambian información genética para generar nuevos. La idea detrás de este operador es que al combinar dos individuos de manera estocástica se podría generar otro u otros que tengan características intermedias, y si ambos padres tienen una buena aptitud los descendientes podrían superarla.

- La **mutación** genera una perturbación en los individuos con el propósito de explorar lugares nuevos del espacio de búsqueda. Uno o más genes de un individuo pueden ser modificados de acuerdo con una probabilidad definida. Si dicha probabilidad es muy alta, la perturbación será muy disruptiva y la búsqueda se convertirá en aleatoria, por otro lado, si la probabilidad es muy baja, no habrá mucha variación de genes entre generación y generación.

- El **elitismo** permite mantener a los individuos con mejor aptitud a la siguiente generación. Es decir, en cada generación se crean nuevos individuos (descendencia) a partir de la generación anterior (padres), mediante el elitismo, un porcentaje de los padres pasarán a la siguiente generación sin ser modificados. Mantener a las soluciones con mejor aptitud, permite a la población converger en soluciones óptimas.

2.4. Neuroevolución

El cerebro humano tiene alrededor de 100 billones de neuronas, casi la misma cantidad que las estrellas en la Vía Láctea, y dichas neuronas forman cerca de 10^{15} conexiones [14], y esta increíble estructura fue creada por un proceso evolutivo. Siendo la manera en que se comunican las neuronas en el cerebro la inspiración de las ANNs, la evolución de estas es la analogía que da cimientos a la neuroevolución.

La neurevolución es un campo de estudio que utiliza algoritmos evolutivos para entrenar ANNs. Ofrece ciertas ventajas sobre la técnica de retropropagación, puesto que no sólo optimiza las conexiones entre neuronas, sino que también puede optimizar la estructura de estas [15]. Lo anterior implica un diseño automático de la arquitectura de las ANNs, reduciendo el esfuerzo manual y su dificultad de implementación. En comparación, al aprendizaje por refuerzo que también ha sido utilizado para optimización de arquitecturas de ANNs, los algoritmos evolutivos han demostrado requerir un menor costo computacional [13].

Los algoritmos evolutivos son libres de gradiente y son basados en poblaciones, por lo que son fáciles de paralelizar y tienen una gran capacidad de exploración del espacio de búsqueda, así como herramientas para salir de óptimos locales, sin embargo, en neuroevolución, una desventaja es el costo de evaluación, ya que para poblaciones muy grandes o conjuntos de datos con muchas muestras esta operación puede tomar mucho tiempo [13].

2.5. NEAT

NEAT es uno de los algoritmos más populares en este campo, diseñado por Kenneth O. Stanley y Risto Miikkulainen de la Universidad de Texas en Austin en el 2002 [16]. Su popularidad se debe a que fue de los primeros algoritmos que proponía evolucionar la topología de las ANNS de manera eficiente, marcando históricamente las conexiones creadas para que en el proceso de cruza los genes fueran consistentes entre ambos padres. Del mismo modo, fue de los primeros en incluir la idea de especiación, buscando proteger nuevas estructuras. Finalmente, algo a resaltar es que propone iniciar con una topología mínima, buscando de este modo encontrar soluciones con el mínimo número de conexiones.

Este algoritmo utiliza una codificación indirecta de la ANN, es decir, representa a la red de una manera en que facilite utilizar operadores evolutivos, la cual se explicará a detalle en la Sección 2.5.1

El procedimiento principal de NEAT se describe en el Algoritmo 1. A grandes rasgos, el algoritmo empieza generando una población inicial de soluciones de tamaño μ. Después, comienza un proceso iterativo que termina cuando una condición de paro se cumpla, la cual puede ser definida por un número máximo de iteraciones o que alguna de las soluciones encontradas alcance un desempeño esperado. En cada iteración, primero se divide la población en distintas especies de acuerdo a una métrica de similitud, y después de cada especie se generan

Algoritmo 1 NEAT

```
1: P_0 ← inicializar()
2: S_0 ← ∅
3: t ← 0
4: mientras la condición de paro no sea verdad hacer
5:     S_{t+1} ← especiación(P_t, S_t)
6:     P_{t+1} ← generar(S_{t+1})
7:     t ← t + 1
```

Algoritmo 2 NEAT: Generar(S)

```
1: (Q, λ) ← elitismo(S, μ)
2: p ← distribución(S, λ)          ▷ distribución de descendientes por especie
3: para cada s ∈ S hacer
4:     i ← p_s
5:     mientras i > 0 hacer
6:         (x_1, x_2) ← selección(s)
7:         y ← cruza(x_1, x_2)
8:         y ← mutación(y)
9:         v ← validar(y)
10:        si v es verdadero entonces
11:            Q ← Q ∪ y
12:            i ← i − 1
13: devolver Q
```

nuevos individuos utilizando operadores de cruza y de mutación. También hay una probabilidad de generar individuos que combinen especies.

El Algoritmo 2 describe la función para generar una nueva población en NEAT. Este algoritmo toma como entrada el conjunto de todas las especies S. En este método primero se realiza un proceso de elitismo y se obtiene un conjunto inicial Q con los individuos de las especies que pasan a la siguiente generación sin ningún cambio, además del valor de λ que es la cantidad de individuos nuevos por generar. Después se calcula la distribución de descendientes de cada especie, el cual depende del promedio de la aptitud de los miembros de cada especie (esto se describe con mayor detalle en la Sección 2.5.4). Posteriormente, se generan la cantidad de hijos correspondientes por cada especie siguiendo un proceso de selección de padres, seguido por los métodos de cruza y mutación. También se incluye un método de validación, previo a agregar los individuos a la nueva población, con el fin de evitar que se generen ciclos en la ANN o que no existan conexiones habilitadas entre las entradas y las salidas.

En las Secciones 2.5.2 y 2.5.3 se explicarán los operadores de cruza y mutación respectivamente, utilizados para generar la nueva población, mientras que en la Sección 2.5.4 se describirá la metodología de especiación del algoritmo.

2.5.1. Codificación genética

En NEAT, el genoma de cada individuo es la representación de la ANN, mientras que el modelo en sí sería el fenotipo. Su esquema de codificación genética fue diseñado para permitir identificar relaciones topológicas entre los padres en el proceso de cruza. Tomando como inspiración el proceso de cruza en la biología, los genes se seleccionan de manera consistente entre ambos padres, es decir, el gen relacionado al color de ojos de la descendencia se selecciona entre el gen relacionado a lo mismo de cada padre, y lo mismo pasa con las demás características fenotípicas. Por lo que en NEAT, los "genes" utilizan un identificador que permite identificar aquellos que coinciden entre ambos padres. Con el fin de intentar no confundir al lector, en este capítulo utilizaremos el término "gen" para referirnos al término utilizado en cómputo evolutivo, más no al biológico.

El genoma incluye dos tipos de genes: los nodos y las conexiones. Un gen nodo se describe por un número de identificación y por su tipo, que puede ser una entrada, un nodo oculto o una salida. Asimismo, un gen conexión también tiene un número de identificación, pero al cual se le conoce como número de innovación, el cuál es utilizado durante el proceso de cruza para identificar los genes correspondientes entre ambos padres. El gen conexión también incluye parámetros como los números de identificación de los genes nodo de entrada y de salida, el valor del peso y una bandera para habilitar o deshabilitar la conexión. La Figura 2.1 muestra un ejemplo de la codificación genética de NEAT.

El algoritmo inicializa todos los individuos como redes completamente conectadas entre los nodos de entrada y los de salida. De este modo, las soluciones inician con una topología mínima, sin nodos ocultos, y la cuál va incrementando con el paso de las generaciones.

Genoma (genotipo)

Nodos

ID	**1**	**2**	**3**	**4**	**5**	**6**
Tipo	Entrada	Entrada	Salida	Salida	Oculto	Oculto

Conexiones

Inovación	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Entrada	1	1	2	2	4	5	6
Salida	4	5	5	6	3	3	3
Peso	w_1	w_2	w_3	w_4	w_5	w_6	w_7
Habilitada	Sí	Sí	Sí	Sí	No	Sí	Sí

Red neuronal artificial (fenotipo)

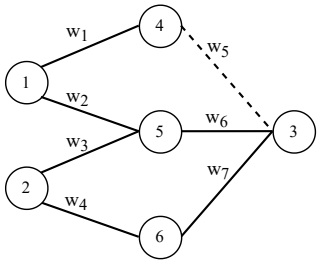


Figura 2.1: Ejemplo de la codificación genética de NEAT.

2.5.2. Operador de cruza

El operador de cruza toma el genoma de 2 individuos padres y ordena los genes conexión de acuerdo con el número de innovación para buscar aquellos genes en los cuales coinciden ambos padres. El operador selecciona de manera aleatoria respecto a aquellos genes que coinciden entre ambos padres para generar al individuo descendiente. Para aquellos genes que sólo se encuentran en uno de los padres, se dividen en dos clases: disjuntos y excedentes. Los disjuntos son aquellos que tienen un número de innovación menor al valor máximo del otro padre, mientras que los excedentes son aquellos que tienen un número de innovación mayor al máximo del otro padre. El individuo descendiente mantiene los genes disjuntos y excedentes que pertenecen al padre con una mejor aptitud de acuerdo con la función objetivo. En el caso de que ambos padres tengan la misma aptitud, entonces se decide de manera aleatoria si se agrega cada uno de estos genes.

Otra característica de este operador es que, si un gen está deshabilitado en el padre y es copiado al descendiente, hay una probabilidad del 25 % de que el gen se habilite.

La Figura 2.2 muestra un ejemplo del operador de cruza. En ella se muestran dos padres, representados por los genes de sus conexiones y su representación en grafo. Los genes en azul son aquellos que ambos padres comparten, los rosas son los genes disjuntos y el gris es el gen excedente. El hijo, se construye de los genes que coinciden de ambos padres de manera aleatoria y los genes disjuntos y excedentes del padre más apto, que para el ejemplo es el padre 1. Notemos que como el padre 2 tiene la conexión 5 deshabilitada, el hijo tiene probabilidad de heredarla del mismo modo y es el caso en el ejemplo (marcado con una línea punteada). Otra cosa por resaltar es que, para la nueva red, al tener la conexión 5 deshabilitada, la entrada 1 no tiene ninguna conexión con la salida (nodo 3), por lo que no influye en la salida, esto permite al algoritmo realizar selección de características durante la optimización de la red.

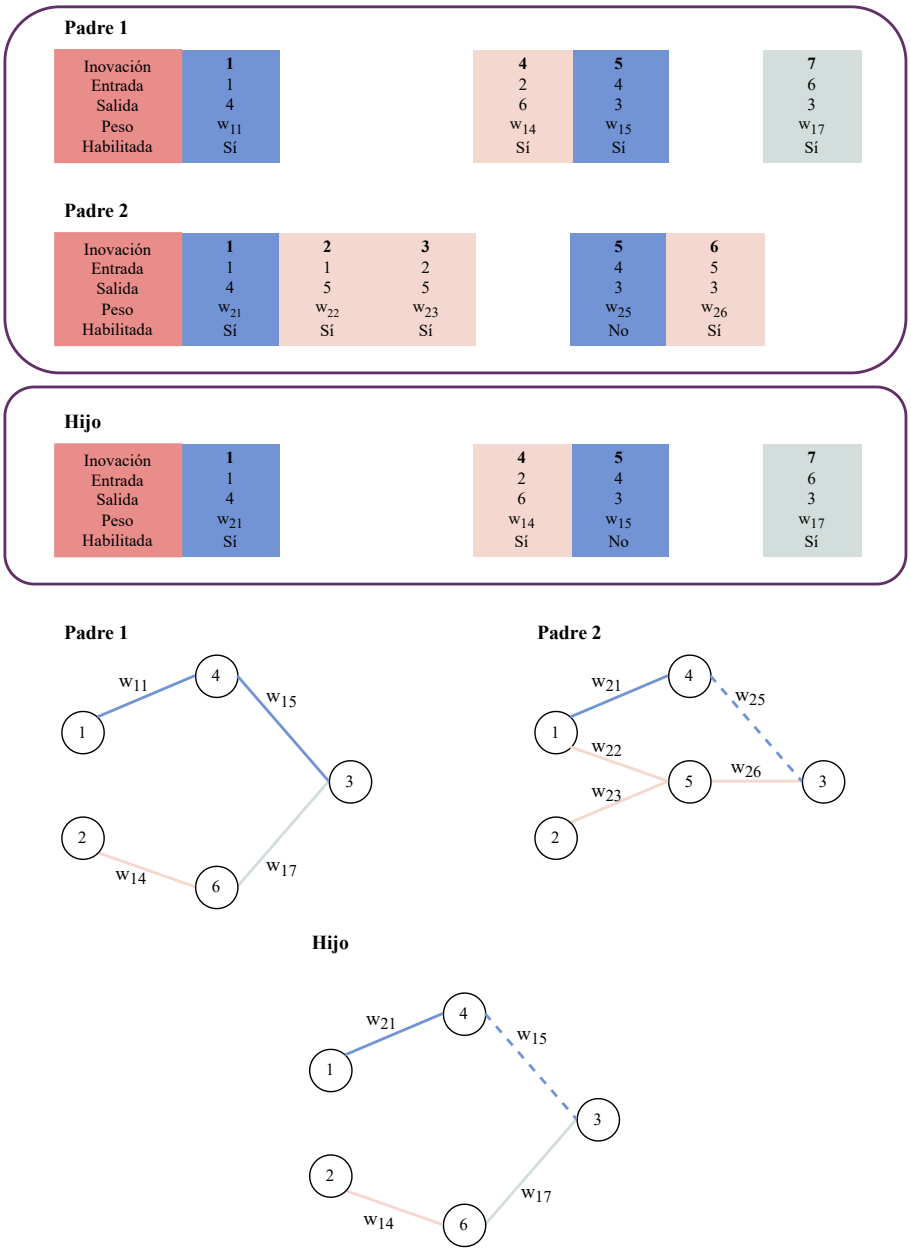


Figura 2.2: Ejemplo del operador de cruza en NEAT.

2.5.3. Operadores de mutación

En NEAT existen operadores de mutación para modificar tanto el peso como la estructura de la red. A continuación, se describen los distintos operadores:

- **Mutación de pesos**: Cada uno de los pesos de las conexiones en la red puede ser modificado por una cierta perturbación. Se pueden utilizar métodos tradicionales de mutación para codificación real del cómputo evolutivo, como la mutación polinomial. Sin embargo, se incluye una probabilidad (10 % en el artículo original) de que el valor del peso sea cambiado por un nuevo valor generado de manera aleatoria, en lugar de ser perturbado.
- **Agregar un nodo**: Este operador permite crear nuevos nodos ocultos en la red. Para ello, se elige una conexión existente y se deshabilita, y en su lugar se crea un nuevo nodo y dos conexiones que lo unen a los nodos de entrada y salida de la vieja conexión. La conexión entre el nodo de entrada al nuevo nodo tendrá un peso con valor igual a 1, mientras que la conexión entre el nuevo nodo y el nodo de salida tendrá el mismo valor de peso de la vieja conexión.
- **Agregar una conexión**: En este operador se crea una nueva conexión entre dos nodos existentes y que no tengan una conexión previa entre ellos. El valor del peso de esta conexión se genera de manera aleatoria. Se debe tener cuidado de no generar un ciclo en la red.

La Figura 2.3 muestra dos ejemplos de los operadores de mutación estructural de NEAT. En ella se muestra primero un ejemplo de la mutación para agregar un nodo, en la que se deshabilita la conexión w_1 y se remplaza por el nodo 4 y las conexiones w_3 y w_4. La conexión w_3 toma un valor de 1 mientras que la conexión w_4 toma el valor de la conexión w_1. También se muestra un ejemplo del operador para agregar una nueva conexión, conectando el nodo 2 y el nodo 4 con una nueva conexión w_5.

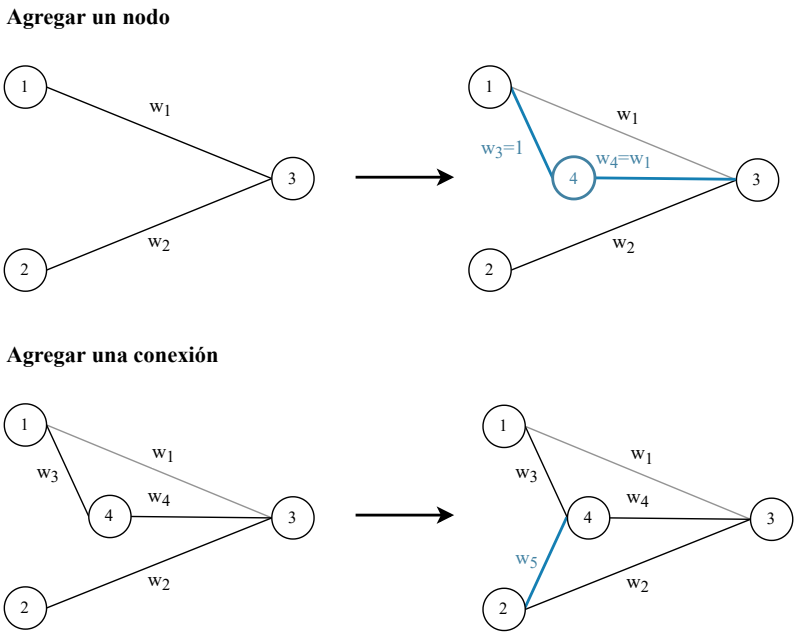


Figura 2.3: Ejemplos de los operadores de mutación estructural de NEAT.

2.5.4. Especiación

La especiación en NEAT busca mantener diversidad en las soluciones encontradas. La idea detrás de esto proviene de que, al agregar nuevos nodos o conexiones durante el proceso de mutación, estas nuevas estructuras pueden tomar algunas generaciones en poder competir con las estructuras viejas, por lo que pueden tener pocas o nulas probabilidades de reproducirse en su siguiente generación. Por lo que, se propone separar las estructuras por especies, y que cada especie tenga un valor de aptitud promediado que permita proteger nuevas estructuras e incrementar el espacio de exploración del algoritmo.

Para dividir los individuos en especies, se propuso la métrica de distancia de compatibilidad, la cual compara dos individuos de acuerdo con el número de nodos disjuntos y excedentes y el valor de los pesos de los nodos que coinciden. La Ecuación 2.4 muestra la ecuación para obtener la distancia de compatibilidad:

$$\delta = \frac{c_1 E}{N} + \frac{c_2 D}{N} + c_3 \bar{W}, \tag{2.4}$$

siendo E y D la cantidad de conexiones excedentes y disjuntas respectivamente, $\bar{W}$ la suma de diferencias entre los pesos de las conexiones que coinciden en ambos padres y N el número de conexiones del genoma más grande. Los coeficientes c_1, c_2 y c_3 permiten ajustar el peso de cada uno de los términos.

Durante cada iteración, se elige un representante aleatoria de cada especie y se utiliza la distancia de compatibilidad para comparar a nuevos individuos y definir a qué especie corresponden de acuerdo a un umbral definido. En caso de que un individuo nuevo no sea similar a ninguna especie existente se crea un nuevo conjunto para esta especie. Después de clasificar a la población por especies, a cada especie se le asigna un valor de aptitud compartida, la cual se calcula como el promedio de la aptitud de cada uno de los individuos contenidos en dicha especie. Este valor de aptitud compartida define la probabilidad de reproducción de dicha especie. De este modo, poblaciones con pocos individuos tienen oportunidad de competir.

El proceso de cruza se realiza entre individuos de la misma especie, pero el algoritmo permite definir un valor de probabilidad para cruza inter-especie, eligiendo un padre de otra especie. En el caso que la especie sólo tenga un individuo, el segundo padre también se elige de otra especie.

El miembro de cada especie con mejor aptitud se define como su campeón, el cual para especies con una cantidad de individuos mayor a un umbral definido, se copia sin ningún cambio a la siguiente generación. Este proceso se describe en el Algoritmo 3. Por otro lado, si durante cierto número de generaciones no se encuentra un individuo con un mejor valor de aptitud que el campeón, esta especie se define como estancada y no podrá reproducirse en la siguiente generación.

El Algoritmo 4 describe el proceso de especiación. El método toma como entrada a la nueva población de individuos y al conjunto viejo de especies. La primera parte del algoritmo elige un representante y a un campeón por cada

una de las especies, incrementa la cuenta de generaciones de estancamiento y quita los elementos de dicha especie. La segunda parte del algoritmo toma a la nueva población y busca si los individuos pertenecen a una especie comparando con su representante y utilizando la distancia de compatibilidad δ. En caso de que sí exista, se agrega a dicha especie y se reinicia el conteo de generaciones de estancamiento. Además, si el nuevo elemento tiene una aptitud mayor al campeón de dicha especie, este se convierte en el nuevo campeón. En caso de que el individuo no pertenezca a ninguna especie existente, se crea una nueva especie a partir de dicho individuo y se agrega al conjunto de especies.

Algoritmo 3 NEAT: Elitismo(S)

```
Q ← ∅                                   ▷ Q es el conjunto de descendientes
λ ← μ                                   ▷ μ es el tamaño de la población
para cada s ∈ S hacer
    si tamaño(s) < α entonces
        Q ← Q ∪ s_campeón
        λ ← λ − 1
devolver Q, λ
```

Algoritmo 4 NEAT: Especiación(Q, S)

```
para cada s ∈ S hacer
    s_representante ← random(s)
    s_campeón ← argmax_{x∈s} x_aptitud
    s_estancamiento ← s_estancamiento + 1
    s ← ∅
para cada q ∈ Q hacer
    agregado ← falso
    para cada s ∈ S hacer                          ▷ para cada especie s
        si δ(q, s_representate) <= β entonces      ▷ si q pertenece a s
            s ← s ∪ q
            agregado ← verdadero
            x ← s_campeón
            si q_aptitud > x_aptitud entonces
                s_campeón ← q
                s_estancamiento ← 0
            Romper el ciclo
    si agregado = Falso entonces                   ▷ si q no pertenece a ninguna s ∈ S
        r ← {q}
        r_representante ← q
        r_campeón ← q
        r_estancamiento ← 0
        S ← S ∪ r
devolver S
```

2.6. FS-NEAT y N3O

Feature Selective NEAT (FS-NEAT) es una variante de NEAT [17], cuyo objetivo es incluir un proceso de selección de características de manera simultánea al ir evolucionando las ANN. La diferencia entre NEAT y FS-NEAT es la manera en que se inicializa la población. Mientras que en NEAT todos los individuos comienzan como una red densamente conectada entre los nodos de entrada y los de salida, en FS-NEAT únicamente se crea una conexión entre un nodo de entrada y uno de salida. Después, durante la ejecución del algoritmo, el operador de mutación que genera nuevas conexiones puede conectar un nodo de entrada a algún nodo oculto o salida y de este modo seleccionar dicha característica. Al empezar con una estructura mínima se espera que el número de características seleccionadas se minimice.

Dando un paso adelante, N3O es una variante a FS-NEAT, cuyo nombre significa "New 3 Operators" y la cual fue diseñada por Grisci et. al. en el 2019 [18]. Esta variante incluyó una modificación al operador de cruza y dos nuevos operadores de mutación estructural. Además, sustituyó el elitismo mediante un porcentaje de la población con mejor desempeño, en lugar del elitismo por campeón de especie como en el algoritmo original. En la siguiente sección se describirá a detalle los operadores evolutivos en N3O.

2.6.1. Operadores evolutivos en N3O

En N3O hay tres cambios principales: una modificación al operador de cruza y dos nuevos operadores de mutación. La modificación al operador de cruza consiste en agregar la posibilidad de que nodos de entrada del padre menos apto puedan ser incluidos en el nuevo individuo. Para ello, se define un umbral y se utiliza un proceso aleatorio para decidir si un nodo de entrada es agregado.

La metodología de N3O, incluye un filtrado estadístico de características previo a la ejecución del algoritmo, utilizando la prueba H de Kruskal Wallis (KW) y descartando aquellas características con un valor p mayor o igual a 0.01. Sin embargo, los valores p obtenidos también son utilizados durante la ejecución en uno de los nuevos operadores de mutación propuestos. Los operadores de mutación se describen a continuación:

- **Agregar una entrada de forma guiada**: Este operador permite agregar una nueva entrada a la red, seleccionando un nodo existente y creando una nueva conexión a dicho nodo. Aquí se utiliza el valor p obtenido en la prueba KW para determinar la probabilidad de selección de cada una de las características.

- **Cambiar una entrada**: Se permite sustituir una entrada existente por una nueva en la red.

Al incluir operadores específicos para agregar nuevas características en la red, N3O puede explorar más el espacio de características en comparación a

FS-NEAT. Además, una diferencia durante la implementación es que en FS-NEAT se debe incluir todos los nodos de entrada al inicializar cada individuo, puesto que depende del operador para agregar un nuevo nodo para incluir nuevas características, mientras que en N3O los individuos puede inicializar únicamente con un nodo de entrada y las características se agregarán mediante los nuevos operadores. Lo anterior, permite reducir la complejidad de las redes.

La Figura 2.4 muestra ejemplos de los operadores de mutación de N3O. En el primer ejemplo se muestra el operador para agregar una nueva entrada, para este se utiliza la probabilidad de selección que está en función al valor p obtenido en la prueba KW y en este caso se selecciona la entrada con el índice 5 y se agrega al nodo 8 creando la nueva conexión w_6. El segundo ejemplo representa el funcionamiento del operador para sustituir una entrada, y en el ejemplo se cambia la entrada 1 por la entrada 4, sin embargo, un detalle importante es que también las conexiones cambian. Esto es debido a que, si se mantuvieran las mismas conexiones, en el proceso de cruza las conexiones podrían concordar en el número de innovación, pero tendrían diferentes nodos de entrada. Por ello, se debe buscar si las conexiones de la nueva entrada ya existen en otros individuos de la población para mantener el mismo número de innovación o en caso contrario, asignarles un nuevo número de innovación.

2.6.2. Función de costo en N3O

La función de costo utilizada en N3O como métrica para evaluar el desempeño de las ANNs generadas fue la entropía cruzada. Sin embargo, modificaron la ecuación original para calcular la función de pérdida para cada clase y después calcular el promedio, con el objetivo de evitar un sesgo por el desbalance de muestras de las clases (problema común en los conjuntos de datos de microarreglos). La Ecuación 2.5a muestra la función de entropía cruzada siendo Q el conjunto de clases, q cada una de las clases, n^q el número de muestras de la clase q, y_i la etiqueta verdadera de la muestra i y a_i el valor de salida de la ANN con la muestra i; adicional a eso un término de regularización L2 fue agregado, siendo una técnica común para evitar sobreajuste (otro problema común al analizar microarreglos) y está expresado en la Ecuación 2.5b, siendo λ el parámetro de regularización, n el número total de muestras, c el número de conexiones y w_k el peso de la conexión k. El término de regularización se divide entre la cantidad de conexiones habilitadas, debido a que el número de conexiones puede variar para cada individuo y de este modo, no penalizar más a aquellas ANNs con mayor cantidad de ellas.

ya que el número de conexiones no es constante entre los individuos y de este modo, no penalizar más a aquellas ANNs con mayor número de ellas.

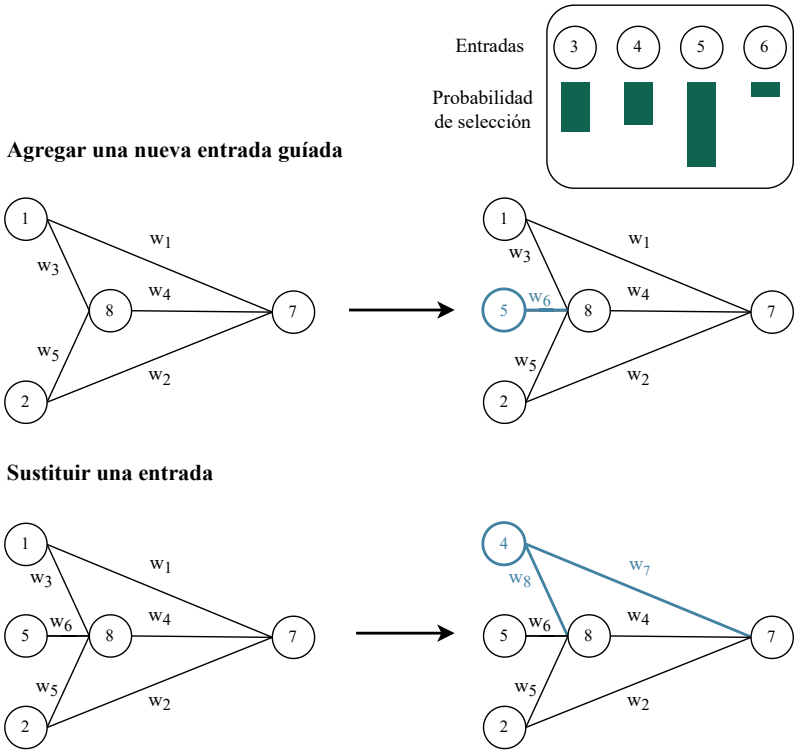


Figura 2.4: Ejemplos de los operadores de mutación agregados en N3O.

$$g(a,y) = \frac{1}{|Q|}\sum_{q\in Q}\left\{-\frac{1}{n^q}\sum_{i=1}^{n^q}\left[y_i \ln a_i + (1-y_i)\ln\left(1-a_i\right)\right]\right\} \tag{2.5a}$$

$$+ \frac{\lambda}{2n}\frac{1}{c}\sum_{k=1}^{c} w_k^2 \tag{2.5b}$$

2.7. Optimización multiobjetivo

En el día a día, nos enfrentamos a diversas tareas a resolver, ya sea elegir el desayuno de hoy o qué ruta tomar hacia el trabajo, para elegir es necesario tomar en cuenta distintos factores. No sólo se elige el desayuno de acuerdo con lo que más se nos antoja, sino que a veces se tiene que tomar en cuenta qué es lo más sano, o no sólo se toma en cuenta el tiempo para ir al trabajo, sino que quizás también te interese gastar lo menos posible en llegar. Resulta sencillo agregar nuevos objetivos a los problemas, sin embargo, complican poder resolverlos, ya que usualmente cuando tienes dos o más objetivos tienden a tener conflicto entre sí. Regresando al ejemplo del desayuno, por lo general, o al menos en el caso de quien escribe, el platillo favorito no suele ser el más sano, y viceversa, el platillo más sano no suele ser el más apetitoso. Es entonces, que se requieren métodos que puedan encontrar soluciones que optimicen tomando en cuenta más de un objetivo.

La idea de una solución óptima difiere entre un problema de un objetivo a uno multiobjetivo, ya que como se mencionó anteriormente, los objetivos tienden a tener conflictos entre sí, por lo que se necesario un compromiso entre los objetivos. Lo anterior se puede expresar como:

$$f(x) = [f_1(x), f_2(x), ..., f_k(x)], \tag{2.6}$$

donde $f(x)$ se compone de los objetivos a optimizar en un espacio $\mathbb{R}^k$ y x representa las variables de decisión

$$x = [x_1, x_2, ..., x_n], \tag{2.7}$$

en un espacio $\mathbb{R}^k$. El problema multiobjetivo se puede expresar de forma general como:

$$\text{mín}\,\{f(x)\}, \tag{2.8}$$

tal que satisfaga:

$$g_i(x) \leq 0, i = 1, ..., I, \tag{2.9a}$$

$$h_i(x) = 0, i = 1, ..., J, \tag{2.9b}$$

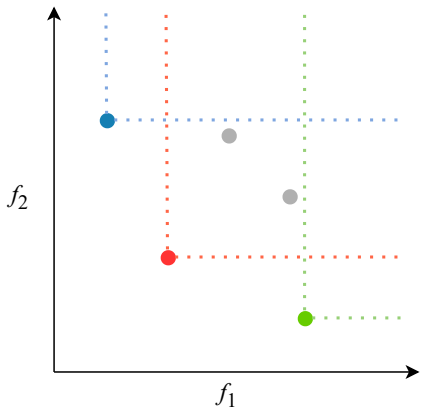


Figura 2.5: Ejemplo ilustrativo para describir el concepto de optimalidad de Pareto. Las líneas punteadas definen el espacio que domina cada solución. La solución roja, azul y verde no se encuentran dominadas y forman el frente de Pareto. Las soluciones grises, son dominadas por la roja.

donde g representa I restricciones de desigualdad y h representa J restricciones de igualdad.

La definición de optimalidad que comúnmente se utiliza en optimización multiobjetivo fue propuesta en 1881 por Francis Ysidro Edworth y fue generalizada por Vilfred Pareto 1896, y es conocida como optimalidad de Edgeworth-Pareto o, más comúnmente como optimalidad de Pareto [19]. Dicha definición de optimalidad es simple, un conjunto de soluciones óptimas de Pareto es aquella en las que no hay una solución fuera o dentro de dicho conjunto la cual sea mejor en cada uno de los objetivos que las soluciones dentro del conjunto. La Figura 2.5 ayuda a ilustrar este concepto, en ella se muestra una gráfica con diferentes soluciones para un problema con 2 objetivos: f_1 y f_2. Se busca minimizar ambos objetivos y se muestran 5 soluciones. La solución azul sería la mejor solución para el objetivo f_1 mientras que la verde la del objetivo f_2. La solución roja no es la mejor en ninguno de los objetivos, pero es mejor que la solución verde respecto al objetivo f_1 y que la azul respecto al objetivo f_2. Además, la solución roja es mejor en ambos objetivos sobre las soluciones grises, esto se describe como dominancia de Pareto. La relación de dominancia se describe con el símbolo "$\prec$". Finalmente, los puntos azul, rojo y verde forman el frente de Pareto, es decir, que son el conjunto de soluciones las cuales no existe otra solución que las domine.

En la práctica, es complicado encontrar el frente de Pareto verdadero debido a que las funciones con muchos objetivos se consideran como cajas negras y por la complejidad del espacio de búsqueda, por lo que en la mayoría de los casos únicamente podemos obtener una aproximación del frente de Pareto en sí [20]. Los algoritmos evolutivos han demostrado un gran potencial para resolver

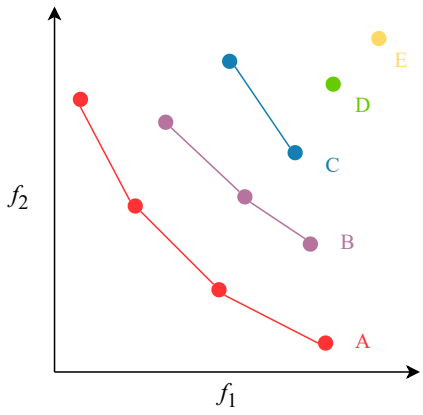


Figura 2.6: Ejemplo ilustrativo para describir el ordenamiento en frentes de Pareto. Cada frente se distingue por un color y letra diferente. El frente A, es aquel cuyas soluciones no son dominadas por ninguna otra solución. El B, es el frente de Pareto sin tomar en cuenta a las soluciones del frente A. El frente C, se forma ignorando las soluciones del A y B, y así sucesivamente.

problemas multiobjetivo, debido a que son basados en poblaciones, permiten encontrar múltiples soluciones del frente de Pareto simultáneamente.

2.7.1. Ordenamiento por dominancia de Pareto

Una de la manera en que podemos ordenar las soluciones es por distintos frentes de Pareto, es decir, genera el primer frente a partir de todas las soluciones que no son dominadas por ninguna otra solución, después genera la segunda a partir de las soluciones que únicamente son dominadas por soluciones del primer frente, y siguen generando frentes hasta que todas las soluciones hayan sido clasificadas en algún frente. La Figura 2.6 muestra de la manera en que las soluciones se ordenan de acuerdo con diferentes frentes. En ella podemos observar que los puntos rojos no tienen otra solución que los domine, por lo que forman el primer frente de Pareto. Si ignoramos los puntos rojos, los morados formarían el siguiente frente de Pareto. Con esto en mente, cada color representa un diferente frente de Pareto. Recalquemos que los frentes pueden estar formados por una o más soluciones.

Existen distintos algoritmos evolutivos multiobjetivo que tienen métodos para ordenar las soluciones por dominancia de Pareto, pero uno de los más populares es el "fast-non-dominated-sorting" propuesto en NSGA-II, cuyas siglas provienen de "Non-dominated Sorting Genetic Algorithm II" [21]. Este procedimiento, el cual se describe en el Algoritmo 5 comienza comparando si existe dominancia entre cada una de las soluciones y se generan dos entidades: el primero, denominado como n_p, expresa la cantidad de soluciones que dominan a

Algoritmo 5 Fast-Non-Dominated Sorting(Q)

```
1:  para cada p ∈ P hacer
2:      S ← ∅
3:      n_p ← 0
4:      para cada q ∈ P hacer
5:          si p ≺ q entonces                          ▷ Si p domina a q
6:              S_p ← S_p ∪ {q}                        ▷ Se agrega q a S_p
7:          si no si q ≺ p entonces                    ▷ Si q domina a p
8:              n_p ← n_p + 1                          ▷ Se incrementa n_p
9:      si n_p = 0 entonces
10:         F_1 ← F_1 ∪ {p}
11: i ← 1
12: mientras F_i ≠ ∅ hacer
13:     Q ← ∅
14:     para cada p ∈ F_i hacer
15:         para cada q ∈ S_p hacer
16:             n_q ← n_q − 1
17:             si n_q = 0 entonces                    ▷ Si n_q = 0
18:                 Q ← Q ∪ {q}                        ▷ q pertenece al siguiente frente
19:     i ← i + 1
20:     F_i ← Q
```

solución p, y el segundo representa al conjunto de soluciones que son dominadas por p descrito como S_p. Siendo M la cantidad de objetivos y N el tamaño de la población este paso tendrá una complejidad computacional de $O(MN^2)$. El segundo paso, consiste en generar los frentes F en conjuntos de soluciones de acuerdo con el nivel en el que pertenecen. Para ello, primero se genera el primer frente F_1 que estará formado por todas las soluciones p con $n_p = 0$ y se decrementa el valor de n una unidad para los elementos de S_p. Aquellas soluciones que al decrementar su valor de n llegan a cero, serán parte del siguiente frente. Se repite el proceso hasta que todas las soluciones hayan sido agregadas a un frente. La complejidad computacional de este segundo paso es de $O(N^2)$, por lo que el procedimiento completo tendrá una complejidad de $O(MN^2)$.

2.7.2. Archivos externos

Para algoritmos evolutivos multiobjetivo, es posible descartar soluciones no dominadas o subóptimas a través de las generaciones. Es por ello, que es común utilizar estrategias que permitan almacenar soluciones relevantes en archivos externos [22] [23]. Estas estrategias definen el proceso para seleccionar o descartar selecciones con el propósito de obtener un conjunto final de soluciones con características determinadas. Por ejemplo, se podría incluir un archivo externo que almacene todas las soluciones no dominadas encontradas a lo largo de la ejecución del algoritmo, o por otro lado, quizás se deseé un conjunto de solu-

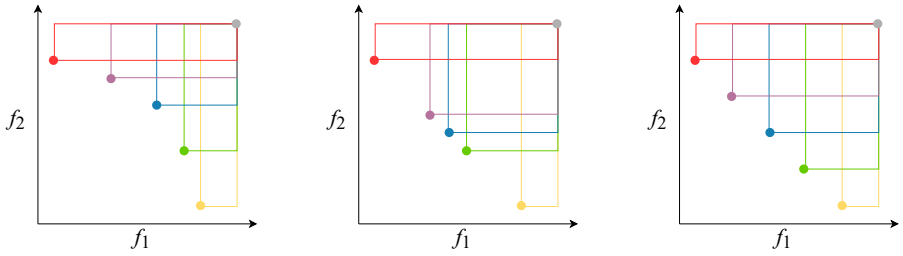
ciones de un tamaño máximo pero que garantice una buena distribución o una mayor diversidad entre ellas, o incluso, hay estrategias que permiten almacenar soluciones subóptimas que podrían ser relevantes para el tomador de decisión (el que toma una solución del conjunto de soluciones final).

2.7.3. Hipervolumen

Para evaluar la calidad del frente de Pareto de las soluciones encontradas son utilizadas métricas que se denominan indicadores de calidad. Uno de ellos, es la métrica $\mathscr{S}$ o hipervolumen el cual fue propuesto por primera vez por Ziztler y Thiele en 1998 [24]. Este indicador utiliza un punto de referencia, el cuál debe ser dominado por cada uno de los puntos del frente de Pareto a evaluar. Después, se calcula la unión de los espacios entre el punto de referencia y cada uno de los puntos del frente de Pareto. La Ecuación 2.10 representa de forma matemática el cálculo del hipervolumen:

$$HV(Y) = \left\{ \bigcup_i h(z_{ref}, y_i) \in Y \right\}, \tag{2.10}$$

donde Y contiene cada uno de los vectores del frente de Pareto y z_{ref} es el vector de referencia. h es una función para calcular el espacio entre el vector de referencia y el frente de Pareto, en el caso de dos objetivos esta sería el área, para tres objetivos el volumen y la complejidad de la función crecerá según la cantidad de objetivos. Al buscar maximizar el hipervolumen del frente de Pareto, se estará optimizando la proximidad al frente además de su distribución. La Figura 2.7 muestra 3 conjuntos de soluciones para un problema con 2 objetivos, f_1 y f_2, y se busca minimizar el valor en ambos. Cada conjunto tiene 5 soluciones representadas por un color distinto (azul, morado, rojo, verde y amarillo) y hay un punto gris que representa el vector de referencia. Los rectángulos formados desde el valor de referencia y las soluciones ayudan a visualizar el área que se forma entre ellos, y la unión de todas esas áreas representan el valor del hipervolumen (en este caso hiper-área). Podemos observar que las soluciones en la Figura 2.7c, se aproximan más al cero con respecto a las soluciones de la Figura 2.7a y tienen una mayor distribución entre ellas que con respecto a las soluciones de la Figura 2.7b, en ambos casos el hipervolumen incrementa mientras que las soluciones tienen una mayor proximidad y distribución. Cabe resaltar que cualquier solución dominada dentro de este conjunto, no aportaría nada al valor del hipervolumen, y por ello, el indicador sólo toma en cuenta el frente de Pareto.



(a) $HV = 11{,}81$ (b) $HV = 14{,}5$ (c) $HV = 15$

Figura 2.7: Ejemplo de 3 conjuntos de soluciones para un problema con dos objetivos f_1 y f_2 para ilustrar el concepto del hipervolumen.

2.8. SMS-EMOA

Existen algoritmos evolutivos multiobjetivo que son basados en indicadores, es decir, que utilizan un indicador para evaluar el frente de Pareto de las soluciones y buscan optimizar dicho valor. Un ejemplo de este tipo de algoritmos es SMS-EMOA [20], el cual utiliza el hipervolumen como indicador. Este es uno de los algoritmos más populares en el campo de multiobjetivo, principalmente para problemas de 2 o 3 objetivos, puesto que el cálculo del hipervolumen incluye un gran costo computacional.

La estructura principal de SMS-EMOA se describe en el Algoritmo 6. Primero se genera una población inicial P de tamaño μ y entra al ciclo principal, en el cual hay dos pasos principales: el primero es generar un nuevo individuo q y agregarlo a la población, y el segundo es identificar el individuo el cual aporta el menor valor de hipervolumen en la población. Para generar un nuevo individuo se debe tomar en cuenta el problema a resolver para determinar la codificación genética a utilizar y posteriormente, los operadores de selección, cruza y mutación. Para reducir la población, primero se utiliza el método de ordenamiento por dominancia de NSGA-II ("fast-non-dominated-sorting" descrito en el Algoritmo 5 en la Subsección 2.7.1) para clasificar cada individuo de acuerdo con el nivel del frente de Pareto al que pertenecen, y posteriormente, se calcula el aporte del hipervolumen de los individuos del último frente para determinar aquel que aporte menos y descartarlo de la población.

El Algoritmo 7 describe la función de reducir, la cual es la parte más característica de SMS-EMOA. En el algoritmo Q se refiere a la población de entrada, la cual se ordena del primer frente $\mathscr{R}_0$ al último frente $\mathscr{R}_v$. Después se busca al individuo r del último frente, el cuál minimiza el valor de la diferencia entre el hipervolumen del último frente con o sin dicho individuo como se describe en la Ecuación 2.11.

$$\Delta_{\mathscr{S}}(s, \mathscr{R}_v) \coloneqq \mathscr{S}(\mathscr{R}_v) - \mathscr{S}(\mathscr{R}_v \backslash \{s\}) \tag{2.11}$$

Algoritmo 6 SMS-EMOA

1: $P_0 \leftarrow$ inicializar()
2: $t \leftarrow 0$
3: **mientras** la condición de paro no sea verdad **hacer**
4: $\quad q_{t+1} \leftarrow$ generar(P_t)
5: $\quad P_{t+1} \leftarrow$ reducir($P_t \cup q_{t+1}$)
6: $\quad t \leftarrow t + 1$

Algoritmo 7 Reducir(Q)

1: $\{\mathscr{R}_1, ..., \mathscr{R}_v\} \leftarrow$ ordenamiento_de_no_dominados(Q)
2: $r \leftarrow min_{s \in \mathscr{R}_v}[\Delta_{\mathscr{S}}(s, \mathscr{R})]$
3: **devolver** $(Q \backslash \{r\})$

El marco de trabajo de SMS-EMOA junto con la metodología en N3O fueron tomadas como base e inspiración para el desarrollo del presente proyecto.

2.9. Configuración automática de algoritmos

El funcionamiento de meta-heurísticas, como los algoritmos evolutivos, depende de múltiples parámetros. Un reto común al utilizar estas metodologías es elegir aquellos parámetros que optimicen el desempeño del algoritmo para determinado tipo de tareas. Es por ello por lo que existen metodologías que permiten encontrar la configuración óptima de parámetros de manera automática.

El problema de configuración de algoritmos se puede describir de la siguiente manera [25]: dado un algoritmo A cuyo funcionamiento depende de los parámetros $p_1, ..., p_k$ y cuyas posibles combinaciones forman un espacio de búsqueda C, encontrar $c* \in C$ tal que optimice A en un conjunto de instancias I de acuerdo a una cierta métrica m.

Entre las metodologías más populares para la realizar la configuración automática de algoritmos están los procedimientos de carreras o *racing* que se introducen a continuación.

2.9.1. Procedimientos de carreras

Los procedimientos de carreras o *racing* en configuración automática de algoritmos, prueban distintas configuraciones en distintas instancias y descartan aquellas que muestren un desempeño significativamente peor que aquella con mejor desempeño [25].

F-Race es una metodología muy popular de esta categoría [26], la cual utiliza la prueba de Friedman para comparar las distintas configuraciones y detectar si hay diferencia significativa entre ellas, y mediante una prueba post hoc, descartar aquellas configuraciones con peor desempeño. Una variante de esta metodología es *F-Race* iterativo, o mejor conocida como *iRace* [27]. Esta variante,

como su nombre lo dice, ejecuta *F-Race* de manera iterativa, y se apoya de un modelo probabilístico para obtener muestras de los parámetros y el cual se actualiza en cada ejecución de acuerdo a las mejores configuraciones encontradas. Este procedimiento se repite hasta encontrar una configuración óptima o hasta un número de ejecuciones del algoritmo definido por el usuario.

Capítulo 3

Antecedentes

En este capítulo se describirán metodologías previamente presentadas en la literatura relacionadas a esta tesis. En la sección 3.1 se presentan diferentes metodologías utilizadas la reducción de características y clasificación de conjuntos de datos de microarreglos. Después en la sección 3.2 se describen algoritmos multiobjetivo propuestos previamente que han utilizado conceptos de NEAT en su diseño.

3.1. Métodos para la reducción de características y clasificación de microarreglos

Como se mencionó anteriormente, los conjuntos de datos de microarreglos tienden a tener dos problemas principalmente: la enorme cantidad de características en comparación al número de muestras y un desbalance entre la cantidad de muestras de cada una de las clases. Ambos problemas dificultan generar modelos de clasificación precisos y confiables. Es por ello por lo que este problema se divide en dos tareas principales: la reducción de características y la clasificación. A continuación, presentamos diferentes metodologías que han sido propuesta para la reducción de dimensión en conjuntos de datos de microarreglos.

3.1.1. Métodos de extracción de características

Existen dos estrategias para reducir las características de un conjunto de datos: la primera es buscar las características más relevantes y la segunda es extraer nuevas características que mantengan las propiedades del conjunto original. Un ejemplo de este último es el análisis de componentes principales (PCA por sus siglas en inglés "Principal Component Analysis"), el cual ya ha sido propuesto como método para la reducción de características en microarreglos de cáncer con métodos de clasificación como el de máquinas de soporte vectoriales (SVM por "Support Vector Machines") [28] o por ANNs [29], y en ambos casos,

mostrando ser un método eficiente para reducir las características y facilitando el entrenamiento de los modelos de clasificación. Otro método de extracción de características es el análisis de componentes independientes (ICA por su traducción al inglés "Independent Component Analysis"), y también ha sido utilizado para la selección de características de microarreglos en [30], donde primero se realizó ICA al conjunto de entrenamiento para reducir las características y después se realizó un segundo proceso de reducción de características utilizando el algoritmo de colonia de hormigas (ACO por "Ant Colony Optimization") y el clasificador bayesiano ingenuo. PCA e ICA son ambas metodologías muy populares de reducción de características, sin embargo, una ventaja que tienen los métodos de selección de características sobre los de extracción, es que al mantener las características originales se puede intentar interpretar la información. En el caso específico de microarreglos de ADN, el encontrar una relación entre ciertos genes y enfermedades puede ser útil para otras áreas de investigación como la medicina o la industria farmacéutica. A dichos genes se les denomina como biomarcadores.

3.1.2. Métodos basados en filtros

Una primera aproximación es utilizar alguna metodología para filtrar genes no relevantes previo a entrenar el modelo de clasificación. La forma más popular de hacer esto es por medio de pruebas estadísticas, aplicándolas a cada gen de manera independiente y utilizando el valor p como umbral para decidir si un gen se mantiene o se elimina del conjunto de datos. Maniruzzaman et al. [31] presentaron un sistema que utilizaba pruebas estadísticas como método de reducción de características. Realizaron experimentos entre 4 pruebas estadísticas y 10 modelos de clasificación en conjuntos de datos de cáncer de colon buscando encontrar la mejor combinación. Las pruebas estadísticas que se utilizaron fueron la prueba U de Mann-Whitney (MW), la prueba t de estudiante, la prueba H de KW, y la prueba F de Fisher. De los experimentos que realizaron determinaron que la prueba MW y el modelo de árboles aleatorios (RF por "Random Forest") juntos obtenían los mejores resultados. Algo a notar en sus resultados es que la prueba estadística que mejores resultados obtuvo al combinar con ANNs fue KW. Otra prueba estadística que ha sido utilizada para el análisis de microarreglos es RankProd, la cual es una prueba no-paramétrica que combina conjuntos de datos para mejorar su precisión, y fue propuesta en [32] junto con un clasificador basado en neuroevolución.

Otro método que ha sido propuesto es el del análisis diferencial de expresión genética [33], el cual es un método estadístico que ajusta sus parámetros al conjunto de datos y determina si la variación entre la expresión genética es relevante o despreciable, y fue utilizado junto con un modelo de ANN que ensamblaba otros 5 modelos de clasificación para predecir muestras de microarreglos de cáncer [34]. Los modelos utilizados fueron SVM, RF, árbol de decisión (DT por "Decision Tree"), k-vecinos cercanos (kNN por "k-Nearest Neighbors") y árboles de decisión con potenciación de gradiente (GBDT por "Gradient Boosting Decisition Tree"), mostrando que al combinar los modelos se podía mejorar

la precisión.

Una metodología de filtro diferente fue presentada en [35], donde utilizaron el puntaje Laplaciano [36] para filtrar los genes relevantes y una red neuronal convolucional (CNN por "Convolutional Neural Network") para la clasificación de los microarreglos. Algo interesantes es que, el uso de una capa convolucional en una ANN es también, un proceso de extracción de características, por lo que esta metodología combina ambos tipos de métodos mencionados hasta ahora.

3.1.3. Métodos basados en agrupamiento

Otro tipo de metodologías han intentado identificar genes similares y con ello, poder descartar información redundante del conjunto de datos o, por otro lado, determinar genes que podrían ser considerados como ruido. En [37], se propuso utilizar el algoritmo de k-medias como método de agrupamiento de genes y se utilizó RF como método posterior para la clasificación de microarreglos. En otro trabajo, se propuso utilizar el algoritmo de conjuntos de agrupamiento proyectivo (PCE por Projective Clustering Ensemble), mostrando que tenía una gran capacidad de identificar genes irrelevantes o ruido en el conjunto de datos [38].

3.1.4. Métodos embebidos

Las metodologías embebidas intentan utilizar al modelo de clasificación para establecer un criterio de relevancia para cada característica [6]. En [39], se propuso utilizar el método de eliminación recursiva de características (RFE por "Recursive Feature Elimination") y el de regresión logística aleatoria (RLR por "Randomized Logistic Regression"), los cuales van reduciendo el número de características mediante entrenar modelos de aprendizaje máquina e identificar aquellos que tienen un menor peso para la clasificación. Se probaron en dos conjuntos de datos de cáncer de mamá y con 8 diferentes algoritmos de clasificación, pero el mejor desempeño lo tuvieron junto con SVM. Sin embargo, entrenar múltiples veces el modelo de SVM implica un gran costo computacional como un largo tiempo de ejecución.

Un algoritmo popular que igual ha sido aplicado para la reducción de características de microarreglos de cáncer es el de SVM basado en RFE (SVM-RFE) [40], el cual entrena modelos de SVM y remueve la característica menos importante indicada por el clasificador en cada iteración. En [41], se propuso una variación de este mismo algoritmo, la cual reduce la cantidad de iteraciones y usa una versión simplificada de SVM, lo cual reduce el tiempo de ejecución del algoritmo.

3.1.5. Métodos envolventes

Además de los métodos basados en filtro y los embebidos, hay otra categoría de métodos para selección de características llamados envolventes o "wrappers". Este tipo de metodologías buscan optimizar al modelo de clasificación como

parte de la selección de características, es decir, utiliza métodos de optimización para encontrar al subconjunto de características que maximice el desempeño del modelo de clasificación. A diferencia de las metodologías basadas en filtros y similar a las embebidas, requiere entrenar el modelo de clasificación múltiples veces lo que implica un mayor costo computacional. Por otro lado, a diferencia de las metodologías embebidas, sólo utiliza al modelo de clasificación como método de evaluación y no como forma para determinar qué características descartar.

Tanto para las metodologías envolventes, múltiples algoritmos han sido propuestos para la selección de características. La elección de estos algoritmos es importante, ya que el espacio de búsqueda es inmenso y cada evaluación requiere entrenar al modelo de clasificación lo cual (dependiendo del modelo seleccionado) puede consumir mucho tiempo. Es por ello por lo que encontrar una buena solución en un corto número de evaluaciones es importante y muchas metaheurísticas han sido propuestas para llevar a cabo esta tarea.

Muchas de las metodologías envolventes que han sido propuestas en la literatura incluyen una metodología de filtro previo para reducir el espacio de búsqueda y el tiempo de ejecución y por ello se les conoce como metodologías híbridas [18] [42] [43] [44] [45] [46] [47] [48] [49] [50].

Dentro de estos métodos se encuentran dos principales aproximaciones: algoritmos evolutivos y algoritmos de inteligencia de enjambre. A continuación, se presentan diversos trabajos de cada una de estas categorías.

Métodos basados en algoritmos evolutivos

Se han propuesto múltiples metodologías que utilizan algoritmos evolutivos para la selección de características. En [51] se presentó un trabajo en el cual se utilizó el algoritmo genético junto con el puntaje del algoritmo SLI-γ que se usaba para inicializar la población de acuerdo con el 1 % de características con mejor puntaje con el gen activado. En otra metodología propuesta por Ghosh et al. [42] se utilizaron 3 filtros de manera independiente (ReliefF, chi-cuadrada y simetría incierta), y combinaba los subconjuntos obtenidos para después utilizar un método envolvente con un algoritmo genético. En comparación, Saeed et al. [43] también presentó una metodología con múltiples funciones para evaluar cada característica de manera independiente, pero usó un sistema de votos para definir cuáles filtrar y además se utilizó la cantidad de votos para modificar la probabilidad de aplicar operadores evolutivos en cada característica.

En [44], se propuso utilizar el micro algoritmo genético posterior a un método de filtro basado en la relación de ganancia. Esta versión del algoritmo genético utiliza una población pequeña en comparación al algoritmo original y permite reiniciar la población cuando converge en un mínimo local. Otra variación del algoritmo genético es su versión adaptativa, que modifica la probabilidad de cruza y mutación durante la ejecución del algoritmo para controlar la exploración y explotación del algoritmo. Shukla et al. [45] propuso una metodología que primero utilizaba un método de filtro basado en el algoritmo de maximización de información condicional mutua y después un algoritmo genético adaptativo. Dashtban y Balafar [46] incluyeron la capacidad de adaptar el tamaño del cro-

mosoma e incluía un método de reinicio de población basado en el algoritmo de ascenso de colinas. Para este segundo método también se compararon dos métodos de filtro para dar un puntaje a las características el de Fisher y el de Laplace, siendo el primero el que mejor resultados obtuvo junto con SVM como clasificador para el método envolvente. Más adelante, Pashaei y Pashaei [52] utilizaron el algoritmo, pero con el filtro de puntaje basado en RF, algoritmo que posteriormente se utilizaba como clasificador. En el mismo año, Shukla [47] presentó una nueva metodología en la que utilizaba un método de filtro de ensamble, junto el algoritmo genético adaptativo con múltiples poblaciones en paralelo y una función de costo que utilizaba dos clasificadores y validación cruzada.

Métodos basados en inteligencia de enjambre

Los algoritmos de inteligencia de enjambre intentan modelar comportamiento colectivo observado en la naturaleza. Uno de los más populares es el de optimización por enjambre de partículas (PSO por Particle Swarm Optimization), y múltiples trabajos han utilizado como parte de metodologías envolventes para la selección de características [48] [53] [54] [55] [56] [57]. Prasad et al. [55] propusieron utilizar una versión recursiva de PSO, utilizando los pesos obtenidos por SVM al clasificar los datos, similar al proceso de SVM-RFE, pero en vez de utilizar los pesos generados por SVM para eliminar características, se usa para guiar el funcionamiento de PSO.

Muchos otros algoritmos bioinspirados también han sido utilizados como parte de metodologías para la selección de genes en conjuntos de datos de microarreglos como ACO [49] [58], optimización hormiga león [50], colonia de abejas artificiales [5] [59], optimización de colonia de bacterias [60], optimización por enjambre de golondrinas [61], entre otros.

3.1.6. Métodos basados en aprendizaje profundo

Como se mencionó anteriormente, las CNN pueden ser una estrategia efectiva para la extracción de características, Mostavi et al. probaron 3 diferentes arquitecturas: 1D-CNN, 2D-Vanilla-CNN y 2D-Hybrid-CNN [62]. Para las últimas dos, las muestras de microarreglos se reacomodaron para ser representadas como una matriz de 2 dimensiones. Los modelos mostraron tener un gran nivel de precisión en un conjunto de datos con 34 clases (33 tipos de cáncer y una normal). Además, del modelo de 1D-CNN se realizó una técnica para identificar regiones salientes y con ello identificar los genes relevantes para cada clase de cáncer, siendo en promedio 108 genes por cada clase. Otras aproximaciones basadas en CNN se presentan en [63], [64] y [65].

Una arquitectura de ANNs que también ha sido propuesta para la reducción de dimensionalidad en conjuntos de datos de microarreglos son los autoencoders. Un autoencoder es una arquitectura diseñada para codificar la información de entrada en una representación y más significativa, y después decodificarla para que la información de entrada reconstruida sea tan similar a la original [66].

Ejemplos de autoencoders que han sido utilizados para reducción de características en microarreglos están el autoencoder variacional [67] [68], autoencoder regularizado [69], autoencoder apilado [70] y autoencoder apilado de eliminación de ruido [71] [72]. En los trabajos citados anteriormente, el uso de autoencoders consiste en un proceso de extracción de características, obteniendo nuevas representaciones de los conjuntos de datos utilizados, y en algunos se utilizaron las nuevas características para entrenar modelos de clasificación. Sin embargo, resaltamos dos cosas diferentes que se realizaron en los últimos dos artículos mencionados: en [71], se utilizó el autoencoder para generar nuevas muestras artificiales y con ello incrementar el tamaño de muestras para entrenar los modelos. Por otro lado, en [72], se utilizó el valor de los pesos obtenidos en el codificador para identificar subconjuntos de características relevantes, haciendo de este, no sólo un método de extracción de características, sino que también uno de selección.

Podemos notar dos retos al utilizar modelos de aprendizaje profundo para el análisis de microarreglos, tanto como para la tarea de reducción de características como para la de clasificación: el primero, ya mencionado anteriormente, es la pequeña cantidad de muestras con las que se cuentan en los conjuntos de datos dificulta el entrenamiento de arquitecturas con muchas capaz profundas, y segundo, de acuerdo con Daoud y Mayo [73], es que es difícil definir una arquitectura adecuada debido a que no hay una regla específica que garantice una buena precisión de los modelos.

3.1.7. Métodos basados en neuroevolución

Grisci et al. [74] propusieron utilizar FS-NEAT para la selección de genes y clasificación de microarreglos en 2018, demostrando que el algoritmo podría tener un desempeño competitivo con otros como SVM o una red neuronal densa además de realizar el proceso de selección de características a la par, reduciendo en promedio el 98 % de características en sus experimentos. Un año después propusieron una nueva metodología junto con el algoritmo N3O [18], junto con una metodología de filtro basada en la prueba KW previa a la ejecución de N3O para reducir las características y obtener el valor p de cada característica el cuál era utilizado por los operadores evolutivos del algoritmo. N3O incluye nuevos operadores a FS-NEAT que permiten una mayor exploración del espacio de búsqueda, mejorando el desempeño del algoritmo para la tarea de selección de genes relevantes y además obtener ANNs con un mejor desempeño al clasificar microarreglos.

Hacemos notar que, para este tipo de métodos, no sólo se está seleccionado las características, sino que también se está construyendo la topología de las ANNs que se utilizan para clasificar los microarreglos, por lo que el espacio de búsqueda es mayor, pero reduce el trabajo manual al automatizar el diseño de la ANN. Además, los métodos mencionados se basan en NEAT, el cuál procura minimizar la topología de las ANNs generadas, por lo que el entrenamiento de estas redes podría implicar un menor costo computacional que una diseñada a mano o algunos otros modelos de clasificación como SVM.

3.1.8. Métodos basados en optimización multiobjetivo

Uno de los mayores desafíos al diseñar una metodología para analizar conjuntos de datos desbalanceados (como es el caso de los microarreglos) es el elegir una métrica a optimizar. La mayoría de las metodologías mencionadas hasta ahora utilizan una función de costo como la de entropía cruzada. Pero se podría intentar optimizar la precisión o la exactitud del modelo de la clasificación, o los falsos negativos y los falsos positivos, o intentar minimizar el tamaño del subconjunto de características seleccionadas, o todas las anteriores. Es por ello por lo que metodologías de optimización multiobjetivo también han sido consideradas para la tarea en cuestión.

Las metodologías que utilizan algoritmos multiobjetivo para la selección de características también son, por lo general, metodologías envolventes, y suelen variar el modelo de clasificación, pero también los objetivos a optimizar. Rangasamy et al. [75] propusieron utilizar el algoritmo genético multiobjetivo (MoGA por "Multo-Objective Genetic Algorithm") eligiendo la precisión, el valor-F y la media geométrica como objetivos, y de este modo evitar un sesgo por una métrica en específico.

Otras aproximaciones han propuesto como objetivos la maximización de la precisión del modelo de clasificación y la minimización del subconjunto de características seleccionadas [76] [77] [78] [79]. De este modo, intentar encontrar un subconjunto mínimo de genes que provean información relevante para clasificar los microarreglos de manera precisa. En 2018, Dussaut et al. [76] utilizó estos dos objetivos y comparó diferentes algoritmos evolutivos multiobjetivo, entre ellos: NSGA-II, SPEA2, el algoritmo genético celular para optimización multiobjetivo (MOCell por "MOCell") y el algoritmo multiobjetivo de selección elitista a través de generaciones, recombinación heterogénea y mutación de cataclismo (MOCHC por "MOCHC"), y utilizó kNN como método de clasificación, demostrando una notable ventaja de MOCHC en ambos objetivos contra los otros contendientes, sin embargo, muchas nuevas metodologías han sido propuestas en los últimos años [77] [78] [79].

Una opción diferente es utilizar una función de pérdida junto con el número de características como objetivos como se propuso en [80]. O no limitarse a sólo dos objetivos, Qing et al. [81] propusieron un método basado en coevolución que utilizaba 3 objetivos, el número de falsos positivos, el de falsos negativos y el de las características seleccionadas (todos a minimizar).

Cabe mencionar que en ninguna de las metodologías presentadas se utiliza SMS-EMOA como algoritmo de optimización o al hipervolumen como métrica para evaluar a la población de soluciones obtenidas. SMS-EMOA es un algoritmo multiobjetivo competitivo para 2 o 3 objetivos, pero si la cantidad de objetivos incrementa el costo computacional del cálculo del hipervolumen incrementa exponencialmente.

La Figura 3.1 muestra un diagrama general de las metodologías de reducción de características empleadas para microarreglos mencionadas en esta sección. La línea naranja en el diagrama señala la metodología propuesta en este proyecto, la cuál es híbrida entre un método basado en filtro con la prueba estadística

de KW seguida de un método envolvente multiobjetivo basado en neuroevolución, siendo que los objetivos a minimizar son el número de características seleccionadas y una función de pérdida de las ANNs generadas.

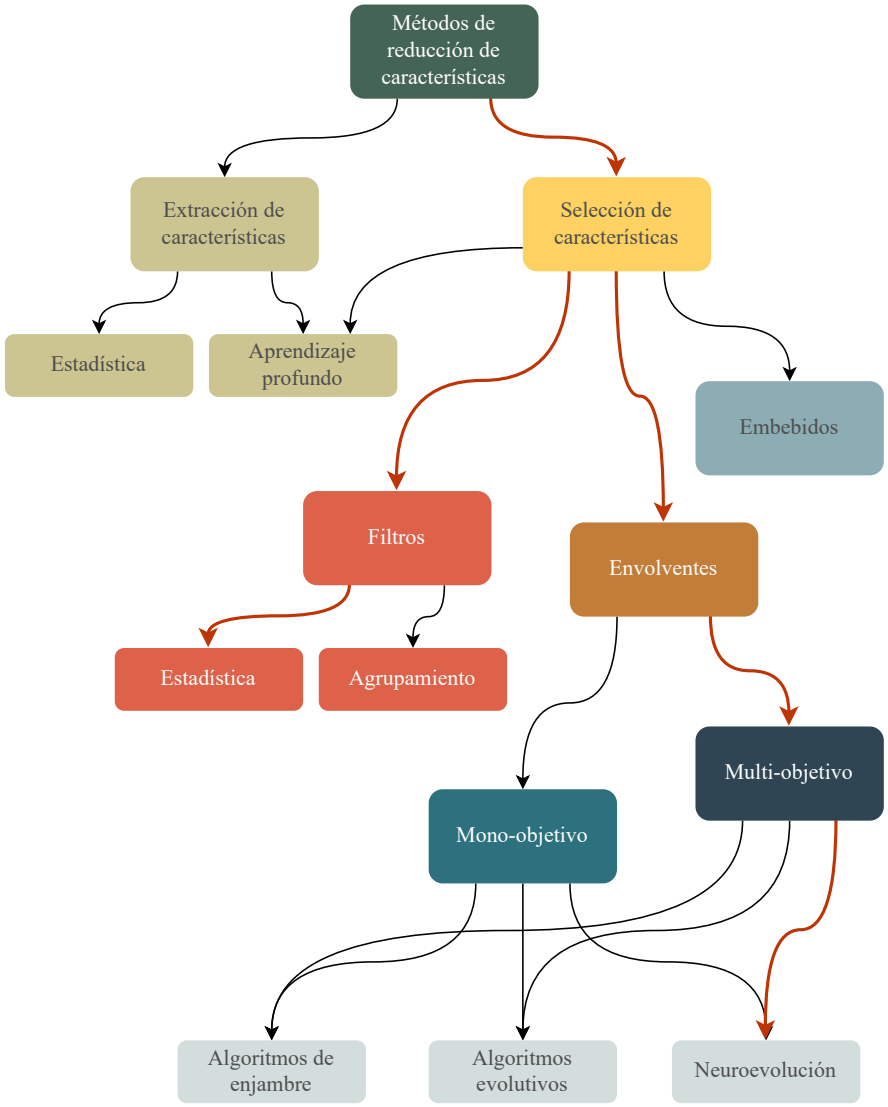


Figura 3.1: Diagrama general de antecedentes de metodologías para la reducción de características en conjuntos de datos de microarreglos. La línea roja representa las categorías de las dos metodologías utilizadas en esta tesis: una de filtro estadístico (KW) y otra envolvente multiobjetivo basada en neuroevolución (SMS-MONEAT).

3.2. Algoritmos multi-objetivo basados en NEAT

Se han propuesto diferentes algoritmos multiobjetivo, los cuales han tomado como base para su diseño a NEAT o componentes de este. A continuación, describimos algunos de ellos:

- NEAT Pareto-Strengh (NEAT-PS) utiliza la métrica de "fuerza de Pareto" para evaluar las soluciones[82], convirtiendo el problema multiobjetivo en uno de un objetivo y utilizando NEAT para optimizarlo. Este se basa en otro algoritmo evolutivo multiobjetivo popular llamado "Strengh Pareto Evolutionary Algorithm 2" (SPEA2) [83].
- NEAT Multi-Objective Diversified Species (NEAT-MODS) es una mejora al algoritmo de NEAT-PS [84], agregando un método para mantener diversidad en las soluciones. Dicho método consistía en unir la población de padres e hijos y ordenarlos como en la metodología de NSGA-II, después se seleccionan las especies cuyas soluciones se encuentren dentro de los K mejores individuos y al final se toman μ individuos de manera equitativa de las especies seleccionadas.
- Modular Multi-Objective NEAT (MM-NEAT) es un algoritmo multiobjetivo para generar ANNs modulares [85], el cuál tomó como base el algoritmo NSGA-II y utilizó la codificación genética y operadores evolutivos de NEAT.
- Multi-Objective NEAT (mNEAT) toma como base NEAT y utiliza el indicador R2 durante el proceso de especiación. La idea es evaluar primero todas las soluciones mediante el R2 y ordenarlas de acuerdo con su valor de forma descendente, después durante el proceso de especiación el primer individuo a cada especie se convierte en su representante. Después, se utiliza el valor de R2 para calcular la aptitud compartida de cada especie y con ello la probabilidad de reproducción de cada especie. Además, incluye un archivo externo para mantener las mejores soluciones durante la ejecución. En una versión modificada presentada en [86], se agrega un nuevo objetivo a la aptitud, como por ejemplo el tamaño de la especie a la que pertenece antes de evaluar y obtener la aptitud compartida de cada especie. De este modo, se penalizan especies con mayor cantidad de individuos y se incrementa la diversidad de las soluciones.
- Multi-Objective NEAT-Indicator Based (mNEAT-IB) junta el marco de trabajo de SMS-EMOA y R2-EMOA (utiliza el indicador R2 en lugar del hipervolumen) junto con la codificación genética y operadores evolutivos de NEAT [87]. Para la reducción de población el algoritmo puede utilizar 1 o 2 métricas entre "non-dominated sorting", el hipervolumen y el indicador R2. Una diferencia importante con SMS-EMOA es que aquí se genera una población descendencia de tamaño λ la cual se combina la población de padres antes del proceso de reducción, mientras que en SMS-EMOA únicamente se genera un hijo principalmente por alto costo computacional de

utilizar el hipervolumen. Pruebas con distintos indicadores mostraron un mejor desempeño utilizando únicamente el indicador R2 en este algoritmo.

Künzel y Meyer-Nieberg [86] compararon los algoritmos mencionados anteriormente en el problema de balanceo de polea doble, mostrando un mejor desempeño de los algoritmos mNEAT y mNEAT-IB sobre el resto tanto en calidad de las soluciones como en diversificación de estas, en especial la versión de mNEAT-IB utilizando R2 como indicador.

El algoritmo propuesto en este proyecto, llamado SMS-MONEAT, también toma el marco de trabajo de SMS-EMOA, manteniendo el esquema $\mu + 1$ (generar un hijo en cada generación) y el procedimiento para reducir la población basado en el hipervolumen. Además, se incorpora la codificación genética de NEAT, así como sus operadores evolutivos y los agregados en N3O. Adicional a esto, se diseñó un método de especiación mediante un archivo externo para incrementar la diversidad de combinaciones de genes seleccionados para las soluciones obtenidas. El diseño de esta metodología fue presentado en el GECCO 2022 [88].

Capítulo 4

Metodología

La metodología propuesta para realizar la selección de características y generar modelos de clasificación basados en ANNs se describe en la Figura 4.1. El primer paso es dividir el conjunto de datos en dos partes: una para construir un conjunto de soluciones (conjunto de entrenamiento) y otra para seleccionar una de las soluciones (conjunto de validación) buscando aquella que generalice mejor ante nuevos datos. Para el conjunto de entrenamiento el primer paso es filtrar características mediante una prueba estadística y después escalando los datos de las características seleccionadas. Después, el algoritmo SMS-MONEAT es ejecutado para generar ANNs con diferentes características seleccionadas y desempeño de clasificación respecto al conjunto de entrenamiento. Un archivo externo es generado mientras SMS-MONEAT es ejecutado cuyo objetivo es mantener diversidad sobre las características seleccionadas por las soluciones. Finalmente, se utiliza el conjunto de validación tomando las características seleccionadas por la prueba estadística y escalando los datos con los parámetros obtenidos en el proceso análogo para evaluar las soluciones y seleccionar la que mejor desempeño tenga.

En las siguientes secciones describiremos a detalle cada uno de los pasos de esta metodología.

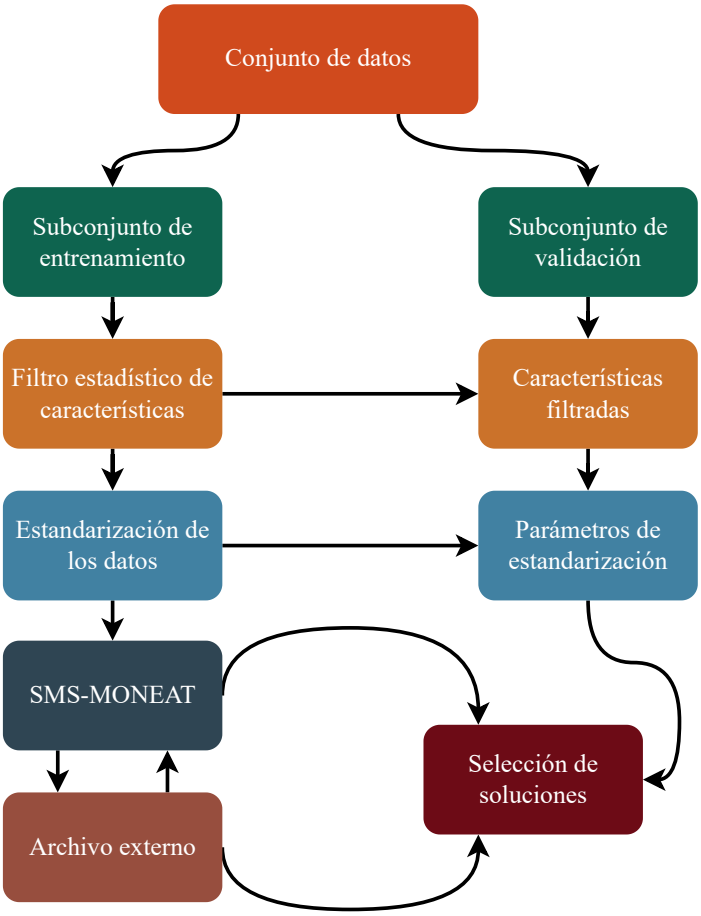


Figura 4.1: Diagrama de la metodología propuesta para la selección de características y entrenamiento de los modelos de clasificación.

4.1. Preprocesamiento del conjunto de datos

El preprocesamiento del conjunto de datos consistió en dos pasos: primero se utilizó un método de selección de características basado en la prueba de KW y después se realizó un escalamiento de los datos utilizando la función *MinMaxScaler* de *Scikit-Learn* [89].

4.1.1. Filtrado estadístico de características

Como primer paso para reducir las dimensiones del conjunto de datos, se utilizó la prueba estadística de KW en cada característica por separada y se utilizó el valor p para estimar la relevancia de cada característica y mantener aquellas con un valor menor a 0.01. En caso de que todas las características sean descartadas, se prueba con un umbral de 0.02, luego 0.03 y así sucesivamente hasta un valor definido por el usuario. El Algoritmo 8 describe el proceso para filtrar las características. Cabe resaltar, que para los conjuntos de datos utilizados en los experimentos el umbral de 0.01 fue suficiente.

4.1.2. Escalamiento de los datos

El objetivo del escalamiento *MinMax* de *Scikit-Learn* es remover la media y escalar a varianza unitaria para cada una de las características, de este modo, las características obtienen un peso similar para la clasificación. Los datos se escalan entre un valor mínimo y máximo definidos, siendo 0 y 1 los valores por defecto respectivamente. Esta función se describe en las Ecuaciones 4.1a y 4.1b, en donde x_{ij} representa el valor de la muestra i y la característica j y se quiere llegar al valor normalizado z_{ij}. Primero en 4.1a se estandariza el valor utilizando el mínimo y máximo de entre todos los valores de X_j (la columna j del conjunto de datos X). Después en 4.1b se escala utilizando un valor α y β que se refieren al mínimo y máximo respectivamente. En nuestro caso se mantuvieron los valores por defecto $\alpha = 0$ y $\beta = 1$.

$$x'_{ij} = \frac{x_{ij} - min(X_j)}{max(X_j) - min(X_j)} \tag{4.1a}$$

$$z_{ij} = \alpha + x'_{ij}(\beta - \alpha) \tag{4.1b}$$

Algoritmo 8 Filtrado estadístico de características (X, y, I)

```
Q ← ∅                                   ▷ Q es el conjunto de características seleccionadas
i ← 1                                   ▷ Número de intento
k ← 0,01
mientras Q = ∅1 o i >= I hacer          ▷ Mientras Q esté vacío o el límite de intentos se haya alcanzado
    para n ← 1 hasta N hacer            ▷ donde N es el número de características
        R_{q∈y} ← ∅                     ▷ se inicializan los conjuntos para cada clase q
        para m ← 1 hasta M hacer        ▷ donde M es el número de muestras
            q ← y_m
            R_q ← X_mn
        p ← KW(R)
        si p < k entonces
            Q ← Q ∪ n
    i ← i + 1
    k ← 0,01 * i
devolver Q
```

4.2. Función de aptitud

La función de aptitud utilizada contiene dos términos: el primero es la función de pérdida y el segundo la cantidad de características seleccionadas. Para la función de pérdida se utilizó la ecuación propuesta para el algoritmo N3O descrita en la Ecuación 2.5, y se decidió utilizar debido a que fue diseñada tomando en cuenta el desbalance de clases común en los microarreglos e incluye el término de regularización L2 para prevenir el sobreajuste de los modelos. La Ecuación 4.2 describe la función de aptitud siendo $g(a, y)$ el valor de entropía cruzada de los valores predichos a y los reales y, y f_s la cantidad de características seleccionadas. El valor de α es una constante para escalar el número de características seleccionadas, ya que la función de pérdida de entropía cruzada binaria tiende a llegar a valores muy pequeños (entre 0 y 1 normalmente) lo cual sería opacado por el segundo objetivo al calcular la contribución del hipervolumen (tomando en cuenta que es parte del proceso para reducir la población de SMS-MONEAT), ya que una sola característica más o menos definiría aquel de mayor contribución sin tomar tanto en cuenta al otro objetivo. Para el escalamiento del segundo término se definió un valor constante de $\alpha = 0{,}1$.

$$f(a, y) = [g(a, y), \alpha f_s] \tag{4.2}$$

4.3. SMS-MONEAT

4.3.1. Descripción general del algoritmo

El nuevo algoritmo toma como base al algoritmo multiobjetivo SMS-EMOA, el cual busca maximizar el hipervolumen del frente de Pareto de las soluciones generadas, e incluye la codificación genética y operadores evolutivos de N3O. El Algoritmo 9 describe el procedimiento principal de SMS-MONEAT. Donde P_t es la población en la iteración t, Q el archivo externo y y es el individuo generado en cada iteración. En comparación al procedimiento de SMS-EMOA, se generan individuos en un esquema $\mu+1$ donde μ es el tamaño de la población P y se genera un hijo en cada iteración. Por otro lado, SMS-MONEAT incluye un archivo externo con una metodología de especiación que se explicará más adelante (Sección 4.4). Además, hay una nueva función para agregar el nuevo individuo x a la población p (Subsección 4.3.4) y la función para reducir también fue modificada (Subsección 4.3.5).

4.3.2. Inicialización de la población

Cada individuo de la población se crea seleccionando una característica como nodo de entrada conectado al nodo de salida. A la conexión se le asigna un valor aleatorio entre límites definidos por el usuario.

También se crea un genoma global, el cual contiene todas las conexiones generadas para mantener un registro histórico y de este modo, mantener el mismo número de innovación en conexiones entre los mismos nodos y así coincidan durante el proceso de cruza.

Algoritmo 9 SMS-MONEAT

```
1:  P_0 ← inicializar()
2:  Q ← ∅
3:  para cada p ∈ P_0 hacer
4:      Q ← agregar_a_archivo(Q, p)
5:  ordenamiento_de_no_dominados(Q)
6:  t ← 0
7:  mientras t ≠ t_max hacer        ▷ Desde t = 0 hasta el máximo número de
    iteraciones t_max
8:      y ← generar_idividuo(P_t)
9:      P_{t+1} ← agregar(P_t, y)
10:     P_{t+1} ← reducir(P_{t+1})
11:     Q ← agregar_a_archivo(Q, y)
12:     t ← t + 1
```

Algoritmo 10 SMS-MONEAT: generar_individuo(P)

```
1: Calcular probabilidad de selección de P.
2: r ← random(0, 1)                          ▷ r es un valor aleatorio entre 0 y 1
3: p1, p2 ← seleccion_de_padres(P)
4: si r < μc entonces
5:     k ← kmax
6:     mientras k > 0 hacer
7:         q ← cruza(p1, p2)
8:         si q es una red válida entonces
9:             q ← mutación(q)
10:            Romper ciclo
11:        k ← k − 1
12: si no
13:    q ← mutación(p1)
14: qaptitud ← evaluar(q)          ▷ Se evalúa con el conjunto de entrenamiento
15: devolver q
```

4.3.3. Procedimiento para generar un nuevo individuo

El Algoritmo 10 muestra el procedimiento para generar un nuevo individuo a partir de los operadores evolutivos. Primero se calcula la probabilidad de selección de cada individuo, después se genera un valor aleatorio r para comparar con la probabilidad de cruza μ_c. Se seleccionan dos padres p_1 y p_2, y en caso de que se supere la probabilidad de cruza se aplica esta operación entre ambos padres para generar a un nuevo individuo q. Se realiza un proceso de validación del individuo generado, es decir, que no se haya generado ningún ciclo entre las conexiones o asegurarse de que exista un camino habilitado entre un nodo de entrada y uno de salida. En caso de que el individuo sea inválido se intenta de nuevo el proceso de mutación, puesto que puede ser estocástico el resultado puede variar. El número de intentos está definido a 5, y en caso de no encontrar una solución válida la aptitud se ajusta a infinito para que esta sea descartada (ver Subsección 4.3.6). Si el hijo generado es una ANN válida, se realiza el proceso de mutación. Por otro lado, si $r >= \mu_c$ entonces q es simplemente uno de los padres después del proceso de mutación. No está expresado explícitamente en el pseudocódigo, pero también hay una probabilidad de mutación y es independiente para cada uno de los operadores de mutación utilizados. Al final se evalúa el individuo generado, primero transformando la codificación genética a una ANN, después evaluando con el conjunto de entrenamiento y utilizando la función de aptitud descrita en la Ecuación 4.2.

Selección de padres

La selección de padres definida fue probabilista, por lo que el primer paso es calcular la probabilidad de selección de cada individuo. Para ello se utilizó el rango de dominancia definido por el algoritmo de *ordenamiento_de_no_dominados()*

Algoritmo 11 SMS-MONEAT: Cálculo de probabilidad de selección (P)

1: $\mu \leftarrow \frac{1}{n} \sum_{p \in P} p_{rank}$ ▷ Se calcula el promedio del rango de la población
2: $\beta \leftarrow 1$
3: **si** $\mu \neq 0$ **entonces**
4: $\quad \beta \leftarrow \frac{1}{\mu}$
5: **para cada** $p \in P$ **hacer**
6: $\quad p_{prob} \leftarrow \exp -\beta p_{rank}$

de NSGA-II. El Algoritmo 11 muestra la forma en que se calculó la probabilidad de selección de cada individuo. En este algoritmo, se calcula primero el promedio y se utiliza para normalizar los valores, excepto, en el caso donde el promedio es 0 (que implicaría que todas las soluciones están en el frente del Pareto). Después se utiliza una función exponencial negativa ya que queremos que las soluciones con un menor rango tengan mayor probabilidad de ser seleccionadas.

Después de calcular la probabilidad de selección de cada individuo, se utilizó una selección por torneo binario, en donde dos individuos son elegidos al azar y se selecciona uno aleatoriamente según su probabilidad de selección. Se repite este proceso dos veces para obtener dos padres.

Operadores de cruza y mutación

El operador de cruza que se utilizó fue el descrito en N3O (Subsección 2.6.1). El único detalle para resaltar es que para determinar al padre con mejor aptitud se utilizó el ranking de dominancia.

Del mismo modo los operadores de mutación utilizados fueron los descritos en N3O, que consisten en:

- Mutación de pesos
  - Un 90 % de probabilidad de sumar una perturbación mediante la mutación polinomial (Ecuación 4.3).
  - Un 10 % de probabilidad de sustituir por un valor nuevo aleatorio.
- Mutación estructural
  - Agregar un nuevo nodo oculto entre dos nodos existentes (deshabilitando una conexión existente y generando dos nuevas conexiones).
  - Agregar una nueva conexión entre dos nodos existentes.

- Agregar un nodo de entrada utilizando el valor p de la prueba de KW para definir la probabilidad de selección del nuevo nodo.

- Sustituir un nodo de entrada existente por uno nuevo.

Para la mutación de pesos se utilizó el operador de mutación polinomial, descrito en la Ecuación 4.3, donde r es un número generado aleatoriamente entre 0 y 1 y η define la variabilidad de la perturbación [90].

$$f(x) = \begin{cases} x + \left[2r^{\frac{1}{\eta+1}} - 1\right] & \text{si } r < 1 \\ x + \left[1 - 2(1-r)^{\frac{1}{\eta+1}}\right] & \text{en caso contrario} \end{cases} \tag{4.3}$$

4.3.4. Procedimiento para agregar un nuevo individuo a la población

El algoritmo de ordenamiento de soluciones no-dominadas de NSGA-II tiene una complejidad $O(n^2)$ (Algoritmo 5), donde n es el tamaño de la población. Este algoritmo consiste en 2 etapas, la primera es contar para cada solución el número de soluciones que la dominan (n) y encontrar al subconjunto de soluciones dominadas (S), y la segunda consiste en generar los frentes, ambas con la misma complejidad computacional. Teniendo en cuenta que SMS-MONEAT sólo agrega un individuo y elimina otra en cada iteración, se puede utilizar la información de la iteración anterior para actualizar los valores de n y S. Es por ello, que para agregar un nuevo individuo el procedimiento es actualizar dichos valores para toda la población, incluyendo el nuevo individuo. El Algoritmo 12 describe este procedimiento. Donde P es la población y y es el nuevo individuo por agregar, y_S y y_n son el conjunto de soluciones dominadas y el número de veces que y es dominada respectivamente. El nuevo individuo se compara con cada uno de los individuos de la población y en caso de que domine o sea dominado se actualizan los respectivos valores, de este modo, (al menos) está parte del algoritmo se ejecuta de manera lineal.

4.3.5. Procedimiento para reducir la población

Para reducir la población se utiliza la contribución del hipervolumen como lo hace SMS-EMOA en el Algoritmo 7. Sin embargo, al ya tener la información de relación de dominancia de cada individuo, sólo falta generar nuevamente los frentes lo cual se describe en el Algoritmo 13. Un punto importante en este paso es el generar una variable temporal con la cuenta de individuos que dominan a cada solución, para no perder esta información y pueda ser utilizada al agregar un nuevo individuo en la población.

Con los frentes generados, se puede eliminar a un individuo del último frente, si dicho frente sólo contiene un individuo se elimina y en caso de que esté

Algoritmo 12 SMS-MONEAT: agregar(P, y)

```
1:  y_S ← ∅
2:  y_n ← 0
3:  para cada p ∈ P hacer
4:      si y ≺ p entonces                          ▷ Si y domina a p
5:          y_S ← y_S ∪ {p}                        ▷ Se agrega p a y_S
6:          p_n ← p_n + 1                          ▷ Se incrementa p_n
7:      si no si p ≺ y entonces                    ▷ Si p domina a y
8:          p_S ← p_S ∪ {y}                        ▷ Se agrega y a p_S
9:          y_n ← y_n + 1                          ▷ Se incrementa y_n
10: si y_n = 0 entonces
11:     y_rank ← 0
12: devolver P ∪ {y}
```

formado por más de un individuo se procede a buscar aquel cuya contribución al hipervolumen sea menor. Antes de realizar el cálculo del hipervolumen se revisa si hay elementos con la misma aptitud dentro de este frente, puesto que su aportación será nula. Si este es el caso, entonces se calcula el hipervolumen sobre los valores repetidos, y del que menor aportación tenga, entonces se elige una de las soluciones con dicho valor de manera aleatoria. En el caso de que todas las soluciones tengan un valor de aptitud diferente, entonces se busca el de menor aportación de todo el frente. Finalmente, se elimina un individuo y se actualizan la relación de dominancia entre el resto de la población lo cual se describe en el Algoritmo 14. El Algoritmo 15 muestra el procedimiento completo para reducir a la población.

Para el cálculo del hipervolumen se utilizó la biblioteca de *Pygmo* [91]. Tomando como referencia el valor máximo para cada objetivo dentro de las soluciones del frente de Pareto (conocido como punto de Nadir) más un valor $\delta = 0{,}1$ (en cada objetivo).

4.3.6. Soluciones inválidas

Las soluciones generadas por el algoritmo son redes neuronales prealimentadas, lo que implica que no tiene ciclos en ellas. En la manera en que se implementó el algoritmo, durante el operador de cruza se podrían generar redes que contenga ciclos o redes que no conecten alguna de las entradas con alguna de las salidas. Esto puede ocurrir por dos razones: la primera es la capacidad de habilitar y deshabilitar conexiones, puesto que podría haber conexiones que habilitadas generen un ciclo y deshabilitadas no, o se podría deshabilitar conexiones e impedir el flujo de datos entre las entradas y las salidas; la segunda es que si ambos padres tienen la misma aptitud, las conexiones excedentes y disyuntivas se agregan de manera aleatoria, y podrían agregarse conexiones que generen un ciclo o podrían no agregarse conexiones y que se pierda el camino entre las entradas y las salidas.

Algoritmo 13 SMS-MONEAT: construir_frentes(P)

```
1: i ← 0
2: F_i ← ∅
3: para cada p ∈ P hacer
4:     p_nt ← p_n
5:     si p_n = 0 entonces
6:         F_i ← F_i ∪ {p}
7: mientras F_i ≠ ∅ hacer
8:     Q ← ∅
9:     para cada p ∈ F_i hacer
10:        para cada q ∈ p_S hacer
11:            q_nt ← q_nt − 1
12:            si q_nt = 0 entonces                ▷ Si q_n = 0
13:                Q ← Q ∪ {q}                     ▷ q pertenece al siguiente frente
14:                q_rank ← i + 1
15:    i ← i + 1
16:    F_i ← Q
17: devolver F
```

Algoritmo 14 SMS-MONEAT: Eliminar(P, x)

```
1: para cada p ∈ P hacer
2:     si x ≺ p entonces                           ▷ Si x domina a p
3:         p_n ← p_n − 1                           ▷ Se decrementa p_n
4:     si no si p ≺ x entonces                     ▷ Si p domina a x
5:         p_S ← p_S\{x}                           ▷ Se elimina x a p_S
6: devolver (P\{x})
```

Cuando una solución inválida es generada, entonces la aptitud de la solución se iguala a infinito para ambos términos, de este modo, la solución será automáticamente descartada al reducir la población.

4.3.7. Descripción de parámetros del algoritmo

El funcionamiento de SMS-MONEAT depende de distintos hiperparámetros. La Tabla 4.1 describe cada uno de los hiperparámetros del algoritmo. De manera general, el algoritmo tiene un tamaño de población μ y genera un individuo en cada generación, hasta I generaciones. También hay una probabilidad para que se ejecuten los operadores de cruza y mutación en cada generación. Además, algunos de los operadores evolutivos dependen de otros parámetros, como el operador de cruza tiene probabilidades para agregar nuevos individuos, deshabilitar conexiones o agregar conexiones que no coincidieron entre ambos padres cuando tienen la misma aptitud. También se define w_1 y w_2 que definen el rango con el cuál se inicializan los pesos de las conexiones al inicializar la

Algoritmo 15 SMS-MONEAT: Reducir(P)

1: $\{\mathscr{R}_1, ..., \mathscr{R}_v\} \leftarrow$ construir_frentes(P)
2: **si** $\mathscr{R}_v$ contiene 1 individuo **entonces**
3: $r \leftarrow \mathscr{R}_v$ ▷ Se escoge el único individuo de $\mathscr{R}_v$ **si no, si** hay valores de aptitud repetida en $\mathscr{R}_v$
4: $Rep \leftarrow$ repetidos($\mathscr{R}_v$)
5: $a \leftarrow min_{s \in Rep}[\Delta_{\mathscr{S}}(s, Rep)]$ ▷ Se escoge el individuo con menor contribución al hipervolumen
6: $A \leftarrow \{x \in \mathscr{R}_v : x_{apitutd} = a_{aptitud}\}$
7: $r \leftarrow$ elección_aleatoria(A)
8: **si no**
9: $r \leftarrow min_{s \in \mathscr{R}_v}[\Delta_{\mathscr{S}}(s, \mathscr{R})]$ ▷ Se escoge el individuo con menor contribución al hipervolumen
10: Eliminar(P, r) ▷ Se actualizan los valores de relación de dominancia
11: **devolver** ($P \backslash \{r\}$)

población o al agregar una nueva conexión. Otro parámetro que se define es el de regularización que se aplica en la función de aptitud.

La Tabla 4.2 muestra otro conjunto de parámetros, los cuáles definen el funcionamiento de las ANNs generadas, los cuáles son la función de agregación y las de activación. La de agregación define a la función matemática que se aplica a las entradas de cada nodo, por ejemplo, se pueden simplemente sumar, lo cual es lo más común, sin embargo, en el diseño de N3O, se utilizó el promedio. Las funciones de activación son aquellas que se ejecutan entre cada una de las capas y crean no linealidad en el modelo y se definió una para los nodos ocultos y otra para el nodo de salida.

Tabla 4.1: Hiperparámetros de SMS-MONEAT.

Nomenclatura	Dominio	Descripción
μ	$\{\mathbb{N}\}$	Tamaño de población
I	$\{\mathbb{N}\}$	Número de generaciones
P_a	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de agregar una entrada del padre menos apto y no existente en el nuevo individuo.
P_b	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de agregar una conexión excedente o disyuntiva cuando los padres tengan la misma aptitud.
P_c	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de que el operador de cruza se ejecute al generar un nuevo individuo.
P_d	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de que una conexión del nuevo individuo se deshabilite si está deshabilitado en alguno de los padres.
P_α	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de agregar una nueva entrada al nuevo individuo.
P_β	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de sustituir una de las entradas al nuevo individuo.
P_γ	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de agregar una nueva conexión al nuevo individuo.
P_δ	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de agregar un nuevo nodo al nuevo individuo.
P_ω	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de que el operador de mutación de pesos de una conexión se ejecute.
P_σ	$\{\mathbb{R}\|[0,1]\}$	Probabilidad de sustituir el peso de una conexión por un valor aleatorio en lugar de aplicar una mutación polinomial.
η	$\{\mathbb{R}\|[0,\text{ínf})\}$	Variabilidad de perturbación de la mutación polinomial.
$[w_1, w_2]$	$\{\mathbb{R}\}$	Límite inferior y superior para inicializar el valor de pesos de las conexiones.
λ	$\{\mathbb{R}\|[0,\text{ínf})\}$	Parámetro de regularización.

Tabla 4.2: Parámetros de las ANNs generadas por SMS-MONEAT.

Nomenclatura	Descripción
$f_{agregaciaón}$	Función de agregación
$f_{activación_1}$	Función de activación para nodos ocultos
$f_{activación_2}$	Función de activación para nodos de salida

4.4. Archivo externo

Uno de los problemas principales en conjuntos de datos desbalanceados y con pocas muestras, como es el caso de los microarreglos, es el sobreajuste de los modelos de clasificación. Para algoritmos basados en una población de soluciones, lo anterior implica que la solución con mejor desempeño ante el conjunto de entrenamiento podría no ser la que tenga el mejor desempeño ante el conjunto de prueba. Esta es la razón por la cuál en esta metodología se decidió utilizar un conjunto de validación para seleccionar una solución de la población final. Sin embargo, es relevante asegurarse de mantener la diversidad en las soluciones y de este modo, mantener soluciones que generalicen mejor ante nuevos datos. Por ello, se incluyó un archivo externo con una metodología de especiación basada en la combinación de características seleccionadas. El tamaño máximo de este archivo externo es igual al del tamaño de la población y con un valor máximo de soluciones por especie definido para que, de este modo, una sola especie con buen desempeño no opaque por completo a las demás. El Algoritmo 16 muestra el procedimiento para agregar un nuevo individuo al archivo, mientras que el Algoritmo 17 describe la manera en que se elimina un individuo de la población.

Para agregar un nuevo individuo al archivo, primero se revisa si este tiene los mismos genes seleccionados que alguna especie existente dentro del archivo. Si este es el caso, entonces se agrega a esta especie y se revisa si no ha superado el valor máximo definido para el tamaño de una especie. Si una especie sobrepasa el número de individuos máximo se descarta el que tenga la menor aptitud (el valor de la función de pérdida). En caso de que no se encuentre una especie con los mismos genes, se crea una nueva especie con el nuevo individuo. Al final se revisa que el archivo no haya excedido su tamaño máximo, y si este es el caso, se reduce el archivo. Para reducir el archivo, lo primero es tomar a las soluciones con menor aptitud de cada especie y se almacena en V, después se generan frentes de Pareto y dependiendo de la cantidad de elementos del último frente ($\mathscr{R}_v$) se realizan las siguientes opciones:

- Si $\mathscr{R}_v$ tiene un elemento este es el que se elimina de la población.
- Si $\mathscr{R}_v$ tiene dos elementos se elimina el que peor función de pérdida tenga.
- Si $\mathscr{R}_v$ tiene más de dos elementos, pero hay elementos repetidos, se busca el que menor contribución de hipervolumen tenga (de los valores únicos) y de este, se elige aleatoriamente cuál eliminar.
- Si las anteriores no se cumplen, se elimina el que menor contribución de hipervolumen de $\mathscr{R}_v$.

Algoritmo 16 Archivo externo: agregar_a_archivo(Q, x)

agregado ← **Falso**
para cada $S \in Q$ **hacer** ▷ Para cada especie S en el archivo Q
    si $S_{genes} = x_{genes}$ **entonces** ▷ Se comparan los genes seleccionados entre la especie s y el nuevo individuo x
        $S \leftarrow S \cup \{x\}$
        si $S_{tamaño} >$ máximo_especie **entonces** ▷ Se revisa si el tamaño de la especie no haya superado el máximo definido
            $z \leftarrow \text{mín}_{s \in S}[s_{aptitud}]$ ▷ Se obtiene el valor con menor aptitud (función de pérdida)
            $s \leftarrow (s \backslash \{z\})$
            devolver Q
        agregado ← **Verdadero**
        Romper ciclo
si agregado = **Falso entonces**
    $s_{nueva} \leftarrow \{x\}$
    $Q \leftarrow Q \cup s_{nueva}$
si $Q_{tamaño} >$ máximo_archivo **entonces** ▷ Se revisa si el tamaño del archivo no haya superado el máximo definido
    $Q \leftarrow$ reducir_archivo(Q)
devolver Q

Algoritmo 17 Archivo externo: reducir_archivo(Q)

$V \leftarrow \emptyset$ ▷ Conjunto para almacenar las soluciones con menor aptitud de cada especie
para cada $S \in V$ **hacer** ▷ Para cada especie s en el archivo V
    $z \leftarrow \text{mín}_{s \in S}[s_{aptitud}]$ ▷ Se obtiene el valor con menor aptitud (función de pérdida)
    $P \leftarrow P \cup \{z\}$
$\{\mathscr{R}_1, ..., \mathscr{R}_v\} \leftarrow$ ordenamiento_de_no_dominados(V)
si $\mathscr{R}_v$ tiene un elemento **entonces**
    $r \leftarrow \mathscr{R}_v$ **si no, si** $\mathscr{R}_v$ tiene 2 elementos
    $r \leftarrow \text{mín}_{s \in \mathscr{R}_v}[s_{aptitud}]$ **si no, si** hay valores de aptitud repetida en $\mathscr{R}_v$
    $Rep \leftarrow$ repetidos($\mathscr{R}_v$)
    $a \leftarrow min_{s \in Rep}[\Delta_{\mathscr{S}}(s, Rep)]$ ▷ Se escoge el individuo con menor contribución al hipervolumen
    $A \leftarrow \{x \in \mathscr{R}_v : x_{apitutd} = a_{aptitud}\}$
    $r \leftarrow$ elección_aleatoria(A)
si no
    $r \leftarrow min_{s \in \mathscr{R}_v}[\Delta_{\mathscr{S}}(S, \mathscr{R})]$
devolver $(Q \backslash \{r\})$

4.5. Selección de soluciones

Al final de la ejecución de SMS-MONEAT se obtendrá una población de soluciones de las cuáles se deberá elegir una de ellas. Para ello se utilizó un subconjunto de validación, para evitar utilizar una solución que se haya sobreajustado al conjunto de entrenamiento. Como se muestra en la Figura 4.1, primero se reduce el conjunto de validación utilizando las características seleccionadas por el filtro estadístico y se escala utilizando los parámetros obtenidos durante el proceso realizado al conjunto de entrenamiento y después se utiliza para la selección de soluciones.

Para cada solución se calculó el valor de la función de pérdida (Ecuación 4.2) y el promedio geométrico (Ecuación 4.4) tanto en el conjunto de entrenamiento como en el de validación, se realizó una suma ponderada de estos valores y se eligió el de menor valor. Se utilizó el promedio geométrico ya que esta métrica permite observar si una solución tiene sesgo a la clase con mayor proporción del conjunto de datos y el cuál se calcula como [92]:

$$PGeo = \sqrt{VP_{tasa} \cdot VN_{tasa}}, \tag{4.4}$$

donde VP_{tasa} es la tasa de verdaderos positivos mientras que VN_{tasa} es la tasa de verdaderos negativos. El cálculo de cada uno de estos términos se expresa en las Ecuaciones 4.5 y 4.6, donde VP es la cantidad de verdaderos positivos, VN la de verdaderos negativos, FP la de falsos positivos y FN la de falsos negativos.

$$VP_{tasa} = \frac{VP}{VP + FN} \tag{4.5}$$

$$VN_{tasa} = \frac{VN}{VN + FP} \tag{4.6}$$

La Ecuación 4.7 describe la suma ponderada utilizada para seleccionar una solución en la cual a y y representan el valor predicho y el real, g es la función de pérdida y los subíndices e y v representan al conjunto de entrenamiento y validación respectivamente. w_1 y w_2 son los pesos, de los que se eligió un valor de $w_1 = 0{,}35$ y $w_2 = -0{,}15$, dando un mayor peso a la función de pérdida sobre el promedio geométrico. El valor del promedio geométrico va de 0 a 1, siendo 1 donde el modelo acierta en todo y 0 donde no se acierte en ninguno de los positivos o de los negativos. Como se busca minimizar la suma ponderada w_2 tiene un valor negativo. Al incluir el promedio geométrico se penaliza una solución si se tiene un valor cercano a 0 en el conjunto de entrenamiento o en el de validación, que implicaría una mala generalización.

$$S = w_1 g(a_e, y_e) + w_2 PGeo_e + w_1 g(a_v, y_v) + w_2 PGeo_v \tag{4.7}$$

Esta metodología fue aplicada durante los experimentos de N3O y SMS-MONEAT, de este último, se aplicó tanto a la población final como a la población del archivo externo.

4.6. Implementación

El código fuente desarrollado para este proyecto, tanto los algoritmos como los experimentos realizados se encuentra en GitHub en la siguiente liga: `https://github.com/dan-gn/SMS-MONEAT`. La implementación fue hecha en Python3 y los algoritmos de NEAT, N3O, SMS-EMOA y SMS-MONEAT fueron implementados de manera propia. Fueron utilizadas bibliotecas comunes como NumPy, PyTorch (para las ANN) y PyGMO (para el hipervolumen).

Capítulo 5

Configuración experimental

Los experimentos realizados se dividieron en dos etapas: la primera consistió en optimizar los hiperparámetros de SMS-MONEAT y N3O relacionados a la probabilidad de ejecución de los operadores evolutivos utilizando iRace y en la segunda etapa se compararon ambos algoritmos en múltiples conjuntos de datos de microarreglos para comparar su desempeño en las tareas de selección de genes y generación de modelos de clasificación. También se comparó la metodología propuesta con una metodología envolvente multiobjetivo utilizando SMS-EMOA y el clasificador kNN. Se eligió utilizar el SMS-EMOA ya que, al ser base de SMS-MONEAT, ambos algoritmos buscan optimizar el hipervolumen y de este modo, esta métrica podría usarse para comparar el desempeño de ambas metodologías.

5.1. Optimización de hiperparámetros para los operadores evolutivos de SMS-MONEAT y N3O

Para este proceso se utilizó iRace, el cual es una herramienta de configuración automática de algoritmos que optimiza ciertos parámetros de acuerdo con instancias definidas. Estas instancias, para el problema en cuestión, implicarían ser conjuntos de datos de microarreglos. Los conjuntos de datos utilizados fueron obtenidos de la base de datos CuMiDa («Curated Microarray Database») [93], la cual contiene conjuntos de datos de microarreglos de diversos tipos de cáncer. La Tabla 5.1 muestra los conjuntos de datos que fueron utilizados en este proceso, siendo 5 conjuntos de datos de 5 diferentes tipos de cáncer y se presenta la cantidad de muestras, características y distribución de las clases.

Para optimizar los hiperparámetros de SMS-MONEAT se utilizó el valor del hipervolumen de la población utilizando la función de pérdida y el número de características como se muestra en la Ecuación 4.2. Debido a que N3O no utiliza el hipervolumen como parte del algoritmo, se decidió el promedio obtenido por la función de pérdida de toda la población. La evaluación de ambos algoritmos

Tabla 5.1: Conjuntos de datos de microarreglos utilizados para la optimización de hiperpárametros.

Conjunto de datos	Tipo de cáncer	Número de muestras	Número de genes	Distribución de clases [%]
GSE8671	Colon	63	54675	51/49
GSE57957	Hígado	75	47323	52/48
GSE71935	Leucemia	46	54675	80/20
GSE42568	Mama	116	54675	87/13
GSE46602	Próstata	49	54675	71/29

Tabla 5.2: Hiperparámetros que fueron optimizados junto con su configuración inicial y rango de búsqueda.

Hiperparámetro	Inicio	Mínimo	Máximo
P_c	0.75	0.50	1.00
P_α	0.05	0.00	0.15
P_β	0.05	0.00	0.15
P_γ	0.05	0.00	0.15
P_δ	0.03	0.00	0.15
P_ω	0.04	0.00	0.15

se hizo con el 30 % de los conjuntos de datos y el resto para como conjunto de entrenamiento.

Para la ejecución de iRace se realizaron 1000 experimentos, con 5 instancias y 6 parámetros a utilizar. La Tabla 5.2 muestra los parámetros que se optimizaron, siendo básicamente las probabilidades de aplicar cada operador de cruza y mutación para cada nuevo individuo, mientras que el resto de hiperparámetros se presentan en la Sección 5.2.1.

5.2. Experimentos realizados para la selección de genes y clasificación de microarreglos.

Se realizaron experimentos para validar el funcionamiento de SMS-MONEAT en conjuntos de datos de microarreglos y comparar su funcionamiento respecto a N3O y SMS-EMOA. Para cada algoritmo se realizaron experimentos en 20 conjuntos de datos de microarreglos de distintos tipos de cáncer: mama, colon, leucemia, próstata e hígado. Con cada conjunto de datos se realizaron 3 repeticiones de validación cruzada a 10 capas con división estratificada para que cada capa mantuviera la proporción de las clases.

La metodología de SMS-MONEAT y N3O fue la misma y se presenta en la Figura 5.1. En este diagrama podemos ver una iteración de la validación

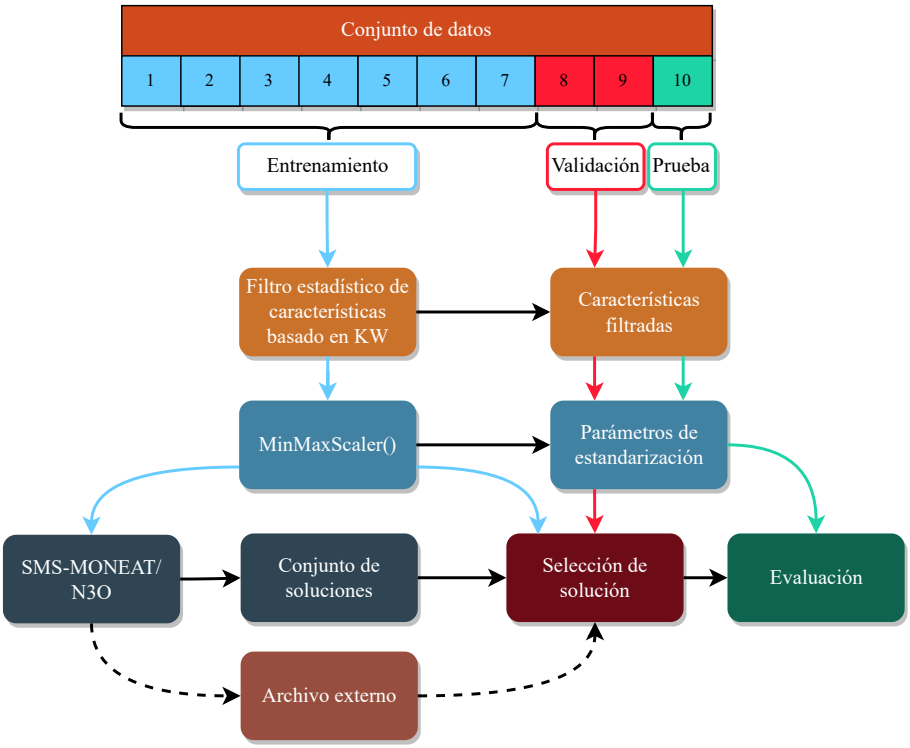


Figura 5.1: Diagrama de una iteración realizada de validación cruzada estratificada a 10 capas de los experimentos realizados a SMS-MONEAT y N3O.

cruzada estratificada en 10 capas utilizada, tomando 7 capas como conjunto de entrenamiento, 2 para validación y 1 para prueba. Primero, el filtro estadístico utiliza el conjunto de entrenamiento para identificar características relevantes y a la par, reducir los otros dos subconjuntos. Después, se ajustan los parámetros de la función *MinMaxScaler* con el conjunto de entrenamiento y se transforman todos los subconjuntos con dichos parámetros. El siguiente paso es la ejecución de SMS-MONEAT o N3O y se genera un conjunto de soluciones. Se procede a seleccionar una de las soluciones utilizando tanto el conjunto de entrenamiento como el de validación. Al final, se evalúa la solución seleccionada utilizando el conjunto de prueba. Adicional a esto, para SMS-MONEAT se selecciona una solución del archivo externo basado en especiación.

Para SMS-EMOA la metodología utilizada se presenta en la Figura 5.2, la cual se diferencia de la anterior debido a que no tiene subconjunto de validación. Esto es porque SMS-EMOA utiliza el modelo de KNN como clasificador y se ejecuta validación cruzada estratificada a 3 capas para evaluar cada modelo.

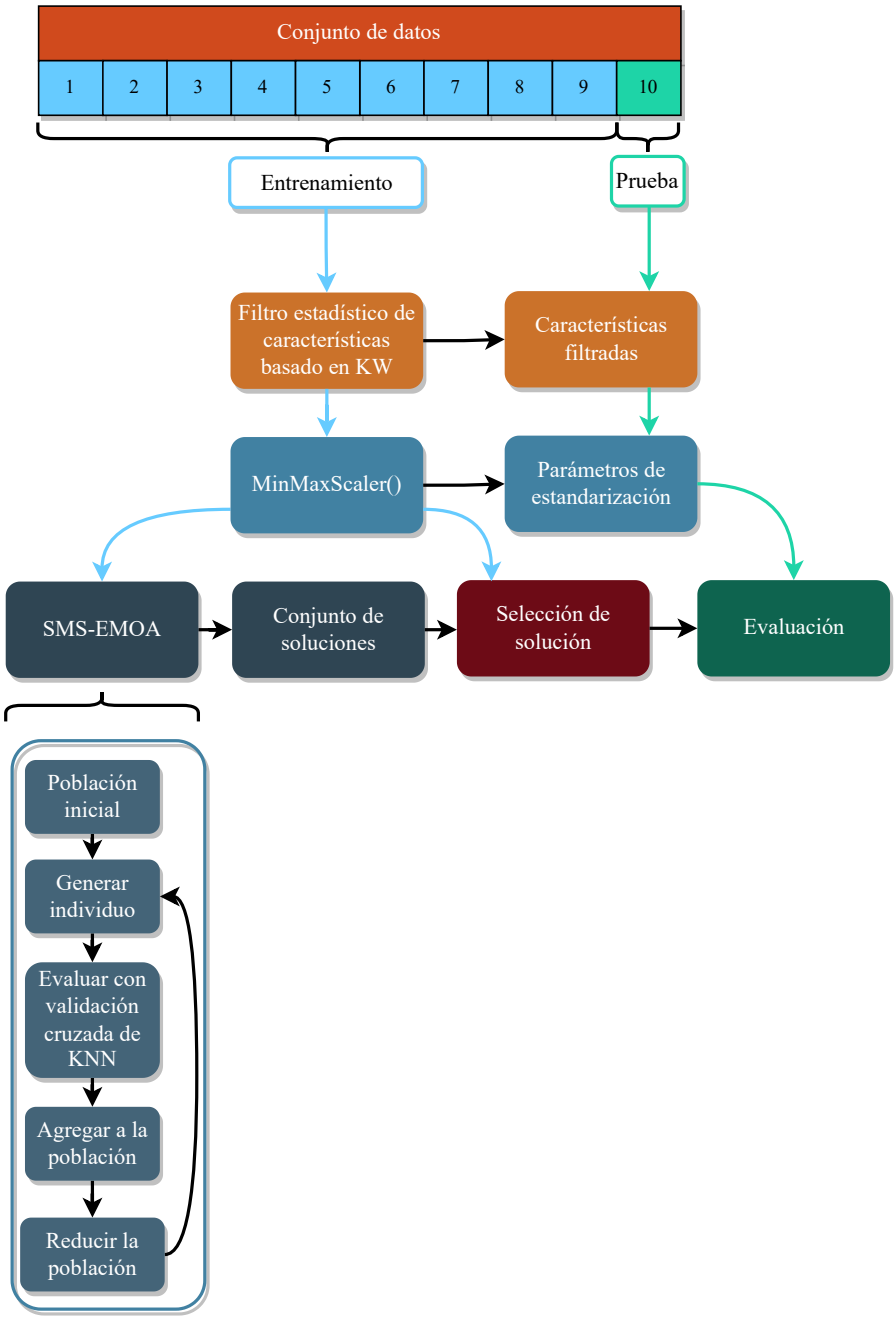


Figura 5.2: Diagrama de una iteración realizada de validación cruzada estratificada a 10 capas de los experimentos realizados a SMS-EMOA.

5.2.1. Hiperparámetros

Los valores de los hiperparámetros utilizados tanto en la configuración automática de los algoritmos como en los experimentos se presentan en las Tablas 5.3 y 5.4. En la Tabla 5.3 se muestran aquellos hiperparámetros que ambos algoritmos comparten, mientras que en la Tabla 5.4 aquellos que sólo aplican para N3O. Los parámetros relacionados con las ANNs y N3O se seleccionaron de acuerdo a los valores utilizados en el diseño original de N3O [18].

En la Tabla 5.4, las variables c_1, c_2 y c_3 hacen referencia a la ecuación de compatibilidad de NEAT para la especiación de las ANNs (Ecuación 2.4), siendo δ_{umbral} el umbral para saber si una solución pertenece a una cierta especie. Además, la variable ϵ sirve para definir la proporción de soluciones elitistas que se mantendrán de la población original en cada generación. Finalmente, el valor $P_{interespecie}$ define la probabilidad de que la cruza se haga entre dos especies distintas.

Para igualar el número de evaluaciones de ambos algoritmos, cada uno tiene un valor diferente de número de generaciones. Tomando en cuenta de que tienen una población de 100, al generar la población inicial ambos algoritmos realizarán 100 evaluaciones. Después en cada generación de SMS-MONEAT se genera un individuo que implica una evaluación, pero N3O hará 90 individuos ya que los otros 10 serán los que se mantienen de la población original por el valor de elitismo $\epsilon = 0{,}1$, es decir, se realizan el mismo número de evaluaciones en 90 generaciones de SMS-MONEAT y en 1 de N3O. Primero se realizaron pruebas con N3O a 100 y 200 generaciones y se notó un mejor desempeño con 200 generaciones, por lo cual este es el valor que se mantuvo para los experimentos. Por lo tanto, el valor fijado para SMS-MONEAT fue de 18000 y siendo un total de 18100 evaluaciones.

Para SMS-EMOA se utilizó una codificación binaria siendo cada gen una característica a seleccionar. Al igual que las otras metodologías se utilizó una población de 100 y el número de evaluaciones fue de 18000, sin embargo, al utilizar validación cruzada de 3 capas el número de generaciones fue de 6000 (siendo 18300 evaluaciones tomando en cuenta la población inicial). Aquí un detalle importante es que la evaluación para SMS-MONEAT implica pasar de la codificación genética a una red neuronal y después evaluar en el conjunto de datos, mientras que para SMS-EMOA es entrenar un modelo de KNN con un conjunto de datos y después predecir los valores de otro conjunto y se está asumiendo que son equivalentes. Se utilizó selección de padres por torneo binario, cruza en un punto y un operador de mutación binario. La probabilidad de mutación y cruza no fueron optimizados por iRace ya que se prefirió ajustar el valor de la probabilidad de mutación de acuerdo a cada experimento. El tamaño del genoma de cada experimento depende de la cantidad de características que se mantengan después del filtro estadístico, por lo que es un valor que varía entre cada conjunto de datos e incluso, entre cada uno de los subconjuntos de entrenamiento durante la validación cruzada. Por lo que elegir un valor constante para la probabilidad de mutación para todos los experimentos podría ocasionar una mayor exploración sobre las características seleccionadas en algunos de ellos

Tabla 5.3: Hiperparámetros para SMS-MONEAT y N3O utilizados durante los experimentos realizados.

Parámetro	Valor
μ	100
I	18000 (SMS-MONEAT), 200 (N3O)
P_a	0.50
P_b	0.80
P_d	0.75
P_σ	0.1
η	5
$[w_1, w_2]$	$[-1, 1]$
λ	0.5
$f_{agregaciaón}$	Promedio
$f_{activación_1}$	$\tanh(2{,}45 * x)$
$f_{activación_2}$	$\exp \frac{-5x}{2}$

Tabla 5.4: Hiperparámetros para N3O utilizados durante los experimentos realizados.

Nomenclatura	Valor
$[c_1, c_2, c_3]$	[1,0, 1,0, 0,4]
δ_{umbral}	3
ϵ	0.1
$P_{interespecie}$	0.001

y casi nula en otros. Por ello, la probabilidad de mutación se definió como el valor inverso a la cantidad de características seleccionadas por el filtro estadístico para cada experimento. En cambio, para el operador de cruza sí se definió un valor constante de 0.75. La Tabla 5.5 muestra los hiperparámetros utilizados para SMS-EMOA, siendo P_c y P_m la probabilidad de cruza y mutación, $genes_{kw}$ la cantidad de genes seleccionados por el filtro estadístico y K la cantidad de capas utilizadas durante la evaluación mediante validación cruzada de kNN.

5.2.2. Conjuntos de datos

La Tabla 5.6 muestra los diferentes conjuntos de datos utilizados durante los experimentos para evaluar el desempeño de SMS-MONEAT. Aquellos que empiezan con el prefijo "GSE" pertenecen a la base de datos de CuMiDa, los demás son conjuntos de datos de referencia. La segunda columna de la tabla describe el tipo de cáncer y un nombre clave que será útil para identificar cada conjunto de datos en las tablas presentadas en el Capítulo 6 donde se presentan los resultados experimentales.

Tabla 5.5: Hiperparámetros para SMS-EMOA utilizados durante los experimentos realizados.

Nomenclatura	Valor
μ	100
I	18000
P_c	0.75
P_m	$(genes_{kw})^{-1}$
K	3

Tabla 5.6: Conjuntos de datos de microarreglos utilizados para evaluar el desempeño de SMS-MONEAT y compararlo contra otras metodologías. La segunda columna describe el tipo de cáncer de cada conjunto de dato y el nombre clave que servirá para identificarlo en futuras tablas.

Conjunto de datos	Nombre clave	Número de muestras	Número de genes	Distribución de clases [%]
GSE25070	Colon_1	52	24526	50/50
GSE32323	Colon_2	33	54675	52/48
GSE44076	Colon_3	194	49386	50/50
GSE44861	Colon_4	105	22277	50/50
GSE14520_U133A	Hígado_1	357	22277	51/49
GSE50579	Hígado_2	76	36547	84/16
GSE62232	Hígado_3	91	54675	89/11
GSE22529_U133A	Leucemia_1	52	22283	79/21
GSE22529_U133B	Leucemia_2	52	22645	79/21
GSE33615	Leucemia_3	71	33579	70/30
GSE63270	Leucemia_4	101	54675	59/41
Golub et al. [94]	Leucemia_5	72	7129	65/35
GSE22820	Mama_1	139	33579	93/07
GSE59246	Mama_2	101	36622	55/45
GSE70947	Mama_3	289	35981	51/49
Van De Vijver et al. [95]	Mama_4	97	24481	53/43
GSE6919_U95Av2	Próstata_1	124	12625	50/50
GSE6919_U95B	Próstata_2	124	12620	52/48
GSE11682	Próstata_3	31	33467	52/48
Singh et al. [96]	Próstata_4	136	12600	57/43

Capítulo 6

Análisis de resultados

6.1. Resultados de optimización de hiperparámetros para los operadores evolutivos de SMS-MONEAT y N3O

Los resultados obtenidos durante la optimización de hiperparámetros de SMS-MONEAT y N3O se presentan en las Tablas 6.1 y 6.2, respectivamente. En cada una de las tablas se muestran las 4 mejores configuraciones encontradas para cada algoritmo (ordenadas de mejor a peor desempeño) y el promedio de estas. Se pueden observar ciertas similitudes en las configuraciones obtenidas. Para SMS-MONEAT todas las configuraciones son similares: una probabilidad de cruza aproximándose al valor mínimo definido de 0.5 (limites mostrados en la Tabla 5.2) y las probabilidades de mutaciones cercanas o mayores a 0.10 a excepción de la probabilidad de mutación de peso que es cercana a 0.5. Mientras que para N3O, se ve una mayor diferencia entre las configuraciones obtenidas. Para la probabilidad de cruza la configuración 3 resalta con un valor mayor en la probabilidad de cruza y probabilidad para agregar un nuevo nodo y uno menor para la probabilidad de agregar una nueva conexión.

Para ambos algoritmos, la mejor configuración tiene un valor similar para la probabilidad de cruza siendo de 56.62 % para SMS-MONEAT y 59.35 %, que como se mencionó anteriormente, se aproxima del límite inferior definido para este parámetro. Esto puede ser debido a que este operador puede ser más disruptivo que los operadores de mutación topológica. N3O realiza cruza entre individuos de la misma especie (en su mayoría) los cuales tienen una topología similar y utiliza la función de pérdida para definir al padre más apto y hace menos común que haya dos padres con la misma actitud. Sin embargo, para SMS-MONEAT, la selección de padres no está influenciada por un proceso de especiación por lo que hay mayor probabilidad de elegir dos padres con topología muy distinta y utilizar el rango de cada individuo puede ocasionar que las conexiones se agreguen de una forma más aleatoria, generando un nuevo individuo más distinto a

Tabla 6.1: Mejores configuraciones obtenidas para SMS-MONEAT mediante iRace.

Configuración	P_c	P_α	P_β	P_γ	P_δ	P_ω
1	0.5662	0.1226	0.0972	0.1352	0.1390	0.0498
2	0.5198	0.1283	0.1126	0.1353	0.1489	0.0501
3	0.5803	0.1443	0.1205	0.1425	0.1369	0.0450
4	0.5234	0.1035	0.1189	0.1164	0.1215	0.0613
Promedio	**0.5474**	**0.1247**	**0.1123**	**0.1324**	**0.1366**	**0.0516**

sus padres. Que este proceso sea más disruptivo para SMS-MONEAT que para N3O, puede que sea la razón por la cual en el promedio para este parámetro en las configuraciones encontradas sea menor para SMS-MONEAT.

Respecto a los operadores de mutación para agregar nuevas características a las ANNs, se puede observar un promedio ligeramente superior para agregar una nueva entrada sobre el operador que sustituye para SMS-MONEAT. El operador para agregar nuevas entradas se guía mediante el valor p obtenido en la prueba de KW, a diferencia del que sustituye una nueva entrada que se hace de manera aleatoria, esta podría ser la razón por la cual el primero tiene una probabilidad mayor. Por otro lado, la configuración 1, 2, y 4 encontradas para N3O, muestran un valor muy bajo para la probabilidad de agregar una nueva entrada, siendo el más alto de estos 0.0175. Es probable, que este operador no tenga gran impacto en el desempeño de N3O como tal, y que sea el operador para sustituir entradas el que tenga un mayor peso en el proceso de selección de características.

En las configuraciones de SMS-MONEAT, los operadores que permiten agregar una nueva conexión o un nuevo nodo tienen valores más altos respecto al resto de operadores de mutación. Para N3O, el operador para agregar una nueva conexión tiene un valor menor al de agregar una nueva conexión. Esto podría tener relación con el valor bajo del operador para agrega nuevas entradas, siendo que si no hay nuevas entradas no es tan necesario agregar tantas nuevas conexiones, pero sí agregar nuevos nodos ocultos.

Por último, la probabilidad de mutación de pesos es menor en las configuraciones de SMS-MONEAT en comparación a las de N3O, siendo este operador el que tiene un promedio más bajo para SMS-MONEAT siendo de 5.16 %, mientras que para N3O es de 11.24 %.

Aunque los algoritmos tengan características similares, las configuraciones encontradas para cada uno son diferentes lo cual implica que tienen un comportamiento distinto. Esto también puede ser ocasionado por el objetivo utilizado en la optimización de cada algoritmo, siendo el hipervolumen de la población para SMS-MONEAT y el promedio de la función de pérdida para N3O. Cabe resaltar que las configuraciones encontradas son específicas para el problema en cuestión y con los conjuntos de datos utilizados, por lo que no implica un buen desempeño en una tarea distinta.

Tabla 6.2: Mejores configuraciones obtenidas para N3O mediante iRace.

Configuración	P_c	P_α	P_β	P_γ	P_δ	P_ω
1	0.5925	0.0016	0.1239	0.0874	0.1339	0.0980
2	0.5774	0.0175	0.1358	0.0922	0.1394	0.1081
3	0.8604	0.1391	0.1326	0.0569	0.1332	0.1251
4	0.5678	0.0106	0.1483	0.1154	0.1421	0.1185
Promedio	**0.6495**	**0.0422**	**0.1352**	**0.0880**	**0.1372**	**0.1124**

6.2. Resultados experimentales sobre los conjuntos de datos de microarreglos

En esta sección se presentan los resultados obtenidos durante los experimentos realizados a los 20 conjuntos de datos de microarreglos de diversos tipos de cáncer. Primero se presentan el número de genes obtenidos durante el proceso de filtro estadístico (Subsección 6.2.1), estos valores son válidos para la metodología de SMS-MONEAT y N3O, puesto a que se utilizó la misma semilla y división de los conjuntos de datos durante los experimentos. Después se presenta una comparación mediante distintos indicadores de los resultados obtenidos por las diferentes metodologías: SMS-MONEAT, N3O y SMS-EMOA/kNN (Subsección 6.2.2).

6.2.1. Selección de genes mediante el filtro estadístico

La Tabla 6.3 muestra el promedio y desviación estándar de características seleccionadas por la metodología de filtro basado en la prueba H de KW para cada uno de los conjuntos de datos. En el último renglón se muestra el promedio del número de genes del conjunto de datos completo y del número de genes seleccionados, siendo en promedio un 25.69 % de genes los que se seleccionan en este paso. Realizar este proceso previo a ejecutar la metodología envolvente permite un mejor desempeño de este al eliminar la mayoría de posibles genes irrelevantes. Sin embargo, podemos observar que dependiendo del conjunto de datos el promedio de genes seleccionados varía desde 4.30 % (Mama_4) hasta 65.16 % (Hígado_1). Para reducir esta variación se podría utilizar diferentes valores para el umbral utilizado, pero durante esta investigación se mantuvo un umbral constante ya que se desea observar el comportamiento de los algoritmos ante conjuntos de datos con diferentes propiedades.

Tabla 6.3: Promedio y desviación estándar del número de genes seleccionados mediante el filtro estadístico basado en KW.

Conjunto de datos	Número de genes	Número de genes seleccionados	Porcentaje de genes seleccionados
$Colon_1$	24526	5668.67 ±551.84	23.11 ±2.25
$Colon_2$	54675	13464.80 ±1717.52	24.63 ±3.14
$Colon_3$	49386	27907.30 ±254.99	56.51 ±0.52
$Colon_4$	22277	5504.00 ±1121.00	24.71 ±5.03
$Hígado_1$	22277	14516.00 ±334.47	65.16 ±1.50
$Hígado_2$	36547	3450.50 ±1084.14	9.44 ±2.97
$Hígado_3$	54675	15200.67 ±1190.94	27.80 ±2.03
$Leucemia_1$	22283	2214.63 ±478.68	9.94 ±2.15
$Leucemia_2$	22645	1794.30 ±612.03	7.92 ±2.70
$Leucemia_3$	33579	10031.83 ±267.23	29.88 ±0.80
$Leucemia_4$	54675	14493.73 ±832.36	26.51 ±1.52
$Leucemia_5$	7129	892.37 ±143.13	12.52 ±2.01
$Mama_1$	33579	7526.73 ±308.25	22.42 ±0.92
$Mama_2$	36622	4185.67 ±807.20	11.43 ±2.20
$Mama_3$	35981	17322.87 ±517.40	48.14 ±1.44
$Mama_4$	24481	1051.47 ±419.88	4.30 ±1.72
$Próstata_1$	33467	1479.07 ±435.39	4.42 ±1.30
$Próstata_2$	12625	1019.13 ±185.53	8.07 ±1.47
$Próstata_3$	12620	616.33 ±147.29	4.88 ±1.17
$Próstata_4$	12600	1845.47 ±730.49	14.65 ±5.80
Promedio	**30332.45**	**7488.48 ±598.75**	**24.69 ±1.97**

6.2.2. Comparación de las metodologías para la selección de genes y clasificación de microarreglos

Primero se evaluó la metodología para seleccionar una solución de la población final para las metodologías basadas en neuroevolución. Después se realizó una comparación entre todas las metodologías: SMS-MONEAT, N3O y SMS-EMOA. Los resultados obtenidos para cada una de las metodologías se compararon utilizando los siguientes criterios:

- El valor de la función de entropía cruzada binaria de las soluciones elegidas evaluadas con el conjunto de prueba.
- El número de características seleccionadas de las soluciones elegidas.
- El hipervolumen de las poblaciones finales en cada experimento de acuerdo al número de características seleccionadas y la función de costo evaluando al conjunto de prueba.
- El promedio geométrico de las funciones elegidas evaluadas con el conjunto de prueba.
- La calidad de las características seleccionadas entrenando un modelo de clasificación SVM con ellas y evaluando su desempeño mediante el promedio geométrico en el conjunto de prueba.
- Tiempo de ejecución de los algoritmos con el mismo número de evaluaciones.

También se realizó una breve comparación al final entre las topologías de las ANNs obtenidas mediante las metodologías basadas en neuroevolución.

Selección de soluciones para las metodologías basadas en neuroevolución

Como se menciona en la Sección 4.5, se utilizó un conjunto de validación para la selección de soluciones de la población final de las distintas metodologías. Para diferenciar entre la población final de SMS-MONEAT contra la población de soluciones almacenadas en el archivo externo, se definirá como SMS-MONEAT$_P$ a la primera y como SMS-MONEAT$_Q$ a la segunda.

El tomador de decisiones selecciona aquella con el valor mínimo respecto a la suma ponderada mostrada en la Ecuación 4.7. Dicha ecuación toma 4 términos: la función de pérdida y el promedio geométrico obtenidos al evaluar en el conjunto de entrenamiento y al evaluar en el conjunto de validación. Incluyendo los términos relacionados al conjunto de validación, se espera que se elijan soluciones que generalicen mejor ante nuevas muestras.

Para observar el impacto de utilizar el conjunto de validación se comparó la solución seleccionada de esta manera contra una solución obtenida utilizando únicamente con el conjunto de entrenamiento (ignorando los términos relacionados al conjunto de validación de la Ecuación 4.7). La Tabla 6.4 muestra el

promedio y desviación estándar obtenidas durante los experimentos realizados utilizando como métrica el promedio geométrico (evaluando ante el conjunto de prueba) para N3O, SMS-MONEAT$_P$ y SMS-MONEAT$_Q$, respectivamente. Para N3O, el promedio obtenido al utilizar únicamente el conjunto de entrenamiento fue de 0.8010, mientras que al incluir al conjunto de validación subió a 0.8044. Para SMS-MONEAT, el promedio bajó para las soluciones seleccionadas de la población final de 0.7993 a 0.7975, pero para las obtenidas del archivo externo este valor subió de 0.8036 a 0.8128, siendo la que tuvo un mayor cambio de las 3 metodologías. Se notó un impacto positivo al utilizar el conjunto de validación para N3O y para SMS-MONEAT$_Q$.

En las comparaciones descritas en las siguientes subsecciones, las soluciones seleccionadas para cada población harán referencia a las que utilizan el conjunto de validación.

Tabla 6.4: Comparación entre el promedio geométrico (PGeo) de las soluciones seleccionadas utilizando el únicamente el conjunto de entrenamiento (subíndice e) o incluyendo el conjunto de validación (subíndice ev) SMS-MONEAT$_P$, SMS-MONEAT$_Q$ y N3O. Se marca el valor más alto obtenido con cada conjunto de datos para cada una de las metodologías.

Conjunto de datos	SMS-MONEAT$_P$		SMS-MONEAT$_Q$		N3O	
	PGeo$_e$	PGeo$_{ev}$	PGeo$_e$	PGeo$_{ev}$	PGeo$_e$	PGeo$_{ev}$
Colon$_1$	0.9124 ± 0.15	0.9161 ± 0.12	0.9044 ± 0.16	0.9247 ± 0.13	0.8829 ± 0.21	0.9025 ± 0.21
Colon$_2$	0.8971 ± 0.27	0.8971 ± 0.27	0.8971 ± 0.27	0.8874 ± 0.27	0.9414 ± 0.12	0.9512 ± 0.11
Colon$_3$	0.9624 ± 0.04	0.9624 ± 0.04	0.9680 ± 0.04	0.9678 ± 0.04	0.9714 ± 0.03	0.9714 ± 0.03
Colon$_4$	0.7989 ± 0.15	0.7963 ± 0.15	0.8155 ± 0.13	0.8174 ± 0.14	0.8418 ± 0.13	0.8355 ± 0.14
Hígado$_1$	0.9020 ± 0.06	0.8999 ± 0.06	0.9002 ± 0.06	0.9004 ± 0.07	0.9258 ± 0.05	0.9248 ± 0.05
Hígado$_2$	0.7748 ± 0.36	0.8022 ± 0.33	0.8512 ± 0.30	0.7885 ± 0.36	0.7201 ± 0.41	0.7201 ± 0.41
Hígado$_3$	0.9453 ± 0.06	0.9449 ± 0.06	0.9477 ± 0.05	0.9520 ± 0.06	0.9804 ± 0.04	0.9823 ± 0.04
Leucemia$_1$	0.8435 ± 0.34	0.8768 ± 0.30	0.8403 ± 0.34	0.8626 ± 0.30	0.8435 ± 0.34	0.8435 ± 0.34
Leucemia$_2$	0.8622 ± 0.34	0.8622 ± 0.34	0.8577 ± 0.34	0.8875 ± 0.30	0.7911 ± 0.40	0.8102 ± 0.37
Leucemia$_3$	0.9673 ± 0.08	0.9673 ± 0.08	0.9673 ± 0.08	0.9708 ± 0.08	0.9734 ± 0.08	0.9734 ± 0.08
Leucemia$_4$	0.8963 ± 0.11	0.8903 ± 0.11	0.9141 ± 0.09	0.9153 ± 0.10	0.9306 ± 0.07	0.9379 ± 0.06
Leucemia$_5$	0.8454 ± 0.21	0.8276 ± 0.22	0.8431 ± 0.20	0.8394 ± 0.20	0.8364 ± 0.16	0.8309 ± 0.15
Mama$_1$	0.9237 ± 0.25	0.9197 ± 0.25	0.9237 ± 0.25	0.9867 ± 0.02	0.9240 ± 0.25	0.9573 ± 0.18
Mama$_2$	0.7589 ± 0.14	0.7424 ± 0.14	0.7422 ± 0.13	0.7562 ± 0.14	0.7850 ± 0.13	0.7879 ± 0.11
Mama$_3$	0.8054 ± 0.09	0.8052 ± 0.09	0.8018 ± 0.08	0.7945 ± 0.07	0.8118 ± 0.09	0.8088 ± 0.10
Mama$_4$	0.6603 ± 0.14	0.6329 ± 0.17	0.6646 ± 0.15	0.6451 ± 0.16	0.6244 ± 0.14	0.6344 ± 0.15
Próstata$_1$	0.5997 ± 0.15	0.5975 ± 0.15	0.5825 ± 0.14	0.5958 ± 0.13	0.6036 ± 0.17	0.6012 ± 0.17
Próstata$_2$	0.6184 ± 0.14	0.6019 ± 0.14	0.6100 ± 0.14	0.5979 ± 0.16	0.5948 ± 0.13	0.6015 ± 0.14
Próstata$_3$	0.3886 ± 0.43	0.3886 ± 0.43	0.4024 ± 0.42	0.5300 ± 0.40	0.3926 ± 0.41	0.3593 ± 0.40
Próstata$_4$	0.6226 ± 0.17	0.6240 ± 0.15	0.6402 ± 0.16	0.6352 ± 0.18	0.6446 ± 0.17	0.6540 ± 0.18
Promedio	0.7993 ± 0.26	0.7978 ± 0.26	0.8037 ± 0.25	0.8128 ± 0.24	0.8010 ± 0.26	0.8044 ± 0.26

Comparación utilizando la función de pérdida y la cantidad de características seleccionadas

Se hizo una comparación entre las soluciones seleccionadas por cada una de las metodologías: SMS-MONEAT$_P$, SMS-MONEAT$_Q$, N3O y SMS-EMOA. Las Tablas 6.7, 6.5 y 6.6 muestran los promedio y desviación estándar obtenidos respecto a la función de pérdida (entropía cruzada descrita en la Ecuación 2.5a) y número de características promedio para las metodologías de SMS-MONEAT, N3O y SMS-EMOA, respectivamente.

La Tabla 6.8 muestra la comparación de los resultados obtenidos entre todas las metodologías. Se puede observar que en cuanto al número de características seleccionadas la metodología de N3O se ve rezagada ante las otras 3, esto hace sentido puesto que no es un objetivo a optimizar por esta metodología. Por otro lado, la metodología de SMS-EMOA se ve rezagada ante la función de entropía cruzada. Respecto al valor promedio de todos los experimentos (último renglón de la tabla), N3O es el que tiene un valor de entropía cruzada menor (0.3948) mientras que SMS-MONEAT$_P$ tiene un menor número de características seleccionadas (2.88).

Se realizaron pruebas no paramétricas de Friedman y un post hoc para comparar los resultados obtenidos por cada metodología. Se utilizó una prueba emparejada debido a que se utilizó la misma semilla para la validación cruzada de los experimentos y, por lo tanto, el conjunto de entrenamiento y prueba eran el mismo para todas las metodologías. La simbología de flechas utilizada en la tabla permite expresar la relación entre las distintas metodologías, en la columna de SMS-MONET$_Q$ se muestra su relación contra SMS-MONEAT$_P$ únicamente, mientras que en las columnas de N3O y SMS-EMOA se muestra su relación respecto a SMS-MONEAT$_P$ y SMS-MONEAT$_Q$, en dicho orden. La flecha $\leftrightarrow$ significa que no se encontró diferencia significativa entre las metodologías con dicha métrica, mientras que las flechas $\uparrow$ y $\downarrow$ significan que sí se encontró diferencia significativa, siendo hacia arriba a favor de dicha metodología y la flecha hacia abajo en contra.

Al comparar las metodologías respecto a la entropía cruzada, no se encontró diferencia significativa entre las soluciones de SMS-MONEAT. Tampoco se encontró diferencia significativa entre SMS-MONEAT$_P$ y N3O, pero sí hubo diferencias significativas entre SMS-MONEAT$_Q$ y N3O en el conjunto de Mama$_2$ favorable para este último. Por otro lado, respecto a SMS-EMOA se demostró diferencia significativa en 12 conjuntos de datos (Colon$_4$, Hígado$_1$, Leucemia$_2$, Leucemia$_3$ y en todos los conjuntos de cáncer de mama y próstata) respecto a SMS-MONEAT$_P$ y N3O. Mientras que entre SMS-EMOA y SMS-MONEAT$_Q$ hubo diferencia significativa en 11 conjuntos de datos (los mismos que SMS-MONEAT$_P$ excepto Leucemia$_2$)

Por otro lado, al realizar la prueba de Friedman para el número de características seleccionadas se obtuvo que existe una diferencia significativa en todos los conjuntos de datos entre N3O y las soluciones de SMS-MONEAT. Además, también se encontró diferencia significativa a favor de SMS-EMOA al comparar contra N3O en 16 conjuntos de datos (menos en Colon$_4$, Hígado$_1$, Mama$_3$ y

Mama_4). Entre SMS-MONEAT$_P$ y SMS-MONEAT$_Q$ tampoco se encontró diferencia significativa en ninguno de los experimentos. Mientras que en relación a SMS-EMOA, se encontró diferencia significativa a favor de SMS-MONEAT$_P$ en 7 conjuntos de datos (Colon_4, Hígado_1, Mama_3, Mama_4, Próstata_1, Próstta_2 y Próstata_4) y sólo 1 en contra (Leucemia_1). También se encontró diferencia significativa entre SMS-MONEAT$_Q$ y SMS-EMOA, dos a favor de este último (Leucemia_2 y Leucemia_3) y 2 en contra (Hígado_1 y Mama_4).

Los resultados obtenidos durante post hoc de la prueba de Friedman fueron utilizados para comparar las soluciones mediante la metodología de Condorcet. La Tabla 6.9 muestra los resultados para la entropía cruzada mientras que la Tabla 6.10 muestra los del número de características seleccionadas. Estas tablas reflejan numéricamente lo que se describe en los dos párrafos anteriores, siendo N3O el ganador de Condorcet respecto a los valores de entropía cruzada obtenidos y SMS-MONEAT$_P$ el ganador respecto al número de características seleccionadas.

Con los resultados mencionados, se puede intuir que entre SMS-MONEAT$_P$ y SMS-MONEAT$_Q$ no hay diferencia significativa ni en la función de pérdida ni en el número de genes seleccionados, sin embargo, el archivo externo, al buscar mantener una mayor diversidad en las soluciones tiende a generalizar mejor ante nuevas muestras reflejado en un promedio menor de entropía cruzada, pero a un mayor número de características seleccionadas. Al comparar contra N3O, sólo se encuentra diferencia significativa en un conjunto de datos respecto a SMS-MONEAT$_Q$ y en ninguno ante SMS-MONEAT$_P$, sin embargo, sí hay una diferencia significativa en todos los experimentos respecto al número de características seleccionadas, por lo que SMS-MONEAT puede encontrar soluciones con valores similares de entropía cruzada, pero con un menor número de características. Finalmente, al comparar contra SMS-EMOA, los resultados muestran una ventaja de los modelos de ANNs generados por SMS-MONEAT y N3O sobre los de KNN utilizados por SMS-EMOA respecto a la función de pérdida mostrando una diferencia significativa en más de la mitad de los experimentos.

Para mayor detalle sobre los resultados descritos en esta subsección se puede referir a las Figuras 1.1, 1.2 y 1.3 del Apéndice A en el cual se muestran Box Plots sobre los resultados obtenidos respecto a la entropía cruzada y el número de características seleccionadas.

Tabla 6.5: Resultados obtenidos del valor de entropía cruzada (EC) y número de genes ante el conjunto de pruebas para las soluciones de N3O.

Conjunto de datos	EC	Genes
$Colon_1$	0.2646± 0.30	13.80± 14.26
$Colon_2$	0.1368± 0.12	16.00± 16.25
$Colon_3$	0.1466± 0.07	11.50± 20.69
$Colon_4$	0.4520± 0.28	9.67± 5.23
$Hígado_1$	0.2760± 0.11	6.73± 4.68
$Hígado_2$	0.4030± 0.41	8.40± 3.81
$Hígado_3$	0.1342± 0.09	9.50± 6.43
$Leucemia_1$	0.4086± 0.84	8.97± 7.64
$Leucemia_2$	0.3248± 0.46	9.07± 5.27
$Leucemia_3$	0.0776± 0.08	11.90± 17.64
$Leucemia_4$	0.1889± 0.13	7.97± 4.79
$Leucemia_5$	0.3230± 0.22	10.03± 6.57
$Mama_1$	0.1351± 0.16	13.37± 14.87
$Mama_2$	0.4687± 0.12	11.13± 4.49
$Mama_3$	0.4354± 0.12	7.20± 3.81
$Mama_4$	0.6524± 0.23	9.23± 4.20
$Próstata_1$	0.7095± 0.24	10.40± 3.64
$Próstata_2$	0.7038± 0.24	9.80± 3.55
$Próstata_3$	1.0628± 0.70	11.27± 6.74
$Próstata_4$	0.5928± 0.17	7.30± 3.71
Promedio	0.3948± 0.40	10.16± 9.67

Tabla 6.6: Resultados obtenidos del valor de entropía cruzada (EC) y número de genes ante el conjunto de pruebas para las soluciones de SMS-EMOA.

Conjunto de datos	EC	Genes
$Colon_1$	7.6439± 13.22	3.07± 1.39
$Colon_2$	8.8550± 15.76	1.33± 0.48
$Colon_3$	2.2455± 3.57	2.63± 1.00
$Colon_4$	16.4272± 12.91	4.47± 1.96
$Higado_1$	3.5434± 2.92	8.80± 3.10
$Higado_2$	13.9422± 22.20	4.40± 2.13
$Higado_3$	2.4910± 3.76	1.87± 0.51
$Leucemia_1$	12.3958± 22.21	2.23± 1.45
$Leucemia_2$	12.9792± 22.36	1.67± 0.84
$Leucemia_3$	3.2666± 8.27	1.57± 0.50
$Leucemia_4$	3.7892± 6.47	3.07± 1.48
$Leucemia_5$	9.7584± 11.34	3.70± 2.00
$Mama_1$	8.6279± 19.56	2.07± 0.25
$Mama_2$	16.3718± 10.05	6.03± 3.29
$Mama_3$	17.7360± 6.57	5.13± 3.04
$Mama_4$	27.2599± 12.91	6.20± 3.02
$Próstata_1$	29.2038± 12.30	6.23± 3.27
$Próstata_2$	31.1768± 10.80	6.17± 3.04
$Próstata_3$	29.0371± 21.41	3.37± 1.10
$Próstata_4$	15.0791± 10.60	4.70± 2.96
Promedio	13.5915± 16.48	3.94± 2.87

Tabla 6.7: Resultados obtenidos del valor de entropía cruzada (EC) y número de genes ante el conjunto de pruebas para las soluciones de SMS-MONEAT.

Conjunto de datos	SMS-MONEAT$_P$		SMS-MONEAT$_Q$	
	EC	Genes	EC	Genes
$Colon_1$	0.2964± 0.44	2.53± 1.22	0.3089± 0.44	3.07± 1.57
$Colon_2$	0.1962± 0.25	2.00± 1.23	0.2415± 0.37	2.17± 0.91
$Colon_3$	0.1420± 0.07	2.10± 0.96	0.1456± 0.07	2.37± 1.03
$Colon_4$	0.4671± 0.25	2.63± 0.89	0.4881± 0.30	2.90± 1.06
$Hígado_1$	0.3184± 0.10	2.10± 0.88	0.3117± 0.09	2.33± 0.92
$Hígado_2$	0.3568± 0.41	2.83± 1.18	0.4172± 0.50	2.83± 1.21
$Hígado_3$	0.1775± 0.12	1.53± 0.97	0.1693± 0.12	2.13± 1.66
$Leucemia_1$	0.3843± 0.85	4.27± 10.41	0.3498± 0.62	4.37± 10.23
$Leucemia_2$	0.3930± 0.91	3.07± 1.26	0.3581± 0.88	3.50± 3.54
$Leucemia_3$	0.1236± 0.29	2.53± 1.14	0.0926± 0.13	2.87± 1.28
$Leucemia_4$	0.2254± 0.11	3.47± 6.07	0.2286± 0.11	4.23± 6.70
$Leucemia_5$	0.3170± 0.19	3.30± 1.29	0.3243± 0.18	3.53± 1.14
$Mama_1$	0.1495± 0.18	2.90± 1.35	0.1522± 0.15	3.13± 1.28
$Mama_2$	0.5792± 0.25	3.67± 1.47	0.5584± 0.18	4.17± 1.37
$Mama_3$	0.4478± 0.09	2.47± 1.20	0.4530± 0.09	2.90± 1.18
$Mama_4$	0.6518± 0.17	3.23± 1.70	0.6226± 0.18	3.47± 1.55
$Próstata_1$	0.6851± 0.15	3.53± 1.59	0.6683± 0.12	3.83± 1.68
$Próstata_3$	0.7400± 0.20	4.07± 1.51	0.6960± 0.15	4.60± 1.43
$Próstata_4$	0.8654± 0.58	2.53± 1.17	0.7307± 0.40	2.77± 1.41
$Próstata_4$	0.6232± 0.18	2.77± 1.45	0.6000± 0.15	3.43± 1.76
Promedio	0.4070± 0.42	2.88± 2.98	0.3958± 0.38	3.23± 3.14

Tabla 6.8: Resultados obtenidos de acuerdo con la entropía cruzada (EC) evaluada sobre el conjunto de prueba y el número de genes seleccionados para los experimentos con las diferentes metodologías: SMS-MONEAT, N3O y SMS-EMOA. El mejor valor obtenido en cada uno de los conjuntos de datos se marca de un color, siendo lavanda para la EC y melón para el número de genes. La simbología de flechas indica diferencias significativas entre las metodologías obtenida mediante el post hoc de la prueba de Friedman: para SMS-MONEATQ se muestra la relación con SMS-MONEAT$_P$, mientras que para N3O y SMS-EMOA se muestra la relación con las dos soluciones de SMS-MONEAT (P izquierda y Q derecha).

Conjunto de datos	SMS-MONEAT$_P$		SMS-MONEAT$_Q$		N3O		SMS-EMOA	
	EC	Genes	EC	Genes	EC	Genes	EC	Genes
Colon$_1$	0.2964	2.53	0.3089 $\leftrightarrow$	3.07 $\leftrightarrow$	0.2646 $\leftrightarrow\leftrightarrow$	13.80 $\downarrow\downarrow$	7.6439 $\leftrightarrow\leftrightarrow$	3.07 $\leftrightarrow\leftrightarrow$
Colon$_2$	0.1962	2.00	0.2415 $\leftrightarrow$	2.17 $\leftrightarrow$	0.1368 $\leftrightarrow\leftrightarrow$	16.00 $\downarrow\downarrow$	8.8550 $\leftrightarrow\leftrightarrow$	1.33 $\leftrightarrow\leftrightarrow$
Colon$_3$	0.1420	2.10	0.1456 $\leftrightarrow$	2.37 $\leftrightarrow$	0.1466 $\leftrightarrow\leftrightarrow$	11.50 $\downarrow\downarrow$	2.2455 $\leftrightarrow\leftrightarrow$	2.63 $\leftrightarrow\leftrightarrow$
Colon$_4$	0.4671	2.63	0.4881 $\leftrightarrow$	2.90 $\leftrightarrow$	0.4520 $\leftrightarrow\leftrightarrow$	9.67 $\downarrow\downarrow$	16.4272$\downarrow\downarrow$	4.47 $\downarrow\leftrightarrow$
Hígado$_1$	0.3184	2.10	0.3117 $\leftrightarrow$	2.33 $\leftrightarrow$	0.2760 $\leftrightarrow\leftrightarrow$	6.73 $\downarrow\downarrow$	3.5434 $\downarrow\downarrow$	8.80 $\downarrow\downarrow$
Hígado$_2$	0.3568	2.83	0.4172 $\leftrightarrow$	2.83$\leftrightarrow$	0.4030 $\leftrightarrow\leftrightarrow$	8.40 $\downarrow\downarrow$	13.9422$\leftrightarrow\leftrightarrow$	4.40 $\leftrightarrow\leftrightarrow$
Hígado$_3$	0.1775	1.53	0.1693 $\leftrightarrow$	2.13$\leftrightarrow$	0.1342 $\leftrightarrow\leftrightarrow$	9.50 $\downarrow\downarrow$	2.4910 $\leftrightarrow\leftrightarrow$	1.87 $\leftrightarrow\leftrightarrow$
Leucemia$_1$	0.3843	4.27	0.3498$\leftrightarrow$	4.37$\leftrightarrow$	0.4086 $\leftrightarrow\leftrightarrow$	8.97 $\downarrow\downarrow$	12.3958$\leftrightarrow\leftrightarrow$	2.23 $\leftrightarrow\leftrightarrow$
Leucemia$_2$	0.3930	3.07	0.3581 $\leftrightarrow$	3.50$\leftrightarrow$	0.3248 $\leftrightarrow\leftrightarrow$	9.07 $\downarrow\downarrow$	12.9792$\leftrightarrow\downarrow$	1.67 $\uparrow\uparrow$
Leucemia$_3$	0.1236	2.53	0.0926 $\leftrightarrow$	2.87$\leftrightarrow$	0.0776 $\leftrightarrow\leftrightarrow$	11.90 $\downarrow\downarrow$	3.2666 $\downarrow\downarrow$	1.57 $\leftrightarrow\uparrow$
Leucemia$_4$	0.2254	3.47	0.2286 $\leftrightarrow$	4.23$\leftrightarrow$	0.1889 $\leftrightarrow\leftrightarrow$	7.97 $\downarrow\downarrow$	3.7892 $\leftrightarrow\leftrightarrow$	3.07 $\leftrightarrow\leftrightarrow$
Leucemia$_5$	0.3170	3.30	0.3243 $\leftrightarrow$	3.53$\leftrightarrow$	0.3230 $\leftrightarrow\leftrightarrow$	10.03 $\downarrow\downarrow$	9.7584 $\leftrightarrow\leftrightarrow$	3.70 $\leftrightarrow\leftrightarrow$
Mama$_1$	0.1495	2.90	0.1522 $\leftrightarrow$	3.13$\leftrightarrow$	0.1351 $\leftrightarrow\leftrightarrow$	13.37 $\downarrow\downarrow$	8.6279 $\downarrow\downarrow$	2.07 $\leftrightarrow\leftrightarrow$
Mama$_2$	0.5792	3.67	0.5584 $\leftrightarrow$	4.17$\leftrightarrow$	0.4687 $\leftrightarrow\uparrow$	11.13 $\downarrow\downarrow$	16.3718$\downarrow\downarrow$	6.03 $\leftrightarrow\leftrightarrow$
Mama$_3$	0.4478	2.47	0.4530 $\leftrightarrow$	2.90$\leftrightarrow$	0.4354 $\leftrightarrow\leftrightarrow$	7.20 $\downarrow\downarrow$	17.7360$\downarrow\downarrow$	5.13 $\downarrow\leftrightarrow$
Mama$_4$	0.6518	3.23	0.6226 $\leftrightarrow$	3.47$\leftrightarrow$	0.6524 $\leftrightarrow\leftrightarrow$	9.23 $\downarrow\downarrow$	27.2599$\downarrow\downarrow$	6.20 $\downarrow\downarrow$
Próstata$_1$	0.6851	3.53	0.6683 $\leftrightarrow$	3.83$\leftrightarrow$	0.7095 $\leftrightarrow\leftrightarrow$	10.40 $\downarrow\downarrow$	29.2038$\downarrow\downarrow$	6.23 $\downarrow\leftrightarrow$
Próstata$_2$	0.7400	4.07	0.6960 $\leftrightarrow$	4.60$\leftrightarrow$	0.7038 $\leftrightarrow\leftrightarrow$	9.80 $\downarrow\downarrow$	31.1768$\downarrow\downarrow$	6.17 $\downarrow\leftrightarrow$
Próstata$_3$	0.8654	2.53	0.7307 $\leftrightarrow$	2.77$\leftrightarrow$	1.0628 $\leftrightarrow\leftrightarrow$	11.27 $\downarrow\downarrow$	29.0371$\downarrow\downarrow$	3.37 $\leftrightarrow\leftrightarrow$
Próstata$_4$	0.6232	2.77	0.6000 $\leftrightarrow$	3.43 $\leftrightarrow$	0.5928 $\leftrightarrow\leftrightarrow$	7.30 $\downarrow\downarrow$	15.0791$\downarrow\downarrow$	4.70 $\downarrow\leftrightarrow$
Promedio	0.4070	2.88	0.3958 $\leftrightarrow$	3.23	0.3948	10.16	13.5915	3.94

Tabla 6.9: Comparación entre las metodologías SMS-MONEAT, N3O y SMS-EMOA utilizando la metodología de Condorcet y los resultados obtenidos del post hoc de la prueba de Friedman respecto a los resultados la función de entropía cruzada.

	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA	**Total**
SMS-MONEAT$_P$	X	0	0	11	**11**
SMS-MONEAT$_Q$	0	X	0	12	**12**
N3O	0	1	X	12	**13**
SMS-EMOA	0	0	0	X	**0**

Tabla 6.10: Comparación entre las metodologías SMS-MONEAT, N3O y SMS-EMOA utilizando la metodología de Condorcet y los resultados obtenidos del post hoc de la prueba de Friedman respecto a los resultados del número de genes seleccionados.

	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA	**Total**
SMS-MONEAT$_P$	X	0	20	7	**27**
SMS-MONEAT$_Q$	0	X	20	2	**22**
N3O	0	0	X	0	**0**
SMS-EMOA	1	2	16	X	**19**

Comparación utilizando el hipervolumen de la población

Otra métrica utilizada para comparar el desempeño de las metodologías fue el hipervolumen. Este fue calculado con el valor de entropía cruzada de todos los individuos de la población (evaluada en el conjunto de prueba) y el número de características como se muestra en la Ecuación 4.2. Sin embargo, el término de regularización en $g(a, y)$ (Ecuación 2.5a) es descartado debido a que está relacionado al número de conexiones de las ANNs y la metodología de SMS-EMOA utiliza un modelo de clasificación distinto y que de este modo fueran comparables.

Considerando los resultados discutidos en la subsección anterior, N3O mostraba un desempeño inferior respecto al número de características seleccionadas y SMS-EMOA respecto a la entropía cruzada, por lo que, esto tendría un impacto en el cálculo del hipervolumen. La Tabla 6.11 muestra los resultados obtenidos respecto al hipervolumen para las poblaciones finales de los experimentos realizados. En la tabla se observa un mejor desempeño por parte de las metodologías de SMS-MONEAT sobre las de N3O y SMS-EMOA. Se observa valores mayores en la población del archivo externo sobre la población final de SMS-MONEAT, esto puede ser debido a una mayor diversidad mantenida en el archivo externo.

Utilizando la prueba estadística de Friedman para comparar todas las metodologías se identificó una diferencia significativa positiva sobre las metodologías basadas en SMS-MONEAT sobre la de N3O en todos los conjuntos de datos. Mientras que al comparar SMS-EMOA contra SMS-MONEAT$_P$ se encontró diferencia significativa en 8 conjuntos de datos a favor de este último (Colon$_4$, Hígado$_1$, Mama$_2$, Mama$_3$, Mama$_4$, Próstata$_1$, Próstata$_2$ y Próstata$_4$). Al comparar SMS-EMOA contra SMS-MONEAT$_Q$ hubo diferencia significativa en 10 conjuntos de datos (Colon$_4$, Hígado$_1$, Hígado$_2$, Mama$_2$, Mama$_3$, Mama$_4$ y todos los conjuntos de datos de cáncer de próstata). Entre las metodologías de N3O y SMS-EMOA hubo diferencias significativas en 14 conjuntos de datos, 9 a favor de N3O (Colon$_1$, Colon$_2$, Hígado$_2$, Leucemia$_1$, Leucemia$_2$, Leucemia$_5$, Mama$_3$, Próstata$_1$ y Próstata$_2$) y 5 a favor de SMS-EMOA (Colon$_3$, Hígado$_3$, Leucemia$_3$, Leucemia$_4$ y Mama$_1$). Por último, entre las metodologías basadas en SMS-MONEAT hubo diferencias significativas en favor a la población del archivo externo en 6 conjuntos de datos (Hígado$_3$, Mama$_2$, Mama$_3$, Próstata$_1$, Próstata$_2$, y Próstata$_3$). Lo anterior se muestra en la Tabla 6.11 mediante el sistema de flechas.

La metodología de Condorcet fue utilizada nuevamente para comparar las metodologías respecto a los resultados del hipervolumen y las diferencias significativas observadas por la prueba de Friedman. Los resultados se presentan en la Tabla 6.12, donde se muestra que los resultados obtenidos mediante la metodología de SMS-MONEAT$_Q$ predomina en esta métrica.

Para mayor detalle sobre los resultados descritos en esta subsección se puede referir a las Figuras 1.4 y 1.5 del Apéndice A en el cual se muestran Box Plots con los resultados respecto al hipervolumen.

Tabla 6.11: Resultados obtenidos respecto al hipervolumen de la población final en cada metodología (SMS-MONEAT, N3O y SMS-EMOA), calculado utilizando el valor de entropía cruzada y el número de genes seleccionados. El mejor valor obtenido en cada uno de los conjuntos de datos se marca de color. La simbología de flechas indica diferencias significativas entre las metodologías obtenida mediante el post hoc de la prueba de Friedman: para SMS-MONEATQ se muestra la relación con SMS-MONEAT$_P$, mientras que para N3O y SMS-EMOA se muestra la relación con las dos soluciones de SMS-MONEAT (P izquierda y Q derecha).

Conjunto de datos	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA
Colon_1	890.1391 ± 5.89	892.2314 ± 1.87 ↔	825.5392 ± 72.35 ↓↓	824.7977 ± 140.59 ↔↔
Colon_2	891.3572 ± 3.25	893.2981 ± 0.57 ↔	819.8589 ± 78.32 ↓↓	787.6227 ± 202.68 ↔↔
Colon_3	891.9539 ± 1.02	892.0700 ± 0.92 ↔	851.3973 ± 63.69 ↓↓	870.6782 ± 41.29 ↔↔
Colon_4	887.6355 ± 3.25	889.2194 ± 1.90 ↔	842.2418 ± 23.58 ↓↓	720.6037 ± 167.33 ↓↓
Hígado_1	889.3346 ± 1.54	889.8277 ± 1.08 ↔	860.8483 ± 25.62 ↓↓	862.0325 ± 28.76 ↓↓
Hígado_2	890.7276 ± 3.87	892.2998 ± 1.25 ↔	846.5080 ± 21.60 ↓↓	736.0552 ± 294.20 ↔↓
Hígado_3	891.4882 ± 1.72	892.4187 ± 0.98 ↑	848.2047 ± 33.23 ↓↓	859.7219 ± 55.13 ↔↔
Leucemia_1	889.5443 ± 10.47	892.5274 ± 2.26 ↔	846.3312 ± 39.63 ↓↓	783.8303 ± 266.54 ↔↔
Leucemia_2	889.9511 ± 11.37	892.7660 ± 1.93 ↔	847.8242 ± 22.47 ↓↓	751.6472 ± 296.58 ↔↔
Leucemia_3	892.5275 ± 3.56	893.3415 ± 0.70 ↔	844.1492 ± 64.41 ↓↓	870.7339 ± 82.82 ↔↔
Leucemia_4	890.8813 ± 1.59	891.6579 ± 1.04 ↔	852.9067 ± 26.80 ↓↓	869.3459 ± 48.16 ↔↔
Leucemia_5	890.3261 ± 2.46	891.0023 ± 1.76 ↔	841.2102 ± 26.33 ↓↓	823.1446 ± 140.66 ↔↔
Mama_1	892.2659 ± 2.59	892.9128 ± 1.14 ↔	830.2418 ± 78.05 ↓↓	841.3190 ± 194.86 ↔↔
Mama_2	886.4560 ± 2.55	888.2366 ± 1.23 ↑	829.6923 ± 26.57 ↓↓	761.6782 ± 105.24 ↓↓
Mama_3	887.4469 ± 1.38	888.0095 ± 1.13 ↑	855.1433 ± 19.92 ↓↓	687.7171 ± 87.57 ↓↓
Mama_4	885.4637 ± 2.48	886.9580 ± 1.50 ↔	837.8147 ± 23.67 ↓↓	627.1595 ± 148.06 ↓↓
Próstata_1	885.0709 ± 1.60	886.1112 ± 1.36 ↑	829.8830 ± 21.00 ↓↓	605.4408 ± 104.22 ↓↓
Próstata_2	884.6659 ± 2.05	886.1194 ± 1.01 ↑	834.7074 ± 17.73 ↓↓	555.5935 ± 115.93 ↓↓
Próstata_3	883.7440 ± 8.34	890.6341 ± 2.26 ↑	823.0365 ± 38.37 ↓↓	690.6214 ± 268.51 ↔↓
Próstata_4	885.3326 ± 2.27	886.1034 ± 1.76 ↔	850.2998 ± 19.88 ↓↓	760.7691 ± 111.10 ↓↓
Promedio	888.8156 ± 5.38	890.3873 ± 2.92	840.8919 ± 43.36	764.5256 ± 188.02

Tabla 6.12: Comparación entre las metodologías SMS-MONEAT, N3O y SMS-EMOA utilizando la metodología de Condorcet y los resultados obtenidos del post hoc de la prueba de Friedman respecto a los resultados del hipervolumen.

	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA	**Total**
SMS-MONEAT$_P$	x	0	20	8	**28**
SMS-MONEAT$_Q$	6	X	20	10	**36**
N3O	0	0	X	9	**9**
SMS-EMOA	0	0	5	X	**5**

Comparación utilizando el promedio geométrico de las soluciones seleccionadas

El promedio geométrico también fue utilizado como métrica para comparar las metodologías de este estudio. La Tabla 6.13 presenta los resultados obtenidos para cada metodología respecto al promedio geométrico. En ella, se puede observar que SMS-MONEAT$_Q$ es la que tiene un valor promedio más alto sobre el total de experimentos en comparación a las otras 3 metodologías con un valor de 0.8128. Por otro lado, la metodología de SMS-EMOA tiene el valor promedio más bajo, aun así, mostró el mejor desempeño en 4 conjuntos de datos: Higado$_1$, Leucemia$_4$, Leucemia$_5$ y Próstata$_4$, especialmente en este último destaca sobre las otras metodologías.

Tras realizar las pruebas post hoc de Friedman, se determinó una diferencia significativa entre SMS-EMOA y las metodologías basadas en SMS-MONEAT en 2 conjuntos de datos, Higado$_1$ y Próstata$_4$, en favor a la metodología de SMS-EMOA. Fueron las únicas diferencias significativas encontradas entre las metodologías con esta métrica, por lo que al realizar la prueba de Condorcet SMS-EMOA fue el ganador. Analizando con mayor detalle ambos conjuntos de datos en los que SMS-EMOA superó a SMS-MONEAT en esta métrica se notó que ambas cuentan con un alto número de muestras con relación a los demás conjuntos de datos: Higado$_1$ cuenta con 357 muestras y Próstata$_4$ con 136, siendo el promedio de 114.85 muestras. Además, ambos conjuntos tienen una cantidad balanceada de muestras. Estas propiedades en los conjuntos de datos pueden ser positivas para la metodología de SMS-EMOA, debido a que a diferencia de las metodologías de SMS-MONEAT y N3O, esta utiliza validación cruzada para evaluar a los individuos de la población. Cabe resaltar que tanto para las métricas de entropía cruzada, número de características e hipervolumen se determinó una diferencia significativa a favor de las metodologías de SMS-MONEAT en estos conjuntos de datos (excepto por SMS-MONEAT$_Q$ en el número de características).

Para mayor detalle sobre los resultados descritos en esta subsección se puede referir a la Figura 1.6 del Apéndice A en el cual se muestran Box Plots con los resultados respecto al promedio geométrico.

Tabla 6.13: Resultados obtenidos respecto al promedio geométrico de las soluciones seleccionadas evaluada en el conjunto de prueba en cada metodología (SMS-MONEAT, N3O y SMS-EMOA). El mejor valor obtenido en cada uno de los conjuntos de datos se marca de color. La simbología de flechas indica diferencias significativas entre las metodologías obtenida mediante el post hoc de la prueba de Friedman: para SMS-MONEATQ se muestra la relación con SMS-MONEAT$_P$, mientras que para N3O y SMS-EMOA se muestra la relación con las dos soluciones de SMS-MONEAT (P izquierda y Q derecha).

Conjunto de datos	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA
Colon$_1$	0.9161 ± 0.12	0.9247 ± 0.13 ↔	0.9025 ± 0.21 ↔↔	0.8970 ± 0.20 ↔↔
Colon$_2$	0.8971 ± 0.27	0.8874 ± 0.27 ↔	0.9512 ± 0.11 ↔↔	0.7776 ± 0.38 ↔↔
Colon$_3$	0.9624 ± 0.04	0.9678 ± 0.04 ↔	0.9714 ± 0.03 ↔↔	0.9703 ± 0.04 ↔↔
Colon$_4$	0.7963 ± 0.15	0.8174 ± 0.14 ↔	0.8355 ± 0.14 ↔↔	0.7655 ± 0.21 ↔↔
Hígado$_1$	0.8999 ± 0.06	0.9004 ± 0.07 ↔	0.9248 ± 0.05 ↔↔	0.9450 ± 0.03 ↑↑
Hígado$_2$	0.8022 ± 0.33	0.7885 ± 0.36 ↔	0.7201 ± 0.41 ↔↔	0.7389 ± 0.42 ↔↔
Hígado$_3$	0.9449 ± 0.06	0.9520 ± 0.06 ↔	0.9823 ± 0.04 ↔↔	0.9138 ± 0.25 ↔↔
Leucemia$_1$	0.8768 ± 0.30	0.8626 ± 0.30 ↔	0.8435 ± 0.34 ↔↔	0.7480 ± 0.42 ↔↔
Leucemia$_2$	0.8622 ± 0.34	0.8875 ± 0.30 ↔	0.8102 ± 0.37 ↔↔	0.7569 ± 0.43 ↔↔
Leucemia$_3$	0.9673 ± 0.08	0.9708 ± 0.08 ↔	0.9734 ± 0.08 ↔↔	0.9576 ± 0.09 ↔↔
Leucemia$_4$	0.8903 ± 0.11	0.9153 ± 0.10 ↔	0.9379 ± 0.06 ↔↔	0.9432 ± 0.09 ↔↔
Leucemia$_5$	0.8276 ± 0.22	0.8394 ± 0.20 ↔	0.8309 ± 0.15 ↔↔	0.8779 ± 0.13 ↔↔
Mama$_1$	0.9197 ± 0.25	0.9867 ± 0.02 ↔	0.9573 ± 0.18 ↔↔	0.8320 ± 0.38 ↔↔
Mama$_2$	0.7424 ± 0.14	0.7562 ± 0.14 ↔	0.7879 ± 0.11 ↔↔	0.7496 ± 0.14 ↔↔
Mama$_3$	0.8052 ± 0.09	0.7945 ± 0.07 ↔	0.8088 ± 0.10 ↔↔	0.7807 ± 0.09 ↔↔
Mama$_4$	0.6329 ± 0.17	0.6451 ± 0.16 ↔	0.6344 ± 0.15 ↔↔	0.6024 ± 0.18 ↔↔
Próstata$_1$	0.5975 ± 0.15	0.5958 ± 0.13 ↔	0.6012 ± 0.17 ↔↔	0.5431 ± 0.23 ↔↔
Próstata$_2$	0.6019 ± 0.14	0.5979 ± 0.16 ↔	0.6015 ± 0.14 ↔↔	0.5315 ± 0.22 ↔↔
Próstata$_3$	0.3886 ± 0.43	0.5300 ± 0.40 ↔	0.3593 ± 0.40 ↔↔	0.3719 ± 0.42 ↔↔
Próstata$_4$	0.6240 ± 0.15	0.6352 ± 0.18 ↔	0.6540 ± 0.18 ↔↔	0.7992 ± 0.13 ↑↑
Promedio	0.7978 ± 0.26	0.8128 ± 0.24	0.8044 ± 0.26	0.7751 ± 0.30

Comparación de las características seleccionadas entrenando un modelo de SVM

Otra comparación que se realizó fue la de entrenar un modelo de clasificación de SVM con los genes seleccionados por cada una de las metodologías y evaluar el promedio geométrico de la solución respecto al conjunto de entrenamiento. El objetivo de este experimento era verificar que las características seleccionadas fueran relevantes para diferenciar entre las clases de los conjuntos de datos utilizados. El modelo de SVM utilizado fue con un kernel RBF y un valor de regularización C balanceado de acuerdo al número de muestras por cada clase. La Tabla 6.14 presenta los resultados obtenidos durante dicho experimento. En ella, se puede observar que, en comparación al valor del promedio geométrico de las soluciones originales, el valor promedio de los todos experimentos sube para las metodologías de N3O, SMS-EMOA y SMS-MONEAT$_P$ y baja para la metodología de SMS-MONEAT$_Q$, emparejando dicho valor entre todas las metodologías, pero siendo esta última la que mantiene el valor más alto. La metodología que tuvo una mayor mejoría fue la basada en SMS-EMOA, lo que demuestra una diferencia favorable entre usar SVM sobre KNN para la clasificación y, por ende, una gran dependencia del modelo de clasificación utilizado. Un punto para resaltar sobre estos resultados es que para entrenar el modelo de SVM se utilizó tanto el conjunto de entrenamiento como el de validación, por lo que, con una mayor cantidad de muestras el modelo de clasificación debería tener un mejor desempeño en sí.

Utilizando la prueba post hoc de Friedman se identificó una diferencia significativa entre la metodología de SMS-EMOA y las basadas en SMS-MONEAT en los mismos dos conjuntos de datos que en el experimento anterior: Higado$_1$ y Próstata$_4$. También se demostró una diferencia significativa entre SMS-EMOA y N3O para el conjunto de datos Próstata$_4$, favoreciendo al primero de estos. Lo anterior se muestra en la Tabla 6.14 mediante el sistema de flechas. Habiendo únicamente esas diferencias significativas, SMS-EMOA sería el ganador de Condorcet para este experimento también.

Para mayor detalle sobre los resultados descritos en esta subsección se puede referir a la Figura 1.7 del Apéndice A en el cual se muestra el Box Plot con los resultados respecto al promedio geométrico obtenido por los modelos de SVM entrenados.

Tabla 6.14: Resultados obtenidos respecto al promedio geométrico de modelos SVM entrenados con las características seleccionadas en el conjunto de prueba en cada metodología (SMS-MONEAT, N3O y SMS-EMOA). El mejor valor obtenido en cada uno de los conjuntos de datos se marca de color. La simbología de flechas indica diferencias significativas entre las metodologías obtenida mediante el post hoc de la prueba de Friedman: para SMS-MONEATQ se muestra la relación con SMS-MONEAT$_P$, mientras que para N3O y SMS-EMOA se muestra la relación con las dos soluciones de SMS-MONEAT (P izquierda y Q derecha).

Conjunto de datos	SMS-MONEAT$_P$	SMS-MONEAT$_Q$	N3O	SMS-EMOA
Colon$_1$	0.9063 ± 0.13	0.9088 ± 0.13 ↔	0.8967 ± 0.14 ↔↔	0.9144 ± 0.14 ↔↔
Colon$_2$	0.9374 ± 0.20	0.9414 ± 0.12 ↔	0.9276 ± 0.20 ↔↔	0.7776 ± 0.38 ↔↔
Colon$_3$	0.9680 ± 0.04	0.9752 ± 0.03 ↔	0.9824 ± 0.03 ↔↔	0.9720 ± 0.04 ↔↔
Colon$_4$	0.8148 ± 0.13	0.8190 ± 0.15 ↔	0.8398 ± 0.12 ↔↔	0.8112 ± 0.20 ↔↔
Hígado$_1$	0.9180 ± 0.05	0.9215 ± 0.05 ↔	0.9346 ± 0.05 ↔↔	0.9517 ± 0.04 ↑↑
Hígado$_2$	0.7896 ± 0.33	0.8695 ± 0.25 ↔	0.7941 ± 0.37 ↔↔	0.7263 ± 0.42 ↔↔
Hígado$_3$	0.9758 ± 0.04	0.9846 ± 0.04 ↔	0.9957 ± 0.02 ↔↔	0.9450 ± 0.18 ↔↔
Leucemia$_1$	0.8581 ± 0.30	0.8293 ± 0.34 ↔	0.8443 ± 0.34 ↔↔	0.7715 ± 0.40 ↔↔
Leucemia$_2$	0.8911 ± 0.30	0.8920 ± 0.30 ↔	0.7858 ± 0.40 ↔↔	0.7955 ± 0.41 ↔↔
Leucemia$_3$	0.9770 ± 0.08	0.9770 ± 0.08 ↔	0.9930 ± 0.03 ↔↔	0.9894 ± 0.03 ↔↔
Leucemia$_4$	0.9179 ± 0.09	0.9282 ± 0.09 ↔	0.9500 ± 0.08 ↔↔	0.9480 ± 0.08 ↔↔
Leucemia$_5$	0.8521 ± 0.14	0.8860 ± 0.13 ↔	0.8809 ± 0.15 ↔↔	0.8793 ± 0.13 ↔↔
Mama$_1$	0.9598 ± 0.18	0.9253 ± 0.25 ↔	0.9948 ± 0.01 ↔↔	0.9267 ± 0.25 ↔↔
Mama$_2$	0.7317 ± 0.15	0.7311 ± 0.13 ↔	0.7987 ± 0.11 ↔↔	0.7945 ± 0.11 ↔↔
Mama$_3$	0.8128 ± 0.07	0.8041 ± 0.06 ↔	0.8255 ± 0.08 ↔↔	0.8373 ± 0.07 ↔↔
Mama$_4$	0.5919 ± 0.22	0.5537 ± 0.29 ↔	0.5254 ± 0.29 ↔↔	0.6125 ± 0.20 ↔↔
Próstata$_1$	0.6039 ± 0.17	0.6111 ± 0.15 ↔	0.6194 ± 0.16 ↔↔	0.6577 ± 0.16 ↔↔
Próstata$_2$	0.5843 ± 0.11	0.5843 ± 0.13 ↔	0.6461 ± 0.12 ↔↔	0.6436 ± 0.18 ↔↔
Próstata$_3$	0.4231 ± 0.39	0.4828 ± 0.42 ↔	0.3662 ± 0.38 ↔↔	0.4190 ± 0.41 ↔↔
Próstata$_4$	0.5986 ± 0.17	0.6102 ± 0.18 ↔	0.6071 ± 0.17 ↔↔	0.7426 ± 0.13 ↑↑
Promedio	0.8056 ± 0.25	0.8118 ± 0.25	0.8104 ± 0.26	0.8058 ± 0.27

Comparación del tiempo de ejecución de cada algoritmo

Los experimentos fueron ejecutados con un procesador Intel i7-6500U y 16Gb de RAM. El promedio y desviación estándar del tiempo de ejecución de los experimentos realizados en cada conjunto de datos y por cada metodología son presentados en la Tabla 6.15. En ella podemos observar que en promedio la metodología de SMS-EMOA es la que tiene un menor tiempo de ejecución, seguida por la de N3O y al final la de SMS-MONEAT. Aunque todas las metodologías realizaron el mismo número de evaluaciones, es importante recordar que las evaluaciones de las metodologías basadas en neuroevolución implican construir la ANN y evaluar en el conjunto de entrenamiento, mientras que la metodología de SMS-EMOA entrena un modelo de KNN con el conjunto de entrenamiento y posteriormente evalúa respecto a un conjunto de validación, y que ambas evaluaciones se consideraron equivalentes. Sin embargo, el orden de complejidad de construir la ANN a partir de la codificación de NEAT depende de la cantidad de nodos y conexiones que tenga cada individuo, por lo que al paso de las generaciones las evaluaciones tenderán a hacerse más costosas, mientras que para SMS-EMOA las evaluaciones tienen una complejidad constante a lo largo de las generaciones. Otra diferencia, es la codificación en sí, N3O y SMS-MONEAT utilizan la codificación de NEAT mientras que la metodología de SMS-EMOA utiliza una codificación binaria. Y esta diferencia de codificación también tiene un impacto en el tiempo de ejecución debido a que de acuerdo a la codificación son los operadores evolutivos utilizados, siendo más complejos los operadores de neuroevolución sobre los de codificación binaria. Los operadores utilizados en SMS-MONEAT y N3O dependen del número de nodos y conexiones de los individuos a lo largo de las generaciones. Por otro lado, para SMS-EMOA se utilizó cruza en un punto el cual tiene una complejidad constante y el operador de mutación binario el cual depende del número total de características del conjunto de datos de cada experimento. También es importante resaltar, que, puesto que en la codificación binaria el tamaño del genoma de cada individuo depende del número de características del conjunto de datos en estudio, la cantidad de memoria utilizada para esta codificación es mayor que los individuos con una codificación de NEAT.

Una prueba post hoc de Friedman fue realizada para comparar los resultados respecto al tiempo de ejecución de todos los experimentos y la metodología de Condorcet fue utilizada para identificar al algoritmo ganador, que para esta métrica fue la basada en SMS-EMOA. La Tabla 6.16 muestra los resultados de la metodología de Condorcet. Se determinó diferencia significativa a favor de SMS-EMOA en comparación a la metodología de SMS-MONEAT (todos menos Colon_2 y Mama_3). También se encontró diferencia significativa favorable para SMS-EMOA al comparar contra la metodología de N3O en 9 conjuntos de datos (Colon_3, Higado_1, Leucemia_1, Leucemia_2, Leucemia_5, Mama1, Próstata_1 y Próstata_3). Finalmente, al comparar la metodología de SMS-MONEAT y N3O, se encontró diferencia significativa a favor de N3O en 18 conjuntos de datos (todos menos Colon_3 e Higado_1) y 1 a favor de SMS-MONEAT (Colon_3). Lo anterior se muestra en la Tabla 6.14 mediante el sistema de flechas.

Para mayor detalle sobre los resultados descritos en esta subsección se puede referir a las Figuras 1.8 y 1.9 del Apéndice A en el cual se muestran Box Plots con los resultados respecto al tiempo de ejecución de las diferentes metodologías.

Tabla 6.15: Resultados obtenidos respecto al tiempo de ejecución para cada conjunto de prueba en cada metodología (SMS-MONEAT, N3O y SMS-EMOA). El mejor valor obtenido en cada uno de los conjuntos de datos se marca de color. La simbología de flechas indica diferencias significativas entre las metodologías obtenida mediante el post hoc de la prueba de Friedman: para SMS-MONEATQ se muestra la relación con SMS-MONEAT$_P$, mientras que para N3O y SMS-EMOA se muestra la relación con las dos soluciones de SMS-MONEAT (P izquierda y Q derecha).

Conjunto de datos	SMS-MONEAT	N3O	SMS-EMOA
Colon_1	350.9027 ± 311.40	126.6473 ± 19.54↑	121.9484 ± 3.30 ↑
Colon_2	255.8082 ± 122.90	160.3370 ± 30.93↑	174.3896 ± 7.52 ↔
Colon_3	487.6789 ± 189.44	562.4455 ± 60.65↓	435.8497 ± 3.96 ↑
Colon_4	538.7949 ± 1098.91	151.7138 ± 17.30↑	143.7420 ± 5.35 ↑
Hígado_1	601.8749 ± 135.42	538.6731 ± 34.01↔	399.5623 ± 10.16 ↑
Hígado_2	380.7000 ± 154.14	123.4471 ± 17.66↑	116.8963 ± 6.19 ↑
Hígado_3	385.5413 ± 212.64	224.1851 ± 18.77↑	239.4493 ± 7.56 ↑
Leucemia_1	322.4098 ± 152.35	134.8021 ± 26.58↑	93.4433 ± 3.42 ↑
Leucemia_2	387.4495 ± 139.98	115.8817 ± 24.78↑	90.7602 ± 4.14 ↑
Leucemia_3	457.5584 ± 167.12	198.4296 ± 37.70↑	163.8477 ± 3.36 ↑
Leucemia_4	453.5147 ± 419.58	237.5982 ± 19.15↑	226.4355 ± 5.09 ↑
Leucemia_5	308.3891 ± 91.78	110.0158 ± 12.87↑	87.4535 ± 3.05 ↑
Mama_1	422.2620 ± 100.20	209.7617 ± 40.23↑	185.6376 ± 4.22 ↑
Mama_2	294.9805 ± 128.25	143.4380 ± 20.37↑	136.4652 ± 10.45 ↑
Mama_3	607.9056 ± 541.11	503.5966 ± 34.53↑	391.8875 ± 5.20 ↔
Mama_4	386.9707 ± 915.51	104.9933 ± 14.85↑	95.2658 ± 3.26 ↑
Próstata_1	388.5974 ± 647.47	123.1197 ± 17.78↑	105.7421 ± 2.87 ↑
Próstata_2	218.9941 ± 50.88	112.7325 ± 12.39↑	102.4727 ± 2.68 ↑
Próstata_3	198.2223 ± 88.42	103.5312 ± 15.89↑	72.3176 ± 1.71 ↑
Próstata_4	273.8947 ± 89.23	127.4118 ± 20.98↑	114.1458 ± 6.54 ↑
Promedio	386.1225 ± 414.84	205.6381 ± 146.60	174.8856 ± 108.23

Tabla 6.16: Comparación entre las metodologías SMS-MONEAT, N3O y SMS-EMOA utilizando la metodología de Condorcet y los resultados obtenidos del post hoc de la prueba de Friedman respecto a los resultados del tiempo de ejecución.

	SMS-MONEAT	N3O	SMS-EMOA	**Total**
SMS-MONEAT	X	1	0	**1**
N3O	18	X	0	**18**
SMS-EMOA	9	18	X	**27**

Comparación estructural de las ANNs generadas

Por último, se compararon las estructuras de las ANNs generadas por los algoritmos SMS-MONEAT y N3O. Podemos describir las ANNs utilizando como características: el número de entradas, el número de nodos ocultos y el número de conexiones. Pero también es importante entender que no todos los nodos ni conexiones participan en la ANN debido a que hay conexiones que están deshabilitadas. Las Tablas 6.17, 6.18 y 6.19, presentan los detalles de las soluciones seleccionadas durante los experimentos para SMS-MONEAT$_P$, SMS-MONAET$_Q$ y N3O, respectivamente.

Si se comparan las estructuras de las soluciones seleccionadas entre la población final y el archivo externo de SMS-MONEAT, se puede observar que las soluciones de SMS-MONEAT$_P$ tienen un mayor número de nodos y conexiones (activas e inactivas) que las de SMS-MONEAT$_Q$. Tomando en cuenta que, a lo largo de las generaciones, las estructuras de las ANNs se van haciendo más complejas, esto podría implicar que las soluciones que son seleccionadas del archivo externo no fueron generadas en las últimas generaciones. A su vez, mantener soluciones de generaciones pasadas podría ser razón de que las soluciones de SMS-MONEAT$_Q$ hayan demostrado generalizar mejor ante nuevas soluciones en los experimentos realizados, lo cual se muestra en los valores promedio de todos los experimentos sobre la métrica de entropía cruzada y promedio geométrico en las Tablas 6.8 y 6.13, respectivamente.

Por otro lado, si se observa la información de las soluciones generadas por N3O, se puede observar que hay un mayor número de nodos de entrada, lo cual es esperado, ya que como se ha mencionado anteriormente, el número de características seleccionadas no es un objetivo para minimizar para esta metodología a diferencia de SMS-MONEAT. Tener un mayor número de entradas también impacta a un mayor número de conexiones, que, en comparación, N3O tiene 21.05 conexiones en promedio de todos los experimentos, siendo mayor al valor de 12.87 que pertenece a SMS-MONEAT$_P$. Por otro lado, ambas metodologías mantienen un número similar de nodos ocultos, lo cual también hace sentido, ya que se utilizó un valor similar al operador de mutación para agregar un nuevo nodo.

Un detalle importante es el porcentaje de nodos y conexiones activas entre cada metodología. En promedio de todos los experimentos realizados mediante N3O, el 40.12 % de los nodos entrada, el 64.71 % de los nodos ocultos y el 48.12 % de las conexiones eran activos. Estos porcentajes son mayores a los de SMS-MONEAT$_P$ con 24.68 % para los nodos entrada, 34.61 % para los nodos ocultos y 30.99 % para las conexiones, y también para las soluciones de SMS-MONEAT$_Q$ con 35.00 % para los nodos de entrada, 42.37 % para los nodos ocultos y 39.54 % para las conexiones. Que haya estructura inactiva en la ANN depende principalmente del operador de cruza, el cual permite desactivar conexiones que a su vez puede eliminar el camino de ciertos nodos hacia el nodo de salida. Asimismo, para el caso en que las soluciones tienen la misma aptitud, el operador de cruza permite incluir nodos y conexiones de forma aleatoria de ambos padres, lo cual hace al operador más disruptivo ya que puede agregar es-

tructuras que no participen de manera activa hacia el nodo de salida. Tomando en cuenta que SMS-MONEAT utiliza el ranking de los frentes de Pareto para decidir al padre más apto durante la cruza, es más probable que ambos padres tengan la misma aptitud en comparación a N3O, por lo que esta podría ser la principal razón de que las soluciones tengan más nodos y conexiones inactivos.

Tabla 6.17: Información topológica de las ANNs de las soluciones seleccionadas en cada conjunto de datos de la metodología SMS-MONEAT$_P$.

Conjunto de datos	Entradas		Nodos ocultos		Conexiones	
	Totales	Activos	Totales	Activos	Totales	Activas
$Colon_1$	8.60± 4.46	2.53± 1.22	14.30± 26.14	6.97± 7.04	40.17± 48.30	19.07± 17.22
$Colon_2$	7.43± 8.27	2.00± 1.23	12.50± 20.70	3.33± 5.11	34.50± 42.90	9.83± 12.36
$Colon_3$	9.97± 16.79	2.10± 0.96	15.30± 30.18	2.40± 4.72	45.23± 83.60	7.70± 11.93
$Colon_4$	14.57± 21.87	2.63± 0.89	18.97± 34.38	2.17± 4.03	57.73± 92.78	7.93± 9.76
$Hígado_1$	11.17± 18.11	2.10± 0.88	12.97± 18.38	1.87± 2.46	40.67± 57.05	6.10± 5.89
$Hígado_2$	14.60± 12.20	2.83± 1.18	17.70± 20.70	4.87± 5.78	55.57± 57.79	13.70± 13.87
$Hígado_3$	4.10± 3.77	1.53± 0.97	9.73± 23.76	1.80± 4.37	24.93± 48.40	5.73± 11.10
$Leucemia_1$	12.27± 12.30	4.27± 10.41	9.90± 10.01	6.23± 6.47	36.40± 32.93	19.90± 25.10
$Leucemia_2$	11.73± 6.18	3.07± 1.26	11.20± 6.73	7.10± 6.26	40.00± 18.35	19.87± 15.15
$Leucemia_3$	8.93± 4.62	2.53± 1.14	13.07± 11.81	6.07± 5.27	40.40± 27.33	16.67± 12.75
$Leucemia_4$	12.87± 24.51	3.47± 6.07	19.73± 37.58	3.73± 5.19	53.53± 99.57	11.77± 14.90
$Leucemia_5$	14.43± 5.18	3.30± 1.29	10.63± 8.55	5.87± 4.97	40.67± 19.10	17.60± 13.23
$Mama_1$	11.93± 6.75	2.90± 1.35	12.90± 8.32	6.13± 5.38	43.60± 23.42	17.40± 13.63
$Mama_2$	12.70± 9.40	3.67± 1.47	19.00± 20.88	4.07± 4.81	55.10± 54.22	12.77± 11.44
$Mama_3$	9.60± 17.57	2.47± 1.20	13.00± 35.21	1.20± 3.14	39.67± 96.67	5.03± 7.47
$Mama_4$	21.13± 29.19	3.23± 1.70	18.63± 44.19	2.87± 4.19	66.13± 131.92	10.80± 12.09
$Próstata_1$	17.77± 15.18	3.53± 1.59	15.57± 23.93	3.87± 4.34	55.87± 77.94	13.17± 11.17
$Próstata_2$	15.57± 6.56	4.07± 1.51	9.20± 8.15	3.60± 4.38	38.40± 22.50	13.00± 10.60
$Próstata_3$	14.10± 5.91	2.53± 1.17	7.70± 6.42	5.87± 5.44	33.10± 16.99	16.23± 12.44
$Próstata_4$	10.97± 8.09	2.77± 1.45	8.83± 7.57	4.30± 4.96	31.90± 21.64	13.10± 13.24
Promedio	12.22± 14.13	2.88± 2.98	13.54± 23.10	4.21± 5.26	43.68± 62.57	12.87± 13.89

Tabla 6.18: Información topológica de las ANNs de las soluciones seleccionadas en cada conjunto de datos de la metodología SMS-MONEAT$_Q$.

Conjunto de datos	Entradas		Nodos ocultos		Conexiones	
	Totales	Activos	Totales	Activos	Totales	Activas
Colon$_1$	8.50± 4.44	3.07± 1.57	11.53± 16.80	6.17± 5.97	33.97± 32.16	17.53± 14.91
Colon$_2$	4.77± 2.40	2.17± 0.91	6.53± 11.19	2.93± 4.28	19.30± 23.16	9.20± 10.13
Colon$_3$	5.67± 3.84	2.37± 1.03	7.93± 14.28	2.20± 4.63	23.83± 34.45	7.50± 11.71
Colon$_4$	11.00± 24.05	2.90± 1.06	13.33± 36.15	2.37± 3.33	42.97± 113.35	8.90± 8.64
Hígado$_1$	8.60± 11.15	2.33± 0.92	8.93± 13.95	1.80± 2.37	28.93± 39.24	6.17± 5.71
Hígado$_2$	9.43± 4.38	2.83± 1.21	11.27± 9.91	4.77± 5.70	35.57± 25.09	13.43± 13.84
Hígado$_3$	4.27± 3.94	2.13± 1.66	5.50± 11.06	1.70± 4.17	16.57± 25.70	6.20± 10.73
Leucemia$_1$	10.97± 11.97	4.37± 10.23	8.00± 8.96	5.13± 6.13	30.43± 31.93	17.20± 25.13
Leucemia$_2$	9.87± 6.71	3.50± 3.54	8.87± 6.88	6.13± 5.94	31.73± 20.11	18.13± 15.97
Leucemia$_3$	8.90± 4.24	2.87± 1.28	12.27± 9.92	5.83± 4.81	38.33± 22.95	16.57± 12.12
Leucemia$_4$	7.30± 7.91	4.23± 6.70	10.07± 13.83	3.37± 4.49	27.87± 32.17	11.83± 14.61
Leucemia$_5$	14.43± 5.49	3.53± 1.14	9.77± 5.78	6.00± 4.96	39.17± 17.71	18.03± 13.11
Mama$_1$	10.97± 4.40	3.13± 1.28	12.30± 7.74	6.43± 5.33	41.40± 21.89	18.13± 13.37
Mama$_2$	10.07± 6.86	4.17± 1.37	12.83± 16.57	3.47± 4.26	40.97± 48.64	12.17± 10.15
Mama$_3$	9.40± 17.80	2.90± 1.18	9.83± 32.40	1.03± 2.75	31.43± 89.72	5.20± 6.66
Mama$_4$	14.90± 16.14	3.60± 1.54	9.83± 20.87	2.23± 3.37	37.53± 59.20	9.40± 10.16
Próstata$_1$	11.30± 5.23	3.83± 1.68	8.30± 8.85	3.77± 5.14	30.83± 22.03	13.33± 13.57
Próstata$_2$	13.00± 5.18	4.60± 1.43	6.67± 5.23	3.20± 4.00	29.57± 15.26	12.37± 9.26
Próstata$_3$	11.67± 6.29	2.77± 1.41	4.97± 4.96	3.70± 3.93	23.70± 13.45	11.77± 9.46
Próstata$_4$	9.77± 4.68	3.43± 1.76	8.03± 7.49	4.33± 5.12	28.53± 19.41	13.60± 13.22
Promedio	9.74± 9.82	3.24± 3.14	9.34± 15.41	3.83± 4.86	31.63± 43.15	12.33± 13.25

Tabla 6.19: Información topológica de las ANNs de las soluciones seleccionadas en cada conjunto de datos de la metodología N3O.

Conjunto de datos	Entradas		Nodos ocultos		Conexiones	
	Totales	Activos	Totales	Activos	Totales	Activas
$Colon_1$	29.30± 21.17	13.80± 14.26	7.70± 4.32	4.90± 3.58	48.87± 28.12	24.97± 20.66
$Colon_2$	28.40± 19.43	16.00± 16.25	7.07± 4.35	5.00± 4.27	46.53± 27.34	28.00± 22.74
$Colon_3$	27.53± 33.67	11.50± 20.69	4.93± 4.53	2.70± 3.10	40.10± 42.16	17.73± 25.91
$Colon_4$	20.23± 9.05	9.67± 5.23	5.33± 3.71	3.53± 3.19	33.63± 15.36	18.00± 11.55
$Hígado_1$	28.87± 31.49	6.73± 4.68	6.20± 2.98	3.00± 2.72	44.93± 36.34	13.47± 10.33
$Hígado_2$	22.83± 10.65	8.40± 3.81	8.27± 3.02	5.43± 3.14	43.47± 15.87	20.73± 10.26
$Hígado_3$	21.83± 13.55	9.50± 6.43	6.07± 4.69	4.60± 4.15	35.87± 21.08	19.87± 14.21
$Leucemia_1$	21.80± 12.08	8.97± 7.64	6.97± 4.16	4.73± 3.80	39.17± 20.45	19.63± 14.78
$Leucemia_2$	26.10± 20.66	9.07± 5.27	8.33± 3.99	6.27± 3.50	46.60± 27.66	23.43± 11.95
$Leucemia_3$	27.40± 22.63	11.90± 17.64	8.83± 4.98	5.77± 4.81	49.13± 31.20	25.07± 25.91
$Leucemia_4$	27.27± 25.29	7.97± 4.79	7.27± 4.03	4.27± 3.87	44.70± 28.98	17.40± 12.54
$Leucemia_5$	26.10± 11.49	10.03± 6.57	10.03± 3.27	6.50± 2.80	50.80± 16.16	24.50± 10.82
$Mama_1$	36.00± 38.76	13.37± 14.87	9.33± 3.66	6.13± 3.81	59.50± 45.28	27.43± 20.66
$Mama_2$	23.73± 11.21	11.13± 4.49	8.13± 3.95	5.40± 3.10	44.83± 19.43	24.07± 11.23
$Mama_3$	19.60± 17.10	7.20± 3.81	3.53± 3.09	1.97± 2.54	29.23± 22.40	11.93± 9.68
$Mama_4$	21.70± 7.27	9.23± 4.20	5.40± 3.45	3.43± 2.60	36.40± 12.06	17.80± 9.30
$Próstata_1$	26.67± 13.67	10.40± 3.64	6.97± 3.31	4.47± 3.19	44.93± 17.71	20.73± 9.22
$Próstata_2$	23.57± 8.98	9.80± 3.55	7.17± 3.36	4.00± 2.83	42.17± 14.33	18.87± 7.97
$Próstata_3$	27.80± 10.15	11.27± 6.74	8.77± 3.73	7.07± 4.02	49.77± 13.38	27.40± 12.13
$Próstata_4$	21.23± 14.77	7.30± 3.71	8.90± 3.19	5.80± 2.82	43.20± 20.79	19.97± 8.32
Promedio	25.40± 19.75	10.16± 9.67	7.26± 4.10	4.75± 3.64	43.69± 25.98	21.05± 15.50

6.2.3. Compendio final de los resultados

En las subsecciones anteriores se presentaron las comparaciones con diferentes métricas entre el desempeño de las metodologías propuestas basadas en el algoritmo SMS-MONEAT contra el algoritmo de N3O y la metodología de SMS-EMOA/KNN. En esta subsección se hará un análisis general de dichas comparaciones.

Primero al comparar las soluciones SMS-MONEAT$_P$ y SMS-MONEATQ respecto a los objetivos a optimizar los cuales fueron la función de entropía cruzada y el número de genes seleccionados no se detectaron diferencias significativas, sin embargo, las soluciones de SMS-MONEAT$_P$ tuvieron un menor número de características mientras que las de SMS-MONEAT$_Q$ un promedio menor de entropía cruzada al igual que un promedio mayor al evaluar con el promedio geométrico. Esta diferencia se atribuye a la metodología de especiación del archivo externo, ya que puede mantener soluciones subóptimas con un mayor número de características pero que puedan generalizar mejor. Al comparar las poblaciones utilizando el hipervolumen, se detectó diferencia significativa en 6 conjuntos de datos favorable a la población del archivo externo lo cual refleja la mayor diversidad de soluciones en dicha población.

N3O fue el algoritmo ganador de Condorcet al comparar utilizando la entropía cruzada, sin embargo, no se demostró diferencia significativa respecto a las soluciones de SMS-MONEAT$_P$ y sólo se demostró en un conjunto de datos respecto a SMS-MONEAT$_Q$. Por otro lado, sí hubo diferencia significativa en todos los conjuntos de datos a favor de las soluciones de las metodologías de SMS-MONEAT respecto al número de características seleccionadas. Con lo anterior, se puede notar que la ventaja de SMS-MONEAT sobre N3O está relacionada con que la capacidad del primero de optimizar más de un objetivo a la vez.

Por otro lado, SMS-EMOA sí incluía como objetivo la minimización del número de características al igual que la metodología de SMS-MONEAT. Sin embargo, las metodologías basadas en SMS-MONEAT mostraron un desempeño superior tanto al comparar con la entropía cruzada como en el número de características. Respecto a la entropía cruzada se demostró diferencia significativa en la mayoría de los conjuntos de datos a favor de las metodologías de SMS-MONEAT sobre la de SMS-EMOA y para el número de características se demostró diferencia significativa en más conjuntos de datos a favor de SMS-MONEAT$_P$ sobre SMS-EMOA que viceversa. Esta diferencia de desempeño de atribuye a que una dependencia de esta metodología sobre el modelo de clasificación que se utilice, siendo que cuando se cuándo se realizó la comparación entrenando un modelo SVM con las características seleccionadas los resultados de la metodología de SMS-EMOA mejoraron respecto al promedio geométrico. Además, SMS-EMOA mostró ventaja en esta métrica sobre las metodologías de SMS-MONEAT, demostrando diferencia significativa favorable en 2 conjuntos de datos. Se observó que ambos conjuntos de datos tenían un valor mayor de número de muestras respecto al promedio de todos los conjuntos de datos utilizados, y esto podría ser razón de que SMS-EMOA tenga un mejor desempeño,

puesto que al usar validación cruzada durante la evaluación pueda encontrar soluciones que generalicen mejor.

Finalmente, también se realizó una comparación respecto al tiempo de ejecución de las 3 metodologías, siendo la de SMS-MONEAT la de mayor tiempo de ejecución. Sin embargo, el mayor tiempo de ejecución se justifica con un mejor desempeño respecto a la entropía cruzada y número de características seleccionadas en comparación a las otras metodologías.

Capítulo 7

Conclusiones y trabajo a futuro

We can only see a short distance ahead, but we can see plenty there that needs to be done.

Alan Turing

En este capítulo se presentan las conclusiones obtenidas de esta tesis (Sección 7.1) y diversas propuestas para continuar este trabajo a futuro (Sección 7.2).

7.1. Conclusiones

En el presente trabajo se presentó un nuevo algoritmo multiobjetivo basado en neuroevolución, el cual tomó la codificación de NEAT y los operadores evolutivos de N3O junto con el uso del hipervolumen como estrategia de optimización multiobjetivo, llamado SMS-MONEAT. El algoritmo se implementó junto con una metodología para la selección de genes y clasificación de microarreglos. Dicha metodología iniciaba con un método de filtro basado en la prueba H de KW para la reducción de características, con el cual se removieron en promedio el 75.31 % de las características durante los experimentos realizados. Posterior a este paso, se normalizaban los datos utilizando la función $MinMaxScaler()$ de Scikit-Learn y se ejecutaba el algoritmo. El algoritmo incluye un archivo externo con un método de especiación basado en las diferentes combinaciones de características seleccionadas de los individuos generados, favoreciendo a la diversidad de soluciones almacenadas. Al final de la ejecución del algoritmo, una metodología para seleccionar una solución de la población fue diseñada basada en una suma ponderada de pesos que incluía el valor de entropía cruzada y el promedio geométrico evaluando en el conjunto de datos utilizado durante el entrenamiento y en un conjunto de validación.

El desempeño del algoritmo se probó en 20 conjuntos de datos y en 30 ex-

perimentos de validación cruzada estratificada a 10 capas y comparada contra el algoritmo N3O y contra otra metodología multiobjetivo utilizando el algoritmo de SMS-EMOA y el modelo de clasificación de KNN. Cabe resaltar que no se encontró en la literatura que el algoritmo de SMS-EMOA haya sido utilizado previamente para la selección de genes y clasificación de microarreglos, pero al ser base fundamental de SMS-MONEAT y utilizar el hipervolumen para la optimización del frente de Pareto, hacía sentido utilizarlo para poder comparar su desempeño en dicho indicador. En dichos experimentos realizados se utilizaron 6 métricas para comparar las soluciones obtenidas mediante N3O, SMS-EMOA, SMS-MONEAT y del archivo externo de este último, las cuales fueron una función de entropía cruzada, el número de características seleccionadas, el hipervolumen obtenido por las dos métricas anteriores, el promedio geométrico de las soluciones generadas, el promedio geométrico de un modelo de clasificación SVM entrenado con las características seleccionadas y el tiempo de ejecución del algoritmo. Se utilizó la prueba no paramétrica de Friedman para determinar diferencias significativas en el desempeño de las metodologías y se utilizó la metodología de Condorcet para definir a un ganador en cada una de las métricas. Respecto a la entropía cruzada, la cual fue la única que fue función objetivo de todas las metodologías, el ganador de Condorcet fue N3O, sin embargo, 12 de 13 victorias fueron hacia la metodología de SMS-EMOA y la otra hacia las soluciones obtenidas del archivo, es decir, no hubo diferencia significativa contra las soluciones de la población final de SMS-MONEAT. Del mismo, modo las victorias de SMS-MONEAT fueron hacia la metodología de SMS-EMOA, con lo que podemos concluir que los modelos de clasificación generados por los algoritmos de N3O y SMS-MONEAT, que son ANN generadas automáticamente por estos algoritmos, tuvieron un mejor desempeño en más de la mitad de los conjuntos de datos que los de KNN generados por la metodología de SMS-EMOA respecto a esta función de pérdida.

La segunda métrica para analizar es la cantidad de características seleccionadas, de la cual el ganador de Condorcet fue SMS-MONEAT, en específico las soluciones tomadas de la población final, pero en segundo se ubicaron las seleccionadas del archivo externo. Al tomar el número de características seleccionadas como objetivo, se obtiene una reducción significativa, la cual fue demostrada en los 20 conjuntos de datos utilizados al comparar las soluciones de SMS-MONEAT y las de N3O. Esto es relevante, ya que estas soluciones, como se mencionó en el párrafo anterior, no mostraron una diferencia significativa respecto a la función de pérdida, es decir, las soluciones de SMS-MONEAT mantienen su calidad respecto a la entropía cruzada, pero con un menor número de características seleccionadas. Lo anterior, se observa con mayor claridad al utilizar el hipervolumen para comparar las metodologías, siendo que SMS-MONEAT tuvo el mejor desempeño en está métrica sobre los otros dos algoritmos, ya que, las soluciones de N3O no competían en el número de características y SMS-EMOA no competía con la entropía cruzada. Adicionalmente, al comparar las soluciones obtenidas por SMS-MONEAT de la población final con las del archivo externo en el hipervolumen, se notó una diferencia significativa en 6 conjuntos de datos a favor de las del archivo externo, esto puede ser debido a

dos razones: mantener una mayor diversidad de soluciones respecto al conjunto de características seleccionadas y mantener soluciones que con un menor ajuste al conjunto de entrenamiento. Ambas razones podrían permitir generalizar mejor hacia el conjunto de prueba, lo cual se observa con mayor detalle al evaluar ante el promedio geométrico, teniendo un valor promedio mayor en total de todos los experimentos, pero sin una diferencia significativa detectada entre cada experimento.

Para las pruebas utilizando el promedio geométrico, pese a que SMS-EMOA tuvo un promedio geométrico total menor que sus competidores, en la prueba de Condorcet resultó como ganador, ganándole en dos conjuntos de datos a las soluciones de SMS-MONEAT. Se identificó que en dichos conjuntos de datos tienen un número relativamente alto de muestras, GSE14520_U133A con 357 (el conjunto de datos utilizado con más muestras) y Singh et al. [96] con 136 (el quinto conjunto utilizado con más muestras), siendo esta una posible razón del desempeño superior de la metodología de SMS-EMOA, puesto a que utiliza un proceso de validación cruzada para evaluar cada individuo. Adicionalmente, utilizar el modelo de SVM y las características seleccionadas dieron el mismo resultado al comparar las soluciones de SMS-EMOA con las de SMS-MONEAT, lo cual implica que esta diferencia proviene de las características seleccionadas más que del modelo de clasificación. Cabe resaltar, que, respecto al número de características seleccionadas, también se demostró una diferencia significativa entre ambos algoritmos a favor de SMS-MONEAT. Entrenar al final un modelo de SVM favoreció particularmente a la metodología de SMS-EMOA respecto al promedio geométrico obtenido, y en menor medida a N3O. Para el caso de SMS-MONEAT el valor del promedio geométrico bajo para las soluciones de la población final y tuvo un efecto mínimo positivo sobre las del archivo externo. Esto podría implicar que los modelos de clasificación generados tienen un desempeño competitivo ante otros modelos de clasificación convencionales como SVM.

El promedio el tiempo de ejecución de SMS-MONEAT fue mayor que los otros dos algoritmos comparados. SMS-EMOA tuvo el menor tiempo de ejecución en la mayoría de los experimentos, esto puede ser por la codificación genética utilizada, ya que los operadores evolutivos de una codificación binaria tienen una menor complejidad que los de la codificación de NEAT. Por otro lado, al comparar N3O y SMS-MONEAT, tener un objetivo extra en este último puede implicar a su vez un mayor costo computacional y a su vez un mayor tiempo de ejecución. Además, el archivo externo aumenta el costo computacional de su ejecución. Sin embargo, tomando en cuenta los resultados obtenidos de SMS-MONEAT, un mayor tiempo de ejecución se compensaría con un mayor hipervolumen en las soluciones encontradas.

Finalmente, también se comparó la topología de las ANNs generadas por N3O y SMS-MONEAT, con lo que se observó un mayor número de conexiones activas del primero sobre el segundo, que se atribuyó al mayor número de nodos entradas debido a que N3O no minimiza el número de características seleccionadas. También se observó un mayor porcentaje de nodos y conexiones activas de las soluciones de N3O sobre las de SMS-MONEAT, esto podría ser debido a

que N3O utiliza la función de pérdida para definir al padre más apto mientras que SMS-MONEAT el ranking de los frentes de Pareto, por lo que, habrá una mayor probabilidad que los padres tengan la misma aptitud en SMS-MONEAT y el operador de cruza sea más disruptivo generando nuevos individuos con un mayor número de conexiones deshabilitadas o no conectadas entre sí.

Respecto al archivo externo implementado para SMS-MONEAT basado en especiación, sólo se encontró diferencia significativa con el conjunto de soluciones de la población final del mismo con el indicador del hipervolumen, siendo favorable para las del archivo externo en 6 conjuntos de datos. El archivo externo busca impulsar la diversidad de soluciones separándolas en diferentes especies de acuerdo a las características seleccionadas, lo que le ocasiona que almacene soluciones dominadas, favoreciendo la diversidad sobre la proximidad del frente. Por otro lado, la población final (SMS-MONEAT$_P$) debería tener un mayor número de soluciones que aproximen el frente de Pareto puesto que el algoritmo busca optimizar el hipervolumen de la población y las soluciones no dominadas no contribuyen al hipervolumen. Sin embargo, el frente de Pareto cambia entre evaluar al conjunto de entrenamiento, al de validación o al de prueba. Por lo que, impulsar la diversidad de soluciones podría permitir mantener soluciones que tengan una mejor generalización ante nuevos datos, siendo esta la razón con la que se justifica el desempeño favorable del uso del archivo externo.

Para el problema de microarreglos, encontrar un subconjunto mínimo de genes relevantes es importante para construir modelos de clasificación que diagnostiquen y detecten enfermedades de manera confiable. Incluir al número de genes como objetivo a minimizar a la par de optimizar al modelo de clasificación permite obtener una mayor diversidad de soluciones, lo cual podría beneficiar a modelos que generalicen mejor ante nuevas muestras. Adicionalmente, el uso de neuroevolución para generar ANNS reduce el trabajo manual y complejidad de diseñar la estructura de la ANN o elegir otro modelo de clasificación. En específico, la codificación de NEAT procura generar ANNs con una topología mínima, con un bajo número de nodos y conexiones, pero optimizando su desempeño, lo cual se comprobó con los resultados obtenidos en este trabajo, al competir contra otros modelos de clasificación como KNN o SVM.

7.2. Trabajo a futuro

A continuación, se proponen algunas ideas que se podrían realizar para continuar con el trabajo de investigación presentado:

- Comparar el desempeño de SMS-MONEAT utilizando diferentes métodos de filtros con puntaje, tanto como metodología de reducción de características previa a la ejecución del algoritmo y para asignar valores a cada característica que mejoren el funcionamiento del operador de mutación para agregar nuevas características (actualmente utiliza el valor p obtenido por la prueba de KW).

- Inspirados en el funcionamiento del algoritmo genético adaptativo, se propone utilizar valores de probabilidad de cruza y mutación que se ajusten automáticamente durante la ejecución de SMS-MONEAT. La cantidad de nodos y conexiones de las redes incrementa a lo largo de las generaciones, por lo que la cantidad de soluciones inválidas que se generan por el operador de cruza también incrementa, es por ello, que adaptar la probabilidad de cruza podría mejorar el desempeño del algoritmo. Además, se podrían ajustar los valores de probabilidad de ciertos operadores de mutación tanto los operadores de pesos como estructurales y de este modo, tener un mejor control sobre la exploración y explotación durante la ejecución.

- Incluir en SMS-MONEAT un operador de mutación que realice refinamientos locales como lo haría un algoritmo memético. Este operador podría realizar una iteración de retropropagación para refinar el valor de los pesos de las conexiones de la ANN. Otra opción sería aplicar retropropagación a los mejores individuos de la población al final de la ejecución de SMS-MONEAT.

- Realizar pruebas utilizando otros indicadores de calidad como método de reducción de población en lugar del hipervolumen para comparar su desempeño.

- Probar una codificación genética diferente junto con el algoritmo de SMS-EMOA como la del algoritmo de arquitecturas unificadas de neuroevolución con diversidad espectral(SUNA por sus siglas en inglés «Spectrum-diverse Unified Neuroevolution Arquitecture») [97], con la cual se puede generar arquitecturas diferentes a NEAT y ha demostrado un desempeño superior en ciertos experimentos sobre ambientes de prueba para tareas de aprendizaje por refuerzo.

- Realizar pruebas utilizando los valores de falso negativos y falsos positivos como objetivos en lugar de la función de entropía cruzada, de este modo, se tendrían 3 objetivos a optimizar, y observar el impacto de esta modificación en la precisión de los modelos generados y el tiempo de ejecución del algoritmo.

Apéndices

A. Gráficas de los resultados experimentales sobre conjuntos de datos de microarreglos

En este apéndice se presentan Box Plots de los resultados experimentales obtenidos por las diferentes metodologías utilizadas en el presente proyecto. Las figuras presentadas hacen referencia a la información presentada en tablas en la Sección 6.2.

La Figura 1.1 muestra los resultados obtenidos respecto a la entropía cruzada para las soluciones de N3O, SMS-EMOA y SMS-MONEAT. Además, se agregó una gráfica con únicamente las soluciones de N3O y SMS-MONEAT presentada en la Figura 1.2, debido a que las soluciones de SMS-EMOA tienen valores muy altos en esta métrica en comparación a las demás metodologías y permitir ver con mayor claridad el desempeño de estas.

Respecto al número de características seleccionadas, se presenta la Figura 1.3, en ella se observan valores de media y variación estándar mayor de N3O sobre sus competidores.

Los resultados obtenidos mediante el hipervolumen para las diferentes metodologías se muestran en la Figura 1.4. En ella, se puede observar una gran variación de las metodologías de N3O y SMS-EMOA, siendo esta última la más pronunciada. Lo anterior, se relaciona a las gráficas de la entropía cruzada y del número de características seleccionadas, debido a que son las dos métricas utilizadas para calcular el hipervolumen. Además, se graficó los resultados de SMS-MONEAT$_P$ y SMS-MONEAT$_Q$ por separado para observar sus diferencias y se muestra en la Figura 1.5. En esta gráfica también podemos observar que las soluciones del archivo externo tienen una variación menor que las de la población final, lo cual podría ser debido a una mejor generalización de los datos debido a una mayor diversidad de las soluciones almacenadas.

El promedio geométrico también fue utilizado como métrica para comparar las métricas. La Figura 1.6 muestra los resultados obtenidos utilizando los modelos generados por cada una de las metodologías, mientras que la Figura 1.7 presenta el promedio geométrico obtenido mediante modelos SVM entrenados con las características seleccionadas en cada experimento. En estas gráficas se observa un comportamiento indiferente al modelo de clasificación utilizado, sin embargo, hay una menor variación de resultados al reentrenar el modelo de

SVM, sobre todo para la metodología de SMS-EMOA. Aunque, como se mencionó en la Subsección 6.2.2, el modelo de SVM fue entrenado con el conjunto de entrenamiento, así como el de validación, por lo que hace sentido que las soluciones muestren un mejor desempeño.

Finalmente, la Figura 1.8 muestra los Box Plots del tiempo de ejecución de los experimentos realizados. En esta se observa que SMS-MONEAT tiene el mayor tiempo de ejecución de las 3 metodologías, sin embargo, se debe tomar en cuenta que esta metodología construye dos poblaciones de soluciones debido a incluir el archivo externo basado en especiación. La metodología de N3O y SMS-EMOA se graficaron también por separado para observar las diferencias entre ambos y se muestra en la Figura 1.9, donde se observa una menor variabilidad de tiempo para el algoritmo de SMS-EMOA.

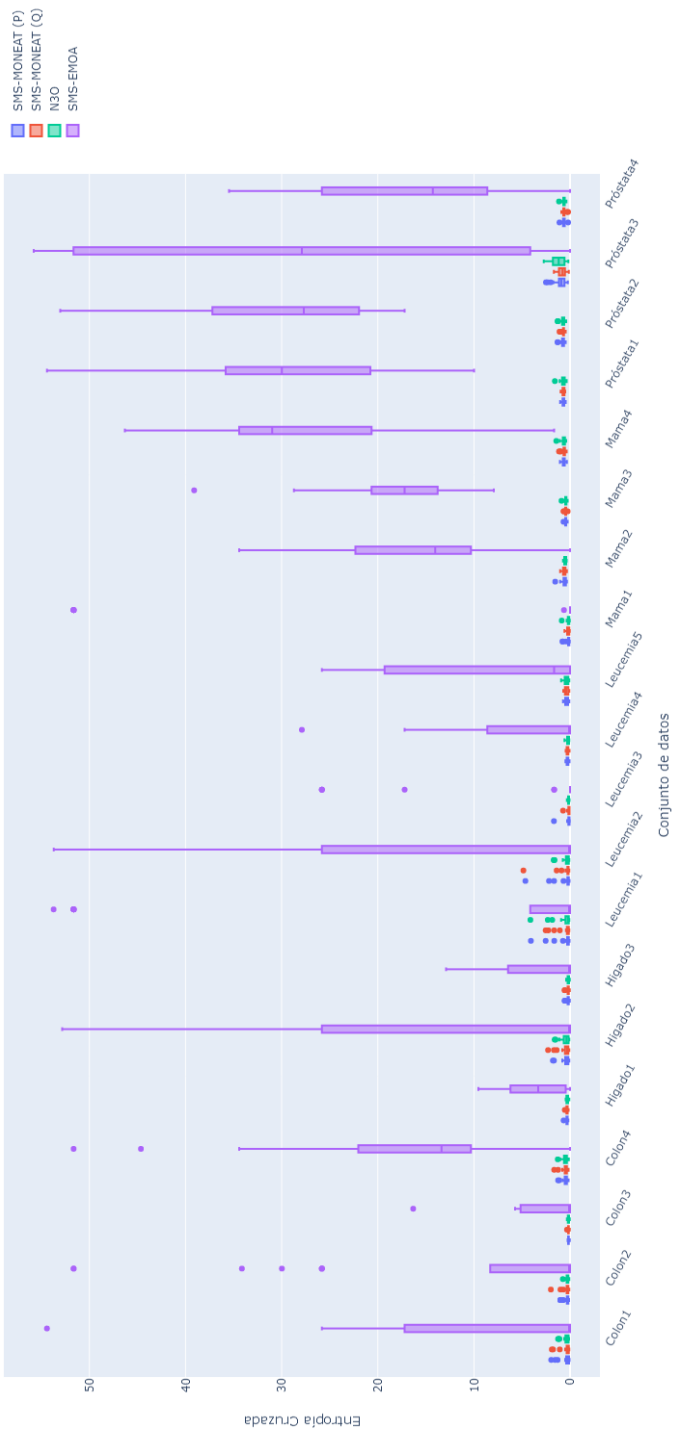


Figura 1.1: Resultados obtenidos de acuerdo con la entropía cruzada de las soluciones obtenidas mediante de N3O, SMS-EMOA y SMS-MONEAT.

Figura 1.2: Resultados obtenidos de acuerdo con el valor de entropía cruzada de las soluciones obtenidas mediante de N3O y SMS-MONEAT.

Figura 1.3: Resultados obtenidos de acuerdo con el número de características seleccionadas mediante de N3O, SMS-EMOA y SMS-MONEAT.

Figura 1.4: Resultados obtenidos de acuerdo con el hipervolumen de las poblaciones finales obtenidas mediante de N3O, SMS-EMOA y SMS-MONEAT.

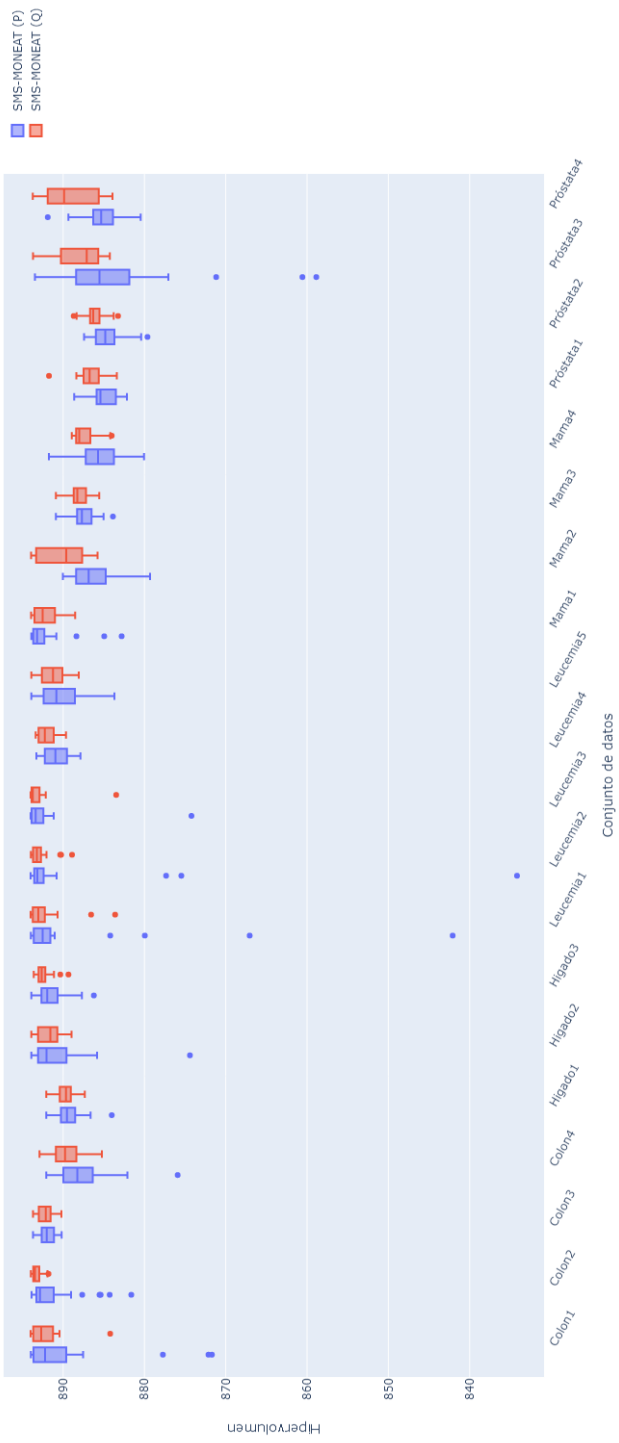


Figura 1.5: Resultados obtenidos de acuerdo con el hipervolumen de la población final y el archivo externo de SMS-MONEAT.

Figura 1.6: Resultados obtenidos de acuerdo con el promedio geométrico de las soluciones obtenidas mediante de N3O, SMS-EMOA y SMS-MONEAT.

Figura 1.7: Resultados obtenidos de acuerdo con el promedio geométrico de modelos SVM entrenados con las características seleccionadas mediante N3O, SMS-EMOA y SMS-MONEAT.

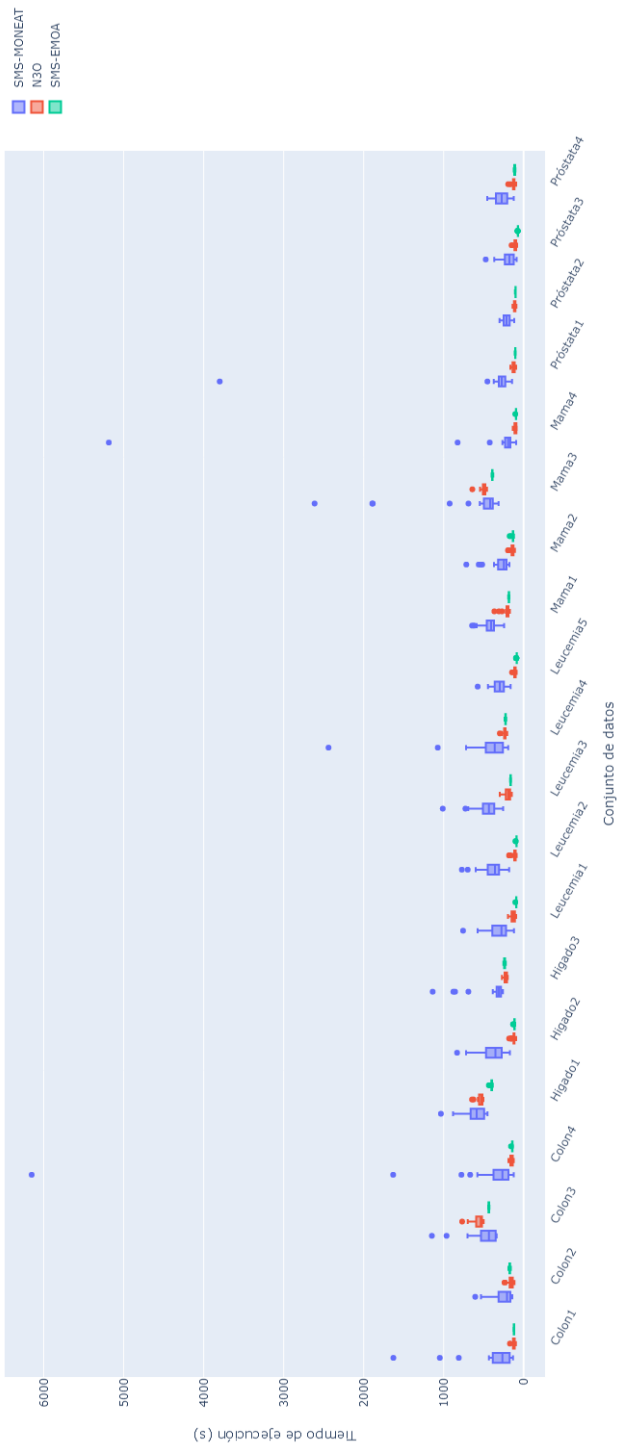


Figura 1.8: Tiempo de ejecución de los experimentos realizados para N3O, SMS-EMOA y SMS-MONEAT.

Figura 1.9: Tiempo de ejecución de los experimentos realizados para N3O y SMS-EMOA.

Bibliografía

[1] Leroy Hood y Lee Rowen. "The human genome project: big science transforms biology and medicine". En: *Genome medicine* 5.9 (2013), págs. 1-8.

[2] *The Human Genome Project*. URL: https://www.genome.gov/human-genome-project.

[3] Sergey Nurk y col. "The complete sequence of a human genome". En: *Science* 376.6588 (2022), págs. 44-53. DOI: 10.1126/science.abj6987. eprint: https://www.science.org/doi/pdf/10.1126/science.abj6987. URL: https://www.science.org/doi/abs/10.1126/science.abj6987.

[4] Manuel Everardo Reyna Murrieta y José Francisco Valenzuela Sánchez. "Microarreglos de ADN: aplicaciones en la microbiología". En: *Epistemus. Ciencia, tecnología y salud* 13.26 (2019), págs. 42-47.

[5] Beatriz A Garro, Katya Rodríguez y Roberto A Vázquez. "Classification of DNA microarrays using artificial neural networks and ABC algorithm". En: *Applied Soft Computing* 38 (2016), págs. 548-560.

[6] Verónica Bolón-Canedo y col. "A review of microarray datasets and applied feature selection methods". En: *Information sciences* 282 (2014), págs. 111-135.

[7] World Health Organization. *Cáncer*. URL: https://www.who.int/es/news-room/fact-sheets/detail/cancer.

[8] Manuel López-Ibáñez y col. "The irace package: Iterated racing for automatic algorithm configuration". En: *Operations Research Perspectives* 3 (2016), págs. 43-58.

[9] Warren S McCulloch y Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". En: *The bulletin of mathematical biophysics* 5.4 (1943), págs. 115-133.

[10] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." En: *Psychological review* 65.6 (1958), pág. 386.

[11] David E Rumelhart, Geoffrey E Hinton y Ronald J Williams. "Learning representations by back-propagating errors". En: *nature* 323.6088 (1986), págs. 533-536.

[12] Dario Floreano, Peter Dürr y Claudio Mattiussi. "Neuroevolution: from arquitectures to learning". En: *Evolutionary intelligence* 1.1 (2008), págs. 47-62.

[13] Edgar Galván y Peter Mooney. "Neuroevolution in deep neural networks: Current trends and future challenges". En: *IEEE Transactions on Artificial Intelligence* 2.6 (2021), págs. 476-493.

[14] Sarah DeWeerdt. "How to map the brain". En: *Nature* 571.7766 (2019), S6-S6.

[15] Kenneth O Stanley y col. "Designing neural networks through neuroevolution". En: *Nature Machine Intelligence* 1.1 (2019), págs. 24-35.

[16] Kenneth O Stanley y Risto Miikkulainen. "Evolving neural networks through augmenting topologies". En: *Evolutionary computation* 10.2 (2002), págs. 99-127.

[17] Shimon Whiteson y col. "Automatic feature selection in neuroevolution". En: *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. 2005, págs. 1225-1232.

[18] Bruno Iochins Grisci, Bruno César Feltes y Marcio Dorn. "Neuroevolution as a tool for microarray gene expression pattern identification in cancer research". En: *Journal of biomedical informatics* 89 (2019), págs. 122-133.

[19] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen y col. *Evolutionary algorithms for solving multi-objective problems*. Vol. 5. Springer, 2007.

[20] Nicola Beume, Boris Naujoks y Michael Emmerich. "SMS-EMOA: Multiobjective selection based on dominated hypervolume". En: *European Journal of Operational Research* 181.3 (2007), págs. 1653-1669.

[21] Kalyanmoy Deb y col. "A fast and elitist multiobjective genetic algorithm: NSGA-II". En: *IEEE transactions on evolutionary computation* 6.2 (2002), págs. 182-197.

[22] Oliver Schütze y Carlos Hernández. *Archiving Strategies for Evolutionary Multi-objective Optimization Algorithms*. Springer, 2021.

[23] Oliver Schuetze y col. "Archivers for the representation of the set of approximate solutions for MOPs". En: *Journal of Heuristics* 25 (2019), págs. 71-105.

[24] Eckart Zitzler y Lothar Thiele. "Multiobjective optimization using evolutionary algorithms—a comparative case study". En: *International conference on parallel problem solving from nature*. Springer. 1998, págs. 292-301.

[25] Holger H Hoos. "Automated algorithm configuration and parameter tuning". En: *Autonomous search* (2012), págs. 37-71.

[26] Mauro Birattari y col. "A Racing Algorithm for Configuring Metaheuristics." En: *Gecco*. Vol. 2. 2002. Citeseer. 2002.

[27] Mauro Birattari y col. "F-Race and iterated F-Race: An overview". En: *Experimental methods for the analysis of optimization algorithms* (2010), págs. 311-336.

[28] Widi Astuti y col. "Support vector machine and principal component analysis for microarray data classification". En: *Journal of Physics: Conference Series*. Vol. 971. 1. IOP Publishing. 2018, pág. 012003.

[29] Hema Shekar Basavegowda y Guesh Dagnew. "Deep learning approach for microarray cancer data classification". En: *CAAI Transactions on Intelligence Technology* 5.1 (2020), págs. 22-33.

[30] Rabia Aziz Musheer, CK Verma y Namita Srivastava. "Novel machine learning approach for classification of high-dimensional microarray data". En: *Soft Computing* 23.24 (2019), págs. 13409-13421.

[31] Md Maniruzzaman y col. "Statistical characterization and classification of colon microarray gene expression data using multiple machine learning paradigms". En: *Computer methods and programs in biomedicine* 176 (2019), págs. 173-193.

[32] QI Hou y col. "RankProd combined with genetic algorithm optimized artificial neural network establishes a diagnostic and prognostic prediction model that revealed C1QTNF3 as a biomarker for prostate cancer". En: *EBioMedicine* 32 (2018), págs. 234-244.

[33] Simon Anders y Wolfgang Huber. "Differential expression analysis for sequence count data". En: *Nature Precedings* (2010), págs. 1-1.

[34] Yawen Xiao y col. "A deep learning-based multi-model ensemble method for cancer prediction". En: *Computer methods and programs in biomedicine* 153 (2018), págs. 1-9.

[35] Shamveel Hussain Shah y col. "Optimized gene selection and classification of cancer from microarray gene expression data using deep learning". En: *Neural Computing and Applications* (2020), págs. 1-12.

[36] Xiaofei He, Deng Cai y Partha Niyogi. "Laplacian score for feature selection". En: *Advances in neural information processing systems* 18 (2005).

[37] Husna Aydadenta y Adiwijaya Adiwijaya. "A clustering approach for feature selection in microarray data classification using random forest". En: *Journal of Information Processing Systems* 14.5 (2018), págs. 1167-1175.

[38] Xianxue Yu, Guoxian Yu y Jun Wang. "Clustering cancer gene expression data by projective clustering ensemble". En: *PloS one* 12.2 (2017), e0171429.

[39] Siyabend Turgut, Mustafa Dağtekin y Tolga Ensari. "Microarray breast cancer data classification using machine learning methods". En: *2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT)*. IEEE. 2018, págs. 1-3.

[40] Isabelle Guyon y col. "Gene selection for cancer classification using support vector machines". En: *Machine learning* 46.1 (2002), págs. 389-422.

[41] Zifa Li, Weibo Xie y Tao Liu. "Efficient feature selection and classification for microarray data". En: *PloS one* 13.8 (2018), e0202167.

[42] Manosij Ghosh y col. "Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods". En: *Medical & biological engineering & computing* 57.1 (2019), págs. 159-176.

[43] Saeed Sarbazi-Azad, Mohammad Saniee Abadeh y Mohammad Erfan Mowlaei. "Using data complexity measures and an evolutionary cultural algorithm for gene selection in microarray data". En: *Soft Computing Letters* (2020), pág. 100007.

[44] C Pragadeesh y col. "Hybrid feature selection using micro genetic algorithm on microarray gene expression data". En: *Journal of Intelligent & Fuzzy Systems* 36.3 (2019), págs. 2241-2246.

[45] Alok Kumar Shukla, Pradeep Singh y Manu Vardhan. "A two-stage gene selection method for biomarker discovery from microarray data for cancer classification". En: *Chemometrics and Intelligent Laboratory Systems* 183 (2018), págs. 47-58.

[46] M Dashtban y Mohammadali Balafar. "Gene selection for microarray cancer classification using a new evolutionary method employing artificial intelligence concepts". En: *Genomics* 109.2 (2017), págs. 91-107.

[47] Alok Kumar Shukla. "Identification of cancerous gene groups from microarray data by employing adaptive genetic and support vector machine technique". En: *Computational Intelligence* 36.1 (2020), págs. 102-131.

[48] Ying Xiong y Fei Han. "A Hybrid Gene Selection Method for Microarray Data Based on Geodesic Distance and Binary Particle Swarm Optimization". En: *IOP Conference Series: Materials Science and Engineering*. Vol. 490. 4. IOP Publishing. 2019, pág. 042014.

[49] Ahmed Bir-Jmel, Sidi Mohamed Douiri y Souad Elbernoussi. "Gene selection via a new hybrid ant colony optimization algorithm for cancer classification in high-dimensional data". En: *Computational and mathematical methods in medicine* 2019 (2019).

[50] D Santhakumar y S Logeswari. "Hybrid ant lion mutated ant colony optimizer technique for Leukemia prediction using microarray gene data". En: *Journal of Ambient Intelligence and Humanized Computing* 12.2 (2021), págs. 2965-2973.

[51] Sedighe Abasabadi y col. "Hybrid feature selection based on SLI and genetic algorithm for microarray datasets". En: *The Journal of Supercomputing* (2022), págs. 1-29.

[52] Elham Pashaei y Elnaz Pashaei. "Gene selection using intelligent dynamic genetic algorithm and random forest". En: *2019 11th international conference on electrical and electronics engineering (ELECO)*. IEEE. 2019, págs. 470-474.

[53] Indu Jain, Vinod Kumar Jain y Renu Jain. "An improved binary particle swarm optimization (iBPSO) for gene selection and cancer classification using DNA microarrays". En: *2018 Conference on Information and Communication Technology (CICT)*. IEEE. 2018, págs. 1-6.

[54] Nashat Alrefai y Othman Ibrahim. "Optimized feature selection method using particle swarm intelligence with ensemble learning for cancer classification based on microarray datasets". En: *Neural Computing and Applications* (2022), págs. 1-16.

[55] Yamuna Prasad, KK Biswas y Madasu Hanmandlu. "A recursive PSO scheme for gene selection in microarray data". En: *Applied Soft Computing* 71 (2018), págs. 213-225.

[56] Subhajit Kar, Kaushik Das Sharma y Madhubanti Maitra. "Gene selection from microarray gene expression data for classification of cancer subgroups employing PSO and adaptive K-nearest neighborhood technique". En: *Expert Systems with Applications* 42.1 (2015), págs. 612-627.

[57] Amalya Citra Pradana, A Aditsania y col. "Implementing binary particle swarm optimization and C4. 5 decision tree for cancer detection based on microarray data classification". En: *Journal of Physics: Conference Series*. Vol. 1192. 1. IOP Publishing. 2019, pág. 012014.

[58] Diana Nurlaily y col. "Support vector machine for imbalanced microarray dataset classification using ant colony optimization and genetic algorithm". En: *AIP Conference Proceedings*. Vol. 2194. 1. AIP Publishing LLC. 2019, pág. 020076.

[59] Beatriz A Garro, Katya Rodríguez y Roberto A Vazquez. "Designing artificial neural networks using differential evolution for classifying DNA microarrays". En: *2017 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. 2017, págs. 2767-2774.

[60] Hong Wang, Xingjian Jing y Ben Niu. "A discrete bacterial algorithm for feature selection in classification of microarray gene expression cancer data". En: *Knowledge-Based Systems* 126 (2017), págs. 8-19.

[61] V Kalaimani y R Umagandhi. "A novel wrapper FS based on binary swallow swarm optimization with score-based criteria fusion for gene expression microarray data". En: *Materials Today: Proceedings* (2020).

[62] Milad Mostavi y col. "Convolutional neural network models for cancer type prediction based on gene expression". En: *BMC medical genomics* 13.5 (2020), págs. 1-13.

[63] Rajendra Rana Bhat, Vivek Viswanath y Xiaolin Li. "DeepCancer: detecting cancer through gene expressions via deep generative learning". En: *arXiv preprint arXiv:1612.03211* (2016).

[64] Yuchen Yuan y col. "DeepGene: an advanced cancer type classifier based on deep learning and somatic point mutations". En: *BMC bioinformatics* 17.17 (2016), págs. 243-256.

[65] Diyar Qader Zeebaree, Habibollah Haron y Adnan Mohsin Abdulazeez. "Gene selection and classification of microarray data using convolutional neural network". En: *2018 International Conference on Advanced Science and Engineering (ICOASE)*. IEEE. 2018, págs. 145-150.

[66] Dor Bank, Noam Koenigstein y Raja Giryes. "Autoencoders". En: *arXiv preprint arXiv:2003.05991* (2020).

[67] Alexander J Titus, Carly A Bobak y Brock C Christensen. "A new dimension of breast cancer epigenetics". En: *9th International Conference on Bioinformatics Models, Methods and Algorithms*. 2018.

[68] Gregory P Way y Casey S Greene. "Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders". En: *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2018: Proceedings of the Pacific Symposium*. World Scientific. 2018, págs. 80-91.

[69] Kumardeep Chaudhary y col. "Deep Learning–Based Multi-Omics Integration Robustly Predicts Survival in Liver CancerUsing Deep Learning to Predict Liver Cancer Prognosis". En: *Clinical Cancer Research* 24.6 (2018), págs. 1248-1259.

[70] Dejun Zhang y col. "Integrating feature selection and feature extraction methods with deep learning to predict clinical outcome of breast cancer". En: *IEEE Access* 6 (2018), págs. 28936-28944.

[71] J Liu y col. *Tumor gene expression data classification via sample expansionbased deep learning. Oncotarget. 2017; 8 (65): 109646.*

[72] Padideh Danaee, Reza Ghaeini y David A Hendrix. "A deep learning approach for cancer detection and relevant gene identification". En: *Pacific symposium on biocomputing 2017*. World Scientific. 2017, págs. 219-229.

[73] Maisa Daoud y Michael Mayo. "A survey of neural network-based cancer prediction models from microarray data". En: *Artificial intelligence in medicine* 97 (2019), págs. 204-214.

[74] Bruno Iochins Grisci, Bruno César Feltes y Márcio Dorn. "Microarray classification and gene selection with FS-NEAT". En: *2018 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. 2018, págs. 1-8.

[75] DeviPriya Rangasamy, Sivaraj Rajappan y Mohan Natesan. "A Multi-Objective Evolutionary Approach for Preprocessing Imbalanced Microarray Datasets". En: *Computing in Science & Engineering* 22.1 (2018), págs. 88-100.

[76] Julieta Sol Dussaut y col. "Comparing multiobjective evolutionary algorithms for cancer data microarray feature selection". En: *2018 IEEE congress on evolutionary computation (CEC)*. IEEE. 2018, págs. 1-8.

[77] Aman Sharma y Rinkle Rani. "C-HMOSHSSA: Gene selection for cancer classification using multi-objective meta-heuristic and machine learning methods". En: *Computer methods and programs in biomedicine* 178 (2019), págs. 219-235.

[78] Mohd Shahizan Othman, Shamini Raja Kumaran y Lizawati Mi Yusuf. "Gene selection using hybrid multi-objective cuckoo search algorithm with evolutionary operators for cancer microarray data". En: *IEEE Access* 8 (2020), págs. 186348-186361.

[79] Ali Dabba, Abdelkamel Tari y Samy Meftali. "A new multi-objective binary Harris Hawks optimization for gene selection in microarray data". En: *Journal of Ambient Intelligence and Humanized Computing* (2021), págs. 1-20.

[80] Abhilasha Chaudhuri y Tirath Prasad Sahu. "Multi-objective feature selection based on quasi-oppositional based Jaya algorithm for microarray data". En: *Knowledge-Based Systems* 236 (2022), pág. 107804.

[81] Yang Qing y col. "Cooperative coevolutionary multiobjective genetic programming for microarray data classification". En: *Proceedings of the Genetic and Evolutionary Computation Conference*. 2021, págs. 804-811.

[82] Willem van Willigen, Evert Haasdijk y Leon Kester. "Fast, comfortable or economical: evolving platooning strategies with many objectives". En: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE. 2013, págs. 1448-1455.

[83] Eckart Zitzler, Marco Laumanns y Lothar Thiele. "SPEA2: Improving the strength Pareto evolutionary algorithm". En: *TIK-report* 103 (2001).

[84] Omer Abramovich y Amiram Moshaiov. "Multi-objective topology and weight evolution of neuro-controllers". En: *2016 IEEE congress on evolutionary computation (CEC)*. IEEE. 2016, págs. 670-677.

[85] Jacob Schrum y Risto Miikkulainen. "Discovering multimodal behavior in Ms. Pac-Man through evolution of modular neural networks". En: *IEEE transactions on computational intelligence and AI in games* 8.1 (2015), págs. 67-81.

[86] Steven Künzel y Silja Meyer-Nieberg. "Coping with opponents: multi-objective evolutionary neural networks for fighting games". En: *Neural Computing and Applications* 32.17 (2020), págs. 13885-13916.

[87] Steven Künzel y Silja Meyer-Nieberg. "Evolving artificial neural networks for multi-objective tasks". En: *International Conference on the Applications of Evolutionary Computation*. Springer. 2018, págs. 671-686.

[88] Daniel García-Núñez, Katya Rodriguez-Vázquez y Carlos Hernández. "Neuroevolution Based Multi-Objective Algorithm for Gene Selection and Microarray Classification". En: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. GECCO '22. Boston, Massachusetts: Association for Computing Machinery, 2022, 647–650. ISBN: 9781450392686. DOI: 10.1145/3520304.3529058. URL: https://doi.org/10.1145/3520304.3529058.

[89] Fabian Pedregosa y col. "Scikit-learn: Machine learning in Python". En: *Journal of machine learning research* 12.Oct (2011), págs. 2825-2830.

[90] Hernán Sosa, Silvia Myriam Villagra y Norma Andrea Villagra. "Operadores de mutación en algoritmos genéticos celulares aplicados a problemas continuos". En: *Informes Científicos Técnicos-UNPA* 6.2 (2014), págs. 141-157.

[91] Francesco Biscani y Dario Izzo. "A parallel global multiobjective framework for optimization: pagmo". En: *Journal of Open Source Software* 5.53 (2020), pág. 2338. DOI: 10.21105/joss.02338. URL: https://doi.org/10.21105/joss.02338.

[92] Yanmin Sun y col. "Cost-sensitive boosting for classification of imbalanced data". En: *Pattern recognition* 40.12 (2007), págs. 3358-3378.

[93] Bruno César Feltes y col. "Cumida: an extensively curated microarray database for benchmarking and testing of machine learning approaches in cancer research". En: *Journal of Computational Biology* 26.4 (2019), págs. 376-386.

[94] Todd R Golub y col. "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring". En: *science* 286.5439 (1999), págs. 531-537.

[95] Marc J Van De Vijver y col. "A gene-expression signature as a predictor of survival in breast cancer". En: *New England Journal of Medicine* 347.25 (2002), págs. 1999-2009.

[96] Dinesh Singh y col. "Gene expression correlates of clinical prostate cancer behavior". En: *Cancer cell* 1.2 (2002), págs. 203-209.

[97] Danilo Vasconcellos Vargas y Junichi Murata. "Spectrum-diverse neuroevolution with unified neural models". En: *IEEE transactions on neural networks and learning systems* 28.8 (2016), págs. 1759-1773.