A Memetic Cellular Genetic Algorithm for Cancer Data Microarray Feature Selection

Matías Gabriel Rojas, Ana Carolina Olivera, Jessica Andrea Carballido, Pablo Javier Vidal

***Abstract*—Gene selection aims at identifying a -small- subset of informative genes from the initial data to obtain high predictive accuracy for classification in human cancers. Gene selection can be considered as a combinatorial search problem and thus can be conveniently handled with optimization methods. This paper proposes a Memetic Cellular Genetic Algorithm (MCGA) to solve the Feature Selection problem of cancer microarray datasets. Benchmark gene expression datasets, i.e., colon, lymphoma, and leukaemia available in the literature were used for experimentation. MCGA is compared with other well-known metaheuristic' strategies. The results demonstrate that our proposal can provide efficient solutions to find a minimal subset of the genes.**

***Index Terms*—Feature Selection, Microarray Classification, Cellular Genetic Algorithm, Memetic Algorithms**

I. Introducción

El uso de sistemas de clasificación en el diagnóstico médico está aumentando gradualmente. No hay duda de que la evaluación de los datos tomados de los pacientes y las decisiones de los expertos son los factores más críticos en el diagnóstico. Las diferentes técnicas de inteligencia artificial para la clasificación facilitan y contribuyen a la decisión de los expertos. Los sistemas de clasificación pueden ayudar a minimizar los errores que se pueden cometer a la vez que proporcionan datos médicos para examinar en un tiempo más corto y de manera más detallada.

Los sistemas de clasificación se utilizan para el diagnóstico de cáncer, una enfermedad que comienza con la división incontrolada de una célula y da como resultado una masa visible llamada tumor (puede ser benigno o maligno). El tumor maligno crece rápidamente e invade los tejidos circundantes causando algún daño. El diagnóstico de cáncer se realiza en función de criterios no moleculares como el tipo de tejido, las propiedades patológicas, la ubicación clínica, entre otros [1].

Es difícil determinar qué genes o características son útiles para identificar y diagnosticar sin conocimiento previo. Los genes irrelevantes y redundantes no son valiosos para la clasificación. Como tal, los métodos efectivos de selección de genes para el cáncer son críticamente necesarios. El proceso de selección tiene como objetivo elegir el número mínimo de genes relativos e informativos que son más predictivos en el proceso de clasificación.

Existen diferentes métodos para la selección de genes o características: filtrado (*filter*) [2], envoltura (*wrapper*) [3], híbrido [4] y embedido (*embebbed*) [5]. Mediante el filtrado, la búsqueda no depende del algoritmo de aprendizaje, sino que se clasifica según los valores aportados por los test estadísticos -t-test, ANOVA- que tienden a determinar la relación individual de las características con la variable de resultado. Por su parte, los métodos de envoltura necesitan un algoritmo de aprendizaje automático (machine learning) que provea información sobre su rendimiento (precisión de clasificación) para utilizarla como criterio de evaluación. Estas cualidades hacen que los métodos de filtro sean menos precisos pero más rápidos en comparación con la envoltura. Los métodos híbridos crean una cooperación entre los filtros y las envolturas para hacer una compensación entre el rendimiento y la complejidad del tiempo. En los métodos embebidos, la selección de características se realiza en sincronización con el proceso de aprendizaje.

La selección de características pertenece a la clase de problemas de optimización combinatoria NP-difícil [6]. Para un conjunto de n genes hay 2^n subconjuntos. Los métodos tradicionales de búsqueda determinista pueden ser computacionalmente costosos a la hora de buscar la solución óptima. Para reducir la complejidad del tiempo, se han propuesto métodos estocásticos alternativos en la literatura: probabilístico [7], heurístico [8], metaheurístico [9] e hibridación de metaheurísticas [10].

Las metaheurísticas se basan en dos conceptos: exploración y explotación del espacio de búsqueda. En particular, las metaheurísticas con población se centran en la exploración del espacio de búsqueda. Un ejemplo de metaheurísticas poblacionales son los algoritmos genéticos (AG) [11]. Un AG tiene una velocidad de convergencia rápida, pero puede perder la diversidad de la población y quedar atrapado en un óptimo local. Para superar este inconveniente, se estudian los AG descentralizados para mantener la diversidad de soluciones. Uno de ellos es el algoritmo genético celular (AGC) [12].

Sin embargo, la exploración puede retrasar la convergencia en el caso de conjuntos de datos de gran tamaño [13]. En un AGC, la población se establece conceptualmente en una estructura topológica (también llamada cuadrícula o malla de población). Según esta estructura, para cada solución (individuo central), varios individuos en sus celdas cercanas de la cuadrícula se asignan como vecinos (vecindario). El individuo central solo interactúa con sus vecinos. De esta manera, los pequeños vecindarios superpuestos del AGC ayudan a

Matías Gabriel Rojas, Ana Carolina Olivera, Pablo Javier Vidal. Instituto Universitario para las Tecnologías de la Información y las Comunicaciones, Universidad Nacional de Cuyo, Facultad de Ingeniería (UNCuyo). CONICET, Mendoza, Argentina.

Jessica Andrea Carballido, Instituto de Ciencias e Ingeniería de la Computación, CONICET, Universidad Nacional del Sur, Bahía Blanca, Argentina

e-mails: rojasmatias994@gmail.com, acolivera@conicet.gov.ar, jac@cs.uns.edu.ar,pjvidal@conicet.gov.ar

Corresponding authors: P.J.Vidal

Manuscript received February, 2020; revised XX XX, 20XX.

explorar el espacio de búsqueda, ya que la difusión lenta inducida de soluciones a través de la población proporciona un tipo de exploración (diversificación). La explotación (intensificación) ocurre dentro de cada vecindario mediante operaciones genéticas, lo que permite que el AGC mantenga un equilibrio adecuado entre diversidad y convergencia [12].

Sin embargo, para algunos problemas, la explotación puede causar una convergencia prematura. Por lo tanto, una metaheurística efectiva tiene que ser capaz de encontrar un equilibrio entre los dos conceptos anteriores. Para ello, se proponen los métodos meméticos [14]. Una estrategia memética consta de al menos dos métodos distintos. En tales estrategias, la parte crucial radica en la elección de sus componentes de acuerdo con sus características.

En este trabajo, se presenta un algoritmo genético celular memético (AGCM), buscando utilizar la capacidad de exploración y explotación del AGC y además, mejorar la capacidad de búsqueda en zonas cercanas mediante la utilización de una hibridación con una búsqueda local diseñada especialmente para el problema de selección de características. Para evaluar el comportamiento del AGCM, se compara con *algoritmos genéticos* (AG) [11], *algoritmos genéticos celulares* (AGC) versión canónica [12], *recocido simulado* (RS) [15] y el algoritmo de *optimización por cúmulo de partículas binario* (ACPB) [16] versión 2011.

El manuscrito se organiza de la siguiente manera: La Sección II introduce el problema y los trabajos relacionados. En la Sección III se presenta el algoritmo genético celular memético propuesto en este trabajo. La Sección IV detalla la configuración de los experimentos y los resultados obtenidos. Finalmente la Sección V, muestra las conclusiones y el trabajo futuro.

II. Selección de Características

Uno de los principales desafíos en bioinformática es seleccionar grupos de genes informativos con una alta potencia predictiva a partir de las muestras que se tienen. El mayor problema en los datos de expresión génica es su alta dimensionalidad. Usualmente estos datos se encuentran en forma de matriz conteniendo una gran cantidad de genes (filas) y una pequeña cantidad de muestras (columnas).

La selección de características es un problema de optimización combinatorial (NP-difícil) [17]. Su objetivo es eliminar características que no aporten al problema de clasificación o bien que sean redundantes puesto que tienen la misma información que otras. La selección de características para datos de cáncer en *microarrays* es el proceso de clasificar cualquier muestra de ADN (Ácido DesoxirriboNucleico) en función de los genes identificados. La muestra puede clasificarse como "Sin cáncer" o "Cáncer" en el caso de un conjunto de datos de clase binaria.

A. Definición del Problema

Sea G un conjunto de datos con M muestras, cada una de ellas con una dimensión de tamaño N (N características), teniendo una matriz de $M \times N$. El objetivo es encontrar un subconjunto de n características ($n \leq N$), de manera tal que mientras más relevantes sean esas n características, la precisión de clasificación mejore.

Sea x un vector binario de tamaño N, $x_i = 1$ denota que la característica i ha sido seleccionada mientras que $x_i = 0$ expresa que la característica i no ha sido seleccionada. La función objetivo a maximizar se puede apreciar en (1) y muestra la función de aptitud que considera maximizar la precisión de clasificación y reducir el número de características seleccionadas al evaluar una solución x,

$$aptitud(x) = \beta * precisión(x) + \alpha * \left(1 - \frac{\sum_{i=1}^{N} x_i}{N}\right) \quad (1)$$

donde β y α son pesos y se establecen en 0.9 y 0.1, respectivamente, con el fin de controlar que el valor de precisión tenga prioridad sobre el tamaño del subconjunto. La *precisión* se calcula como la suma de las clasificaciones de cáncer correctas divididas por el número total de clasificaciones considerando las características seleccionadas en x.

B. Trabajos Relacionados

En esta ultima década las técnicas metaheurísticas, y en particular aquellas híbridas han sido aplicadas con éxito para este tipo de problemas [18]–[21]. Al-Thanoon *et al.* [18] utilizan diferentes versiones del algoritmo de luciérnagas en conjunto con el algoritmo de cúmulo de partículas. Los resultados experimentales muestran la superioridad del enfoque presentado en términos de precisión de clasificación y el número de descriptores seleccionados.

Recientemente, Xu y Yan [19] han hibridado el algoritmo Nelder-Mead con el algoritmo de libélula para mejorar la capacidad de exploración del mismo. Mafarja y Mirjalili [20] propusieron un modelo de hibridación que utiliza el Recocido Simulado (RS) y el Algoritmo de Optimización de Ballenas (AOB). Baliarsingh *et al.* [21] proponen una nueva versión utilizando el algoritmo de optimización de ingeniería social con el algoritmo del pingüino emperador.

Los estudios anteriores demuestran que los métodos metaheurísticos híbridos exhiben un mejor rendimiento en comparación con otras estrategias de búsqueda locales o globales.

Para nuestro conocimiento, en la literatura no se ha encontrado ningún enfoque memético que utilice los componentes y su variación para el problema de selección de características propuestos en el presente trabajo.

III. Algoritmo Genético Celular Memético

En esta sección presentamos el algoritmo genético celular memético (AGCM) propuesto especialmente para el problema de selección de características. Se introducen cada uno de los componentes para luego presentar el esquema general del AGCM.

Los algoritmos meméticos (AM) representan un paradigma de optimización basado en el trabajo cooperativo. Los AM buscan obtener una sinergia entre la combinación de diferentes técnicas de optimización con el objetivo de mejorar el proceso de búsqueda. Cada técnica utilizada puede estar basada en metaheurísticas existentes o componentes de las

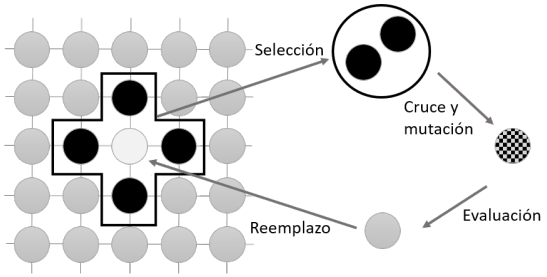

Fig. 1. Selección de un vecindario y aplicación de operadores genéticos en un AGC

mismas tratando de realizar una explotación sistemática del conocimiento acerca del problema que se desea resolver.

Un AM emplea al menos dos componentes base. El primero, es el componente global, dado por un algoritmo de exploración general, que posibilita buscar dentro del espacio de soluciones de manera efectiva. El segundo componente, es una búsqueda local que sirve para ayudar a la búsqueda del algoritmo global. De estas forma, se evita caer en óptimos locales e intensificar la explotación de espacios cercanos a las soluciones que sean prometedoras. De esta manera la búsqueda local incrementa la calidad de los resultados.

El enfoque propuesto en esta sección tiene como componente global de exploración un algoritmo genético celular (AGC). El segundo componente, es una búsqueda local (BL) diseñada específicamente para la selección de características. La BL examina pequeñas variaciones en las soluciones que posibiliten lograr un salto de calidad que ayude en la búsqueda del valor óptimo.

En el resto de esta sección se describen en profundidad los componentes principales del AGCM, el algoritmo genético celular (Sección III-A) y la búsqueda local (Sección III-B). Posteriormente, se explica el procedimiento que permite calcular la precisión para cada solución (Sección III-C). Finalmente en la Sección III-D, se detalla el funcionamiento del AGCM con sus componentes.

A. Algoritmo Genético Celular

Un algoritmo genético celular (AGC) [12] es una variante del algoritmo genético (AG) [11], [12]. Difiere sustancialmente en el manejo de la población, el AGC utiliza una distribución descentralizada en la que las soluciones (individuos) tentativas evolucionan en vecindarios superpuestos. En un AGC, los individuos son situados en una malla toroidal bidimensional y se aplican operadores de perturbación (usualmente operadores genéticos) teniendo en cuenta a los individuos cercanos (vecindario). Este vecindario ayuda en la exploración del espacio de búsqueda debido a que, por medio de una lenta difusión de la calidad de las soluciones a través de la población, se esta proporcionando exploración, mientras que, la explotación tiene lugar dentro de cada vecindario.

El AGC comienza creando la población inicial y evaluándola. Una vez terminado este primer paso, se verifica si la condición de parada se ha cumplido, en caso negativo comienza el ciclo evolutivo. Primeramente, se selecciona un individuo y su vecindario. En este vecindario, los individuos sólo pueden interactuar con sus vecinos en cada paso del algoritmo. Se selecciona un determinado número de individuos del vecindario como padres de acuerdo a cierto criterio, se aplican los operadores de perturbación (recombinación y mutación), y se reemplaza el individuo actual por el descendiente recientemente creado siguiendo un criterio de reemplazo. Este paso se repite y continua hasta satisfacer la condición de finalización del algoritmo. La aplicación de los diferentes operadores genéticos sobre un vecindario se puede apreciar en la Fig. 1. El tipo de vecindario en este caso se denomina L5 (linear de 5 individuos) integrado por los individuos ubicados al Norte, Este, Oeste y Sur de la solución actual. El AGC es el componente global del algoritmo genético celular memético propuesto.

B. Búsqueda Local

Una búsqueda local (BL) permite intensificar la explotación en un espacio cercano a la solución analizada, para tratar de encontrar espacios prometedores que aún no han sido explorados. Para ayudar al AGC a mejorar la capacidad de explotación se utiliza el algoritmo de búsqueda de vecindario variable (BVV) [22]. Esta técnica comienza con una solución inicial e intenta mejorarla visitando distintos vecindarios a través de perturbaciones que dependen del problema a optimizar. Para el problema de selección de características la perturbación propuesta toma una solución y genera una copia de la misma. Luego, se seleccionan y desactivan aleatoriamente hasta tres características en forma consecutiva, repitiendo este proceso una cantidad de veces determinada. Si la solución obtenida es mejor a la solución actual se reemplaza.

C. Cálculo de la Precisión

Un clasificador permite evaluar la precisión que tienen las características seleccionadas en una solución a partir de la correcta o incorrecta clasificación de las muestras en un grupo (clase) determinado. Un clasificador utiliza una parte de las muestras como entrenamiento y otra para evaluar que tan buena es la capacidad de clasificación. Para este trabajo se utiliza el método de los k Vecinos más Cercanos (k-VC). Una muestra es asignada a la clase más común entre sus k vecinos más cercanos medidos por una función de proximidad [23]. En el AGCM propuesto, k-VC se aplica para evaluar la precisión de la clasificación de las características seleccionadas en una solución dentro del cálculo de la aptitud (1).

D. Esquema General del AGCM

A continuación, la Fig. 2 describe la interacción de los diferentes componentes del AGCM. Observando la Fig. 2 podemos considerar tres etapas:

- Etapa de inicialización: Se ingresa el conjunto de datos, se inicializa la población de forma aleatoria. Para cada solución generada, se evalúa su aptitud considerando la precisión obtenida por el algoritmo k-VC.
- Etapa Evolutiva: Se evalúa la condición de parada. En caso de no cumplirla, a partir de la población inicial evaluada, se aplican los operadores genéticos del AGC

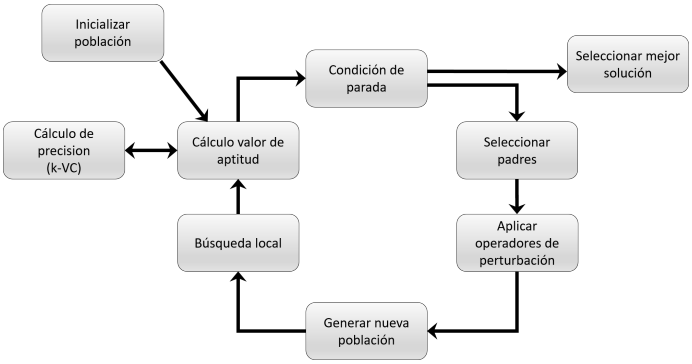

Fig. 2. Esquema del AGCM

para poder generar una nueva población. En el AGC, cada solución selecciona un vecindario y sobre este se aplican los operadores genéticos (Fig. 1). Una vez obtenida la nueva población, se procede a aplicarle BVV. Este ciclo se repite hasta alcanzar la condición de parada.

- Etapa de finalización: Una vez que la etapa evolutiva ha alcanzado la condición de parada, se procede a seleccionar la mejor solución de la población final e informarla.

IV. Experimentos y Resultados

En esta sección, se introducen las instancias utilizadas y los algoritmos junto con sus configuraciones. También, se describen los experimentos y el análisis estadístico realizado sobre los resultados obtenidos.

A. Configuración de Rxperimentos

Las instancias utilizadas en este trabajo provienen de tres tipos diferentes de cáncer y se usan ampliamente en la literatura:

- Colon [24]: tiene 2000 genes, 40 de 62 muestras de tejido han sido tomadas de biopsias tumorales y etiquetadas como negativas y 22 son de biopsias normales y etiquetadas como positivas.
- Linfoma [25]: tiene 4026 genes, 24 de 42 muestras de tejido han sido etiquetadas como centro B germinal y 22 como centro B activado.
- Leucemia [26]: tiene 7129 genes, 47 de 72 muestras de tejido han sido tomadas de pacientes con leucemia linfoblástica aguda (LLA) y 25 de pacientes con leucemia mieloide aguda (LMA).

B. Preprocesamiento de Características y División de Muestras

Los conjuntos de datos fueron preprocesados utilizando la herramienta estadística R y sus paquetes ClusterSim (*Searching for Optimal Clustering Procedure for a Data Set*) [27] y CARET (*Classification And REgression Training*) [28] para normalizar y reducir la dimensionalidad en columnas de la matriz. El preprocesamiento de los datos consta de dos fases. Primero, se realiza una normalización de cada característica (c_j) para cada muestra (m_i) como se muestra en (2).

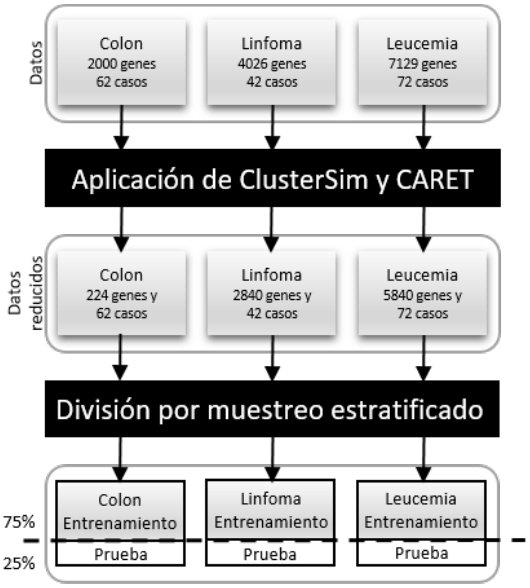

Fig. 3. Preprocesamiento de las muestras con ClusterSim y CARET. Posterior división por muestreo estratificado.

$$c'_j(m_i) = \frac{(c_j(m_i) - \min_j)}{(\max_j - \min_j)}, \forall i \tag{2}$$

donde max_j y min_j son el valor máximo y mínimo de la expresión génica respectivamente para la característica c_j sobre el conjunto de datos [27]. Se utiliza ClusterSim para normalizar cada columna mediante el comando *data.Normalization* con el parámetro de normalización *n4*.

Posteriormente, CARET permite encontrar aquellas características que presentan una alta correlación y suprimirlas de los datos. El proceso de eliminación de las correlaciones existentes es realizado para mejorar la precisión y realizar un entrenamiento más rápido. El coeficiente de correlación mínimo utilizado para medir la relación entre los datos es de 0.75. Este valor es recomendado por los desarrolladores del paquete, además, según [29], los coeficientes de correlación cuya magnitud está entre 0.7 y 0.9 indican variables que pueden considerarse altamente correlacionadas. Esto implica que un valor de 0.75 es adecuado para todos los conjuntos de datos. Como resultado se obtuvo una reducción del conjunto de datos de cáncer de colon, de 2000 a 224 genes; del conjunto de datos de linfoma, de 4026 a 2840 genes y del conjunto de datos de leucemia de 7129 a 5816 genes.

Finalmente, los conjuntos de datos reducidos se dividen en dos partes, un conjunto de entrenamiento (75%) y uno de prueba (25%) para ejecutar el método de clasificación k-VC. La Fig. 3 muestra el proceso general realizado para el preprocesamiento de conjunto de datos.

C. Algoritmos Evaluados

Diferentes autores en la literatura han trabajado y verificado la capacidad de los algoritmos evolutivos sobre problemas de decisión binaria. En el presente trabajo, se utilizaron cinco algoritmos de optimización en su versión canónica: *algoritmos genéticos* (AG) [11], *algoritmos genéticos celulares*

TABLA I
CONFIGURACIÓN DE CADA ALGORITMO

Algoritmo	Parámetro	Valor
AGC	Tamaño población	10×10
	Operador Cruce	HUX - prob.:1.0
	Operador Mutación	Bit-Flip - prob.:(1/*n*)
	Operador Selección	Torneo Binario
	Vecindario	C9
AGCM	Tamaño población	10×10
	Operador Cruce	HUX - prob.:1.0
	Operador Mutación	Bit-Flip - prob.:(1/*n*)
	Operador Selección	Torneo Binario
	Vecindario	C9
	Búsqueda local (BL)	Búsqueda de Vecindarios Variable
	Evaluaciones (BL)	50
RS	Tamaño población	1
	Operador Mutación	Bit-Flip - prob.:(1/*n*)
AG	Tamaño población	100
	Operador Cruce	HUX - prob.:1.0
	Operador Mutación	Bit-Flip - prob.:(1/*n*)
	Operador Selección	Torneo Binario
ACPB	Tamaño población	30
	Actualización de Posición	Función Sigmoidal

(AGC) [12], *recocido simulado* (RS) [15] y el algoritmo de *optimización por cúmulo de partículas binario* (ACPB) [16] versión 2011. La Tabla I informa la configuración utilizada por cada uno de ellos. Los parámetros incluidos en esta tabla se han alcanzado a través de pruebas preliminares. Se utilizó un diseño simplificado con tres valores discretos para cada parámetro (pequeño, mediano, alto) de acuerdo a su función teniendo en cuenta la literatura existente [12], [22], [30]. Por otra parte, el valor de vecinos utilizados para clasificar con k-VC es de 5.

Todos los experimentos se han realizado con el procesador AMD FX(tm)-8320 con ocho núcleos, una memoria física total de 16 GB. El sistema operativo es Ubuntu 18.04 LTS. Los algoritmos se han desarrollado en jmetalpy implementado con python 3.7 [31]. Para todos ellos, la condición de parada se establece en 10000 evaluaciones del valor de aptitud. Dada la naturaleza no determinista de nuestros enfoques, se realizaron 20 ejecuciones independientes para cada algoritmo en cada conjunto de datos a fin de evaluar el rendimiento de los algoritmos y la significancia estadística de los resultados.

Para comparar el rendimiento de los algoritmos se utilizó la prueba de Wilcoxon (con un nivel de significancia α = 0.01) con el fin de identificar diferencias entre algoritmos [32]. La prueba de Wilcoxon realiza comparaciones individuales entre dos algoritmos (comparaciones por pares) que tienen como objetivo detectar diferencias significativas entre ellos. Los resultados se consideran significativos cuando el valor p es menor a 0.01.

D. Resultados Experimentales

El resultado de los experimentos pretende evaluar el desempeño y la robustez del enfoque propuesto (AGCM) en comparación con otras técnicas metaheurísticas sobre tres conjunto de datos. Se busca conocer el efecto de la selección de características en la mejora de la precisión de la clasificación. A su vez se pretende tener en cuenta también el tiempo computacional de ejecución. Las tablas tienen resaltado en color gris oscuro el mejor valor obtenido para la estadística analizada y en gris claro el segundo mejor valor.

TABLA II
VALOR PROMEDIO Y DESVIACIÓN ESTÁNDAR DEL VALOR DE APTITUD PARA COLON, LINFOMA Y LEUCEMIA

Algoritmo	Colon		Linfoma		Leucemia	
	Aptitud	Desv. St.	Aptitud	Desv. St.	Aptitud	Desv. St.
AGC	0.995	±0.0031	0.892	±0.0201	0.828	±0.0177
AGCM	0.994	±0.0011	0.998	±0.0006	0.962	±0.0283
AG	0.998	±0.0005	0.896	±0.0199	0.833	±0.0201
RS	0.983	±0.0327	0.914	±0.0526	0.832	±0.0113
ACPB	0.976	±0.0033	0.871	±0.0011	0.817	±0.0179

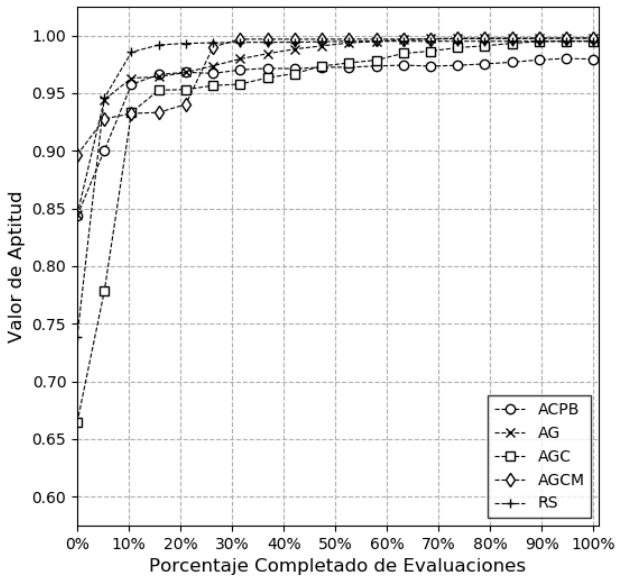

Fig. 4. Evolución del valor de aptitud para la instancia de colon

La Tabla II muestra el mejor valor de aptitud promedio con su desviación estándar obtenido de las 20 ejecuciones. Se observa que los mejores resultados (valor más alto) se obtienen con el algoritmo AGCM para dos de los tres conjuntos de datos (salvo el de colon, donde aparece como tercero), lo que sugiere que la selección de características mediante un método que equilibre la exploración con la explotación y la introducción de una búsqueda local considerando las particularidades del problema, puede mejorar significativamente la precisión de la clasificación.

Siguiendo con el análisis de la Tabla II, AG consigue los mejores resultados para la instancia colon. En el caso de ACPB, sus resultados son los peores con respecto a los otros algoritmos. En el caso de RS consigue para una de las tres instancias los mejores segundos valores por detrás de AGCM. Estos resultados indican que el enfoque AGCM puede realizar una exploración más inteligente del espacio de búsqueda que el resto de técnicas.

En el contexto de este trabajo, AGCM ha podido obtener soluciones de mayor calidad combinando ambas técnicas (AGC y la búsqueda local BVV), en lugar de utilizar las técnicas canónicas.

La evolución del valor de aptitud de los algoritmos evaluados se presentan en las Figs. 4, 5 y 6 para las instancias de

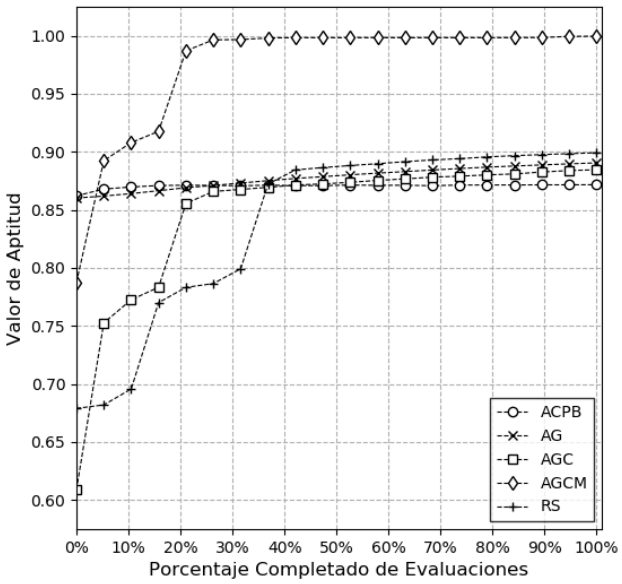

Fig. 5. Evolución del valor de aptitud para la instancia de linfoma

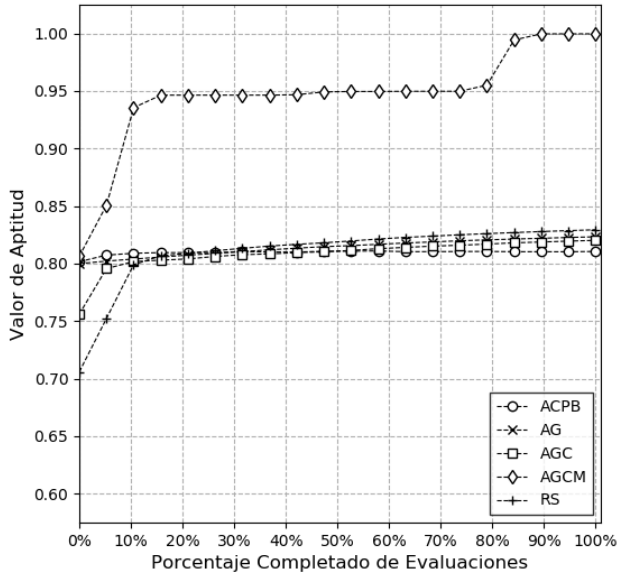

Fig. 6. Evolución del valor de aptitud para la instancia de leucemia

colon, linfoma y leucemia, respectivamente. Para cada figura el eje x muestra el porcentaje de evaluaciones completadas, frente al valor de aptitud obtenido por la mejor solución en ese momento.

Como se observa en las figuras, los algoritmos AGCM, RS y AG logran alcanzar casi el 100% antes de completar el 50% de las evaluaciones. En el caso del AGC y ACPB, convergen hacia el valor óptimo más lento. ACPB muestra una tendencia similar de crecimiento pero está sujeta a un estancamiento temprano del cual no logra escapar. Los comportamientos del AGCM, RS y AG se presentan como los algoritmos que mejor logran alcanzar un valor óptimo del problema. Las Figs. 5 y 6 demuestran comportamientos totalmente diferentes. El AGCM logra escalar a valores óptimos de manera rápida antes de alcanzar el 40% para la instancia de linfoma y antes del 95% para leucemia. EL resto de algoritmos tienden a producir mejoras no significativas quedando muy lejos del óptimo o directamente estancándose.

TABLA III
TEST DE WILCOXON PARA LOS VALORES DE APTITUD DE LAS INSTANCIAS DE COLON, LINFOMA Y LEUCEMIA

	AGC	AG	AGCM	RS
ACPB	▽ ▽ ▽	▽ ▽ ▽	▽ ▽ ▽	▽ ▽ ▽
AGC		▽ ▽ ▽	▲ ▽ ▽	▽ ▽ ▽
AG			▲ ▽ ▽	▲ ▽ ▽
AGCM				▽ ▲ ▲

La Tabla III muestra el detalle del análisis estadístico confirmando los resultados mostrados anteriormente por el AGCM. Los valores obtenidos se presentan en forma de tabla, como una comparación algoritmo a algoritmo para cada instancia probada. Un triángulo negro hacia arriba (▲) muestra que la configuración de la fila obtiene valores estadísticamente más altos que la configuración de la columna. Por el contrario, un triángulo blanco hacia abajo (▽) indica que la configuración de la fila obtiene valores estadísticamente más bajos que la configuración de la columna. Si no se encuentran diferencias significativas, el lugar se completa con un guión (–).

El análisis exhibido en la Tabla III corrobora que el AGCM tiene una diferencia estadística significativa con el resto de los algoritmos en la mayoría de los casos (salvo con el RS y el AG para la instancia de colon). En segundo lugar queda el RS y luego el AG con una diferencia significativa del 75% y 67% de las comparaciones, respectivamente. Esto indica que a medida que escala el problema, el AGCM sigue obteniendo resultados competitivos en comparación al resto de ellos. Podemos afirmar entonces que el AGCM presenta un desempeño superior.

La Tabla IV y la Tabla V muestran el valor mínimo, máximo y promedio de número de características seleccionadas y la precisión de clasificación obtenidos por cada algoritmo en cada instancia. En la Tabla IV, se observa que el menor número de características es obtenido, en las instancias linfoma y leucemia, por el AGCM seguido por el RS. Para la instancia de colon, casi todos los algoritmos obtienen un número de características similar. Para las instancias de linfoma y leucemia se observa una gran diferencia entre la cantidad de características seleccionadas por el AGCM y el RS con respecto al resto de algoritmos. Si tomamos el valor mínimo de características obtenidas, el AGCM utiliza un 4%, 1% y 1.5% para las instancias de colon, linfoma y leucemia respectivamente, lo cual indica que el AGCM realiza un excelente trabajo encontrando características de gran importancia para la clasificación.

Los valores obtenidos en la Tabla V indican que para la instancia de colon todos, al menos una vez, logran el 100% de precisión, en cambio para linfoma solo el AGCM y el RS llegan a obtener el 100% de precisión. En la instancia de leucemia, solamente el AGCM alcanza al 100% de precisión. El resto de algoritmos logran obtener valores de clasificación mínimo, máximo y promedio muy similares.

Las Figs. 7, 8, y 9 comparan la evolución de los números de características y la precisión de clasificación a medida que se van completando las evaluaciones de los distintos algoritmos para la instancia de colon, linfoma y leucemia,

TABLA IV
VALOR MÁXIMO, MÍNIMO Y PROMEDIO DE NÚMERO DE CARACTERÍSTICAS PARA CADA INSTANCIA Y ALGORITMO

Alg.	Colon			Linfoma			Leucemia		
	Min	Max	Avg	Min	Max	Avg	Min	Max	Avg
ACPB	37.00	65.00	53.05	1049.00	1158.00	1100.40	2243.00	2481.00	2342.80
AG	4.00	8.00	4.60	516.00	586.00	538.75	1526.00	1662.00	1577.35
AGC	5.00	35.00	11.15	597.00	666.00	643.25	1660.00	1879.00	1734.70
AGCM	9.00	19.00	12.60	31.00	103.00	51.50	88.00	423.00	175.70
RS	2.00	8.00	5.30	237.00	333.00	278.65	1142.00	1250.00	1185.90

TABLA V
VALOR MÁXIMO, MÍNIMO Y PROMEDIO DE PRECISIÓN PARA CADA INSTANCIA Y ALGORITMO

Algoritmo	Colon			Linfoma			Leucemia		
	Min	Max	Avg	Min	Max	Avg	Min	Max	Avg
ACPB	1.00	1.00	1.00	0.90	0.90	0.90	0.83	0.89	0.84
AG	1.00	1.00	1.00	0.90	1.00	0.91	0.83	0.89	0.84
AGC	1.00	1.00	1.00	0.90	1.00	0.91	0.83	0.89	0.84
AGCM	1.00	1.00	1.00	1.00	1.00	1.00	0.89	1.00	0.96
RS	0.87	1.00	0.98	0.80	1.00	0.92	0.83	0.89	0.84

respectivamente. El número de características se muestra en el eje y de la izquierda, el porcentaje de precisión alcanzado se muestra en el eje y de la derecha y el porcentaje de evaluaciones realizadas por cada algoritmo se presenta en el eje x. En general, se observa que el número de características seleccionadas disminuye a medida que aumentan las evaluaciones, mientras que la precisión aumenta con el incremento de la evaluación.

La Fig. 7 muestra que el AGCM presenta un escalado suave llegando al 100% de precisión de clasificación al completar el 90% de evaluaciones. El resto de algoritmos quedan entre el 90% y el 95% de precisión presentando también un escalado continuo de la mejor solución. Por otro lado, la reducción del número de características decrece de manera continua para el caso de AGCM, AG y RS. Si bien todos reducen el número de características de manera continua y logran llegar al 100% de precisión, los que reducen más el número utilizado son el AG y el AGCM, mientras que el ACPB es el que utiliza más características para llegar al 100% de precisión.

Para la Fig. 8 el AGCM alcanza valores de clasificación del 100% de precisión (al 30% aproximadamente de las evaluaciones) y el número de características seleccionadas va decreciendo de manera rápida logrando reducir de forma drástica el número de características antes del 40% de evaluaciones. El porcentaje de precisión para el resto algoritmos esta relegado a aproximadamente sólo el 85% y el número de características seleccionadas está en el rango de mil.

La Fig. 9 indica que el AGCM logra alcanzar o quedar muy cerca del 100% de precisión al completar aproximadamente el 80% de evaluaciones. El resto de algoritmos se mantiene por debajo del 90% durante todo el proceso de ejecución. Con respecto al número de características seleccionadas, todos los algoritmos evolucionan de manera similar al principio, pero AGCM se separa del resto alrededor del 20% de las evaluaciones, logrando el número mínimo de características.

E. Comparación con Resultados de la Literatura

A fin de comparar el AGCM propuesto con otros métodos de la literatura para el problema de selección de características,

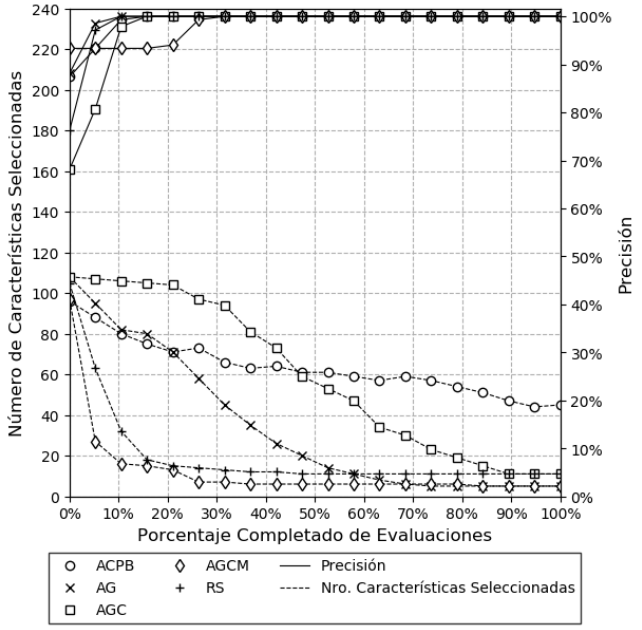

Fig. 7. Evolución del numero de características y la precisión de clasificación para la instancia colon

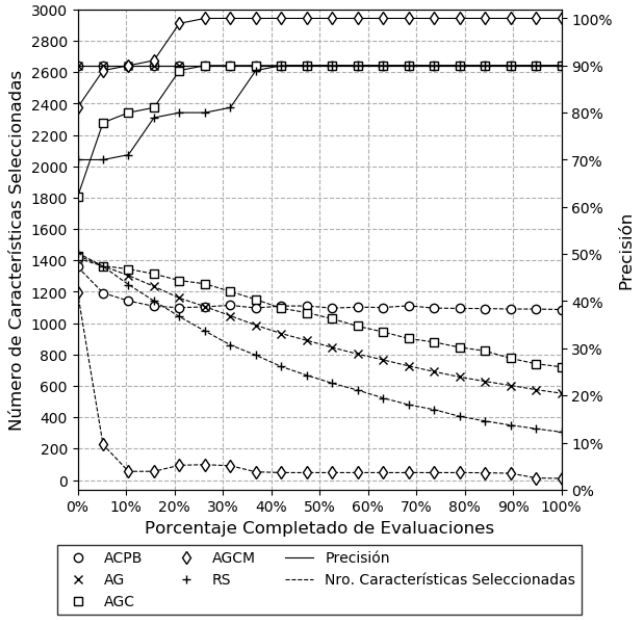

Fig. 8. Evolución del numero de características y la precisión de clasificación para la instancia linfoma

se han agregado los resultados reportados en [33] y [34]. En particular, en [33] utilizan la restricción de probar con un número máximo de características seleccionadas, por ello, los mejores valores los obtienen con un valor máximo de 50 en todas las instancias. Los autores informan el número de evaluaciones utilizadas como 2500 y 20000 evaluaciones, respectivamente. También, se presentan los resultados de [35]–[37] aunque en estos trabajos no informan explícitamente cuántas iteraciones o evaluaciones han realizado para obtener sus resultados. Los algoritmos utilizados por cada uno de ellos

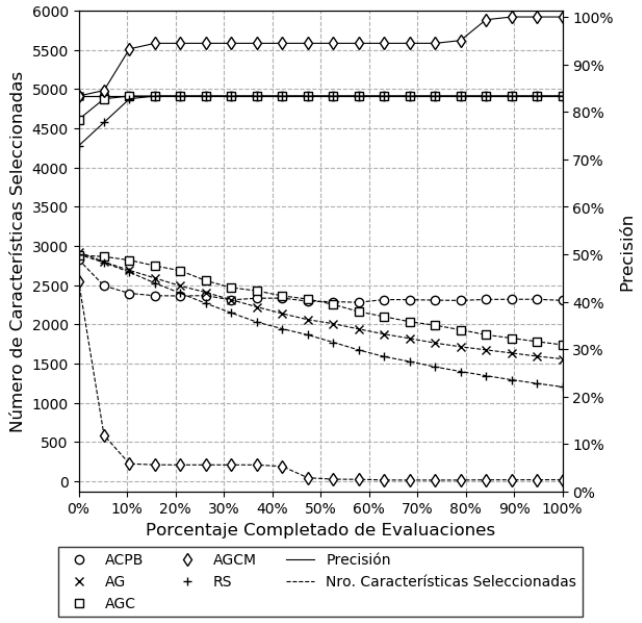

Fig. 9. Evolución del numero de características y la precisión de clasificación para la instancia leucemia

son:

- algoritmo genético adaptativo (modificación de probabilidad de cruce y mutación a medida que la evolución avanza) con técnica de maximización de la información mutua condicional (AGA+MIMC) [33].
- algoritmo genético con técnica de maximización de la información mutua condicional (AG+MIMC) [34].
- algoritmo genético (AG) [35].
- algoritmo de mono araña (AMA) [36].
- algoritmo de rebaño de kril binario con técnica de ganancia de información (ARKB+GI) [37].

La Tabla VI muestra la comparación con estos trabajos. La primera columna corresponde al algoritmo. Luego, para cada instancia se presenta la precisión máxima obtenida (columnas 2, 4 y 6) y el número de características seleccionadas (columnas 3, 5 y 7).

Se puede observar que nuestra aproximación presenta un excelente nivel de clasificación comparado con el resto de algoritmos teniendo en cuenta el número de evaluaciones realizadas. Con respecto al número de características seleccionadas, se puede observar que para la instancia colon los valores obtenidos son menores en comparación al resto salvo para AMA. Para Linfoma, AGCM no logra superar a los valores obtenidos por los dos mejores algoritmos, quedando muy cerca del segundo algoritmo con menos características seleccionadas (AG+MIMC) aunque AG+MIMC presenta un nivel menor de precisión a la hora realizar la clasificación. Finalmente para Leucemia, las características seleccionadas son valores cercanos al valor mínimo presentado en esta tabla obtenido por el ARKB+GI. En resumen, AGCM obtiene mejores resultados en comparación a AGA+MIMC, AGMIMC y AG. No logra superar a AMA aunque esto puede deberse al número de evaluaciones, las cuales no son reportadas en el trabajo [36]. Similar ocurre con ARKB+GI con la instancia leucemia aunque se observa que el AGCM lo supera ampliamente en colon.

TABLA VI
PRECISIÓN ALCANZADA Y NÚMERO DE CARACTERÍSTICAS PARA CADA INSTANCIA Y ALGORITMO DE LA LITERATURA

Algoritmo	Colon		Linfoma		Leucemia	
	Prec.	NoC	Prec.	NoC	Prec.	NoC
AGCM	1.00	9.00	1.00	31.00	1.00	88.50
AGA+MIMC [33]	0.87	50.00	0.97	50.00	0.87	50.00
AG+MIMC [34]	0.82	19.00	0.82	26.00	-	-
AG [35]	0.94	158.00	0.93	164.00	0.94	146.00
AMA [36]	1.00	6.00	1.00	6.00	1.00	8.00
ARKB+GI [37]	0.96	17.10	-	-	1.00	4.10

F. Análisis de Tiempo de Ejecución

El tiempo de ejecución promedio en segundos para cada algoritmo es presentado en la Tabla VII. Se observa que los mejores tiempos (más cortos) son obtenidos por el RS seguido del AG. En el caso del AGC y AGCM obtienen tiempos cercanos a los del AG.

Para la instancia de colon, los tiempos de ejecución obtenidos son similares entre los algoritmos. Para las otras instancias las diferencias en los tiempos de ejecución del algoritmo no son exageradas con respecto al mejor tiempo obtenido.

En general, si se considera el desempeño, precisión de clasificación y menor número de características seleccionadas, se puede afirmar que AGCM exhibe el comportamiento más robusto, pudiendo alcanzar o estar cerca del óptimo en un tiempo de ejecución rápido.

TABLA VII
VALOR PROMEDIO Y DESVIACIÓN ESTÁNDAR DEL TIEMPO DE EJECUCIÓN (SEGUNDOS) PARA LAS INSTANCIAS DE COLON, LINFOMA Y LEUCEMIA

Algoritmo	**Colon**		**Linfoma**		**Leucemia**	
	Tiempo	Desv. St.	Tiempo	Desv. St.	Tiempo	Desv. St.
ACPB	56.70	±5.71	313.00	±28.30	717.00	±72.50
AGC	84.70	±38.10	271.00	±61.00	601.00	±122.00
AG	43.10	±3.96	153.00	±18.30	361.00	±36.40
AGCM	37.60	±5.20	196.00	±33.90	394.00	±60.40
RS	43.60	±4.73	143.00	±32.20	325.00	±32.80

V. CONCLUSIÓN

El presente trabajo propone una variante memética entre un algoritmo genético celular y una búsqueda local diseñada especialmente para el problema de selección de características. Primero se han filtrado las características que tienen una alta relación entre ellas y luego se ha aplicado el algoritmo memético. Para evaluar la precisión de cada solución se utiliza sobre el conjunto de características reducidas el clasificador k-VC. El modelo propuesto se ha evaluado para tres conjuntos de datos de clase binaria. El rendimiento del AGCM ha sido probado comparándolo con otras técnicas utilizadas intensamente en la literatura para este tipo de problemas. De los resultados experimentales se observa que el esquema propuesto supera a otras técnicas tanto en la precisión obtenida

como el número de características seleccionadas en un tiempo de ejecución menor.

Los resultados demostraron la superioridad del enfoque propuesto. Se observó que el operador de búsqueda local incrementa la capacidad del algoritmo base para explorar y explotar las soluciones en este problema de selección de características. Los resultados cuantitativos y cualitativos también indicaron que la velocidad de convergencia del algoritmo propuesto a valores óptimos es alta, lo que resultó en la selección precisa de características relevantes para clasificar de forma correcta las instancias evaluadas. Por lo tanto, consideramos que el algoritmo genético celular memético es una propuesta robusta que puede ser utilizada en este tipo de problemas.

Como trabajo futuro queda ampliar a otras instancias la aplicación de nuestra propuesta. Además, dado que en este trabajo se utilizó un solo clasificador (k-VC) para evaluar los algoritmos, proyectamos analizar el impacto de utilizar otros clasificadores. También queremos evaluar distintas formas de separar los datos en conjuntos test-entrenamiento con el fin de encontrar la más apropiada para este tipo de problemas.

AGRADECIMIENTOS

Los fondos para realizar esta investigación provienen en parte de la Universidad Nacional de Cuyo a través del Proyecto Tipo 1 Código B081 y en parte de la Universidad Nacional del Sur, Secretaría General de Ciencia y Tecnología con el proyecto PGI 24/N042.

REFERENCES

[1] X. L. Du, C. R. Key, C. Osborne, J. D. Mahnken, and J. S. Goodwin, "Discrepancy between consensus recommendations and actual community use of adjuvant chemotherapy in women with breast cancer," *Annals of Internal Medicine*, vol. 138, no. 2, pp. 90–97, 2003.

[2] S. Liu, C. Xu, Y. Zhang, J. Liu, B. Yu, X. Liu, and M. Dehmer, "Feature selection of gene expression data for cancer classification using double RBF-kernels," *BMC Bioinformatics*, vol. 19, no. 1, p. 396, Oct 2018.

[3] N. Mekour, R. Hamou, and A. Amine, "filter/wrapper methods for gene selection and classification of microarray dataset," pp. 65–80, 07 2019.

[4] J. Apolloni, G. Leguizamón, and E. Alba, "Two hybrid wrapper-filter feature selection algorithms applied to high-dimensional microarray experiments," *Applied Soft Computing*, vol. 38, pp. 922–932, 2016.

[5] A. Anaissi, P. J. Kennedy, and M. Goyal, "Feature selection of imbalanced gene expression microarray data," in *2011 12th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*. IEEE, 2011, pp. 73–78.

[6] W. Siedlecki and J. Sklansky, "On automatic feature selection," in *Handbook of Pattern Recognition and Computer Vision*. World Scientific, 1993, pp. 63–87.

[7] H. Liu, R. Setiono *et al.*, "A probabilistic approach to feature selection-a filter solution," in *ICML*, vol. 96. Citeseer, 1996, pp. 319–327.

[8] N. Zhong, J. Dong, and S. Ohsuga, "Using rough sets with heuristics for feature selection," *Journal of intelligent information systems*, vol. 16, no. 3, pp. 199–214, 2001.

[9] Y. Li, G. Wang, H. Chen, L. Shi, and L. Qin, "An ant colony optimization based dimension reduction method for high-dimensional datasets," *Journal of Bionic Engineering*, vol. 10, no. 2, pp. 231–241, 2013.

[10] J. Jona and N. Nagaveni, "Ant-cuckoo colony optimization for feature selection in digital mammogram," *Pakistan journal of biological sciences*, vol. 17, no. 2, p. 266, 2014.

[11] J. H. Holland, "Genetic algorithms," *Scientific American*, Jul. 1992.

[12] E. Alba and B. Dorronsoro, *Cellular Genetic Algorithms*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[13] C. A. Harrington, C. Rosenow, and J. Retief, "Monitoring gene expression using dna microarrays," *Current Opinion in Microbiology*, vol. 3, no. 3, pp. 285 – 291, 2000.

[14] C. Cotta, L. Mathieson, and P. Moscato, *Memetic Algorithms*. Cham: Springer International Publishing, 2017, pp. 1–32.

[15] C. M. Affonso and R. V. da Silva, "Demand side management of a residential system using Simulated Annealing," *IEEE Latin America Transactions*, vol. 13, no. 5, pp. 1355–1360, May 2015.

[16] J. Kennedy and R. Eberhart, *Swarm intelligence*. San Francisco: Morgan Kaufmann Publishers, 2001.

[17] M. Xiong, W. Li, J. Zhao, L. Jin, and E. Boerwinkle, "Feature (gene) selection in gene expression-based tumor classification," *Molecular genetics and metabolism*, vol. 73, no. 3, pp. 239–247, 2001.

[18] N. A. Al-Thanoon, O. S. Qasim, and Z. Y. Algamal, "A new hybrid firefly algorithm and particle swarm optimization for tuning parameter estimation in penalized support vector machine with application in chemometrics," *Chemometrics and Intelligent Laboratory Systems*, vol. 184, pp. 142 – 152, 2019.

[19] J. Xu and F. Yan, "Hybrid nelder–mead algorithm and dragonfly algorithm for function optimization and the training of a multilayer perceptron," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3473–3487, sep 2018.

[20] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302 – 312, 2017.

[21] S. K. Baliarsingh, W. Ding, S. Vipsita, and S. Bakshi, "A memetic algorithm using emperor penguin and social engineering optimization for medical data classification," *Applied Soft Computing*, vol. 85, p. 105773, 2019.

[22] P. Hansen and N. Mladenović, *Variable Neighborhood Search*. Cham: Springer International Publishing, 2018, pp. 759–787.

[23] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[24] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences*, vol. 96, no. 12, pp. 6745–6750, 1999.

[25] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu *et al.*, "Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling," *Nature*, vol. 403, no. 6769, pp. 503–511, 2000.

[26] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri *et al.*, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *science*, vol. 286, no. 5439, pp. 531–537, 1999.

[27] M. Walesiak, A. Dudek, and M. A. Dudek, "clustersim package," 2011.

[28] M. Kuhn, "CARET package," *Journal of Statistical Software*, vol. 28, no. 5, pp. 1–26, 2008.

[29] K. G. Calkins, "Applied statistics—lesson 5. correlation coefficients; 2005," 2019.

[30] E. Talbi, L. Jourdan, J. Garcia-Nieto, and E. Alba, "Comparison of population based metaheuristics for feature selection: Application to microarray data classification," in *2008 IEEE/ACS International Conference on Computer Systems and Applications*, March 2008, pp. 45–52.

[31] A. Benitez-Hidalgo, A. J. Nebro, J. Garcia-Nieto, I. Oregi, and J. Del Ser, "jMetalPy: a python framework for multi-objective optimization with metaheuristics," *arXiv preprint arXiv:1903.02915*, 2019.

[32] C. Heumann, M. Schomaker, and Shalabh, *Introduction to Statistics and Data Analysis*. Springer International Publishing, 2016.

[33] A. K. Shukla, P. Singh, and M. Vardhan, "A two-stage gene selection method for biomarker discovery from microarray data for cancer classification," *Chemometrics and Intelligent Laboratory Systems*, vol. 183, pp. 47–58, Dec. 2018.

[34] ——, "A new hybrid feature subset selection framework based on binary genetic algorithm and information theory," *International Journal of Computational Intelligence and Applications*, vol. 18, no. 03, p. 1950020, Sep. 2019.

[35] A. Daisy and R. Porkodi, "Classification of human cancer diseases gene expression profiles using genetic algorithm by integrating protein protein interactions along with gene expression profiles," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*. IEEE, Mar. 2018.

[36] G. R. Kancharla, N. R. Eluri, S. Dara, and N. Ansari, "An efficient algorithm for feature selection problem in gene expression data: A spider monkey optimization approach," *SSRN Electronic Journal*, 2019.

[37] G. Zhang, J. Hou, J. Wang, C. Yan, and J. Luo, "Feature selection for microarray data classification using hybrid information gain and a modified binary krill herd algorithm," *Interdisciplinary Sciences: Computational Life Sciences*, May 2020.

Matías Gabriel Rojas is an informatic engineer graduated at the Gastón Dachary University in Posadas, Misiones, Argentina. His research topic belongs to bioinformatics algorithms area, focusing on optimization algorithms for feature selection. ORCID: 0000-0003-3881-0888

Ana Carolina Olivera is an Adjunct Researcher at National Council of Scientifics and Technological Researches from the Ministerio de Ciencia y Tecnología de la Nación, Argentine. Dr. in Computer Science from Universidad Nacional del Sur. She is an Associate Professor at the Facultad de Ingeniería from Universidad Nacional de Cuyo. Her research focuses on metaheuristics and optimization in complex problems. She has published several book chapters, articles in indexed journals and proceedings of refereed international conferences. ORCID: 0000-0001-7825-1959

Jessica Andrea Carballido is an Adjunct Researcher at National Council of Scientifics and Technological Researches from the Ministerio de Ciencia y Tecnología de la Nación, Argentine. Dr. in Computer Science from Universidad Nacional del Sur. She is an Associate Professor at the Dpto. de Cs. e Ing. de la Computación from Universidad Nacional del Sur. Her research focuses on evolutionary computation applied to bioinformatics, mainly for cancer studies from microarray and RNA-seq experiments. She has published several book chapters, articles in indexed journals and proceedings of refereed international conferences. ORCID: 0000-0001-6284-1049

Pablo Javier Vidal is an Adjunct Professor at the Universidad Nacional de Cuyo, and at the Universidad Nacional de la Patagonia Austral, Argentine. Dr. in Software Engineering and Artificial Intelligence, from Universidad de Málaga, Spain. He is an Assistant Researcher at National Council of Scientifics and Technological Researches from the Ministerio de Ciencia y Tecnología de la Nación, Argentine. His main research topics are: parallel and distributed computing, bioinformatics and metaheuristics. ORCID: 0000-0001-6502-8010