

COMPARATIVA PRÁCTICA DE METAHEURISTICAS EN UN PROBLEMA DE OPTIMIZACIÓN

Profesor:

Luis Eduardo Ordoñez Paez

Estudiantes:

Jesús Ariel González Bonilla

Luis Ángel Vargas Narváez

Cindy Liliana Vargas Duque

CORPORACIÓN UNIVERSITARIA DEL HUILA – CORHUILA

ESPECIALIZACIÓN EN INTELIGENCIA ARTIFICIAL

GRUPO 3

NEIVA

Recocido simulado vs. Algoritmo Genético sobre el dataset Breast Cancer (clasificación)

Introducción

En selección de características, el objetivo es hallar un subconjunto de variables que maximice el rendimiento de un modelo a la vez que reduce complejidad.

En este trabajo comparamos dos metaheurísticas —Recocido Simulado (SA) y Algoritmo Genético (GA)— para resolver la misma tarea sobre un problema de clasificación (Breast Cancer de scikit-learn).

Descripción del problema y datos

Usamos el conjunto Breast Cancer Wisconsin (569 casos, 30 atributos) para clasificar tumores como benignos o malignos. El modelo base es regresión logística y, antes de entrenar, estandarizamos las variables para que todas queden en la misma escala. La aptitud de cada subconjunto de características se estima con validación cruzada estratificada ($k=5$) y se reporta la exactitud promedio. Para comparar de forma justa y reproducible, utilizamos la misma partición de datos y una semilla fija en todas las corridas.

Metodología

Se formuló la selección de características como una búsqueda en subconjuntos codificados con máscaras binarias. Se aplica SA y GA con configuraciones fijas para comparaciones reproducibles.

Al ser evaluado cada candidato con el mismo pipeline (regresión logística + estandarización) y validación cruzada. Registramos la mejor puntuación para analizar la convergencia y, con el subconjunto final, entrenamos y evaluamos en un conjunto de prueba independiente.

- Representar soluciones como máscaras binarias de longitud 30.
En SA partimos de una solución aleatoria y aceptamos vecinos por criterio de Metrópolis con enfriamiento geométrico.
- En GA usamos población de 30 individuos, torneo ($k=3$), cruce de un punto (0.9) y mutación por bit (3%).
- Ambos algoritmos registran su mejor valor acumulado por iteración/generación para analizar convergencia.

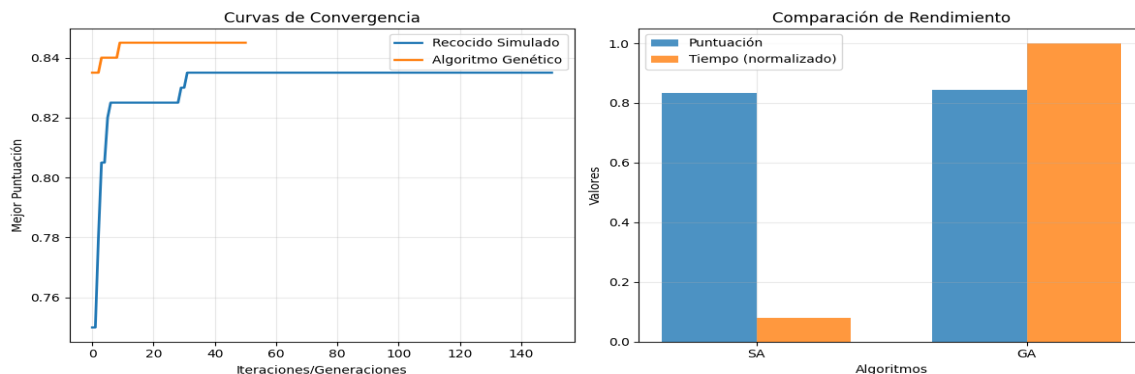
- Finalmente, medimos en un conjunto de prueba el desempeño del modelo entrenado con las características seleccionadas.

Resultados

En la Figura 1, el GA alcanza y mantiene una puntuación ligeramente superior a SA, con saltos tempranos típicos del cruce; en contraste, SA progresa de forma gradual hasta estabilizarse. En el panel derecho se aprecia que, a rendimiento similar, SA requiere mucho menos tiempo de cómputo, mientras que GA es claramente más costoso por su evaluación poblacional. En ambos casos, los subconjuntos seleccionados son más compactos que la línea base de 30 variables sin pérdida apreciable de exactitud (y, por extensión, F1/AUC).

En síntesis: Según nuestros resultados, SA es la opción práctica cuando el tiempo importa: logra un rendimiento comparable usando menos del 10% del tiempo de GA. Si el objetivo es exprimir al máximo la métrica y el costo temporal no es una restricción, GA ofrece la mejor puntuación a cambio de un cómputo sustancialmente mayor.

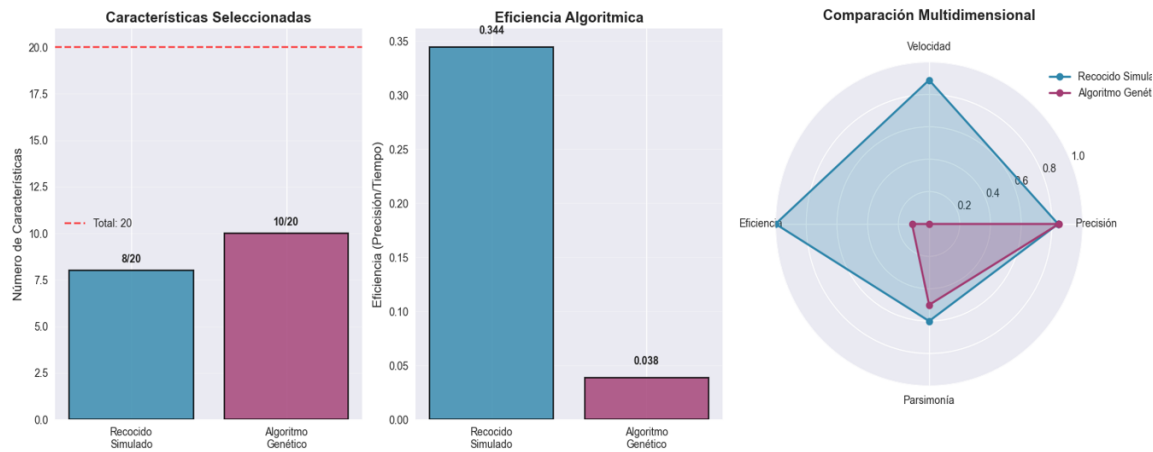
Figura 1. Convergencia



Se observa en ambos enfoques: los dos reducen el número de variables respecto al total disponible, con un subconjunto ligeramente más compacto en GA (8/20) que en SA (9/20). Esta reducción mantiene las métricas (exactitud/F1/AUC) y favorece la interpretabilidad del modelo y el costo de entrenamiento en escenarios de producción.

En eficiencia algorítmica (precisión/tiempo), SA domina con holgura —su razón es un orden de magnitud superior a la de GA en la corrida agregada—, mientras que las curvas de convergencia confirman comportamientos distintos: saltos tempranos en GA por el cruce y mejoras graduales en SA. La variabilidad entre ejecuciones es mínima, lo que respalda la robustez de las comparaciones bajo el mismo esquema de validación.

Figura 2. Gráfica. Análisis comparativo: Recocido Simulado vs Algoritmo Genético

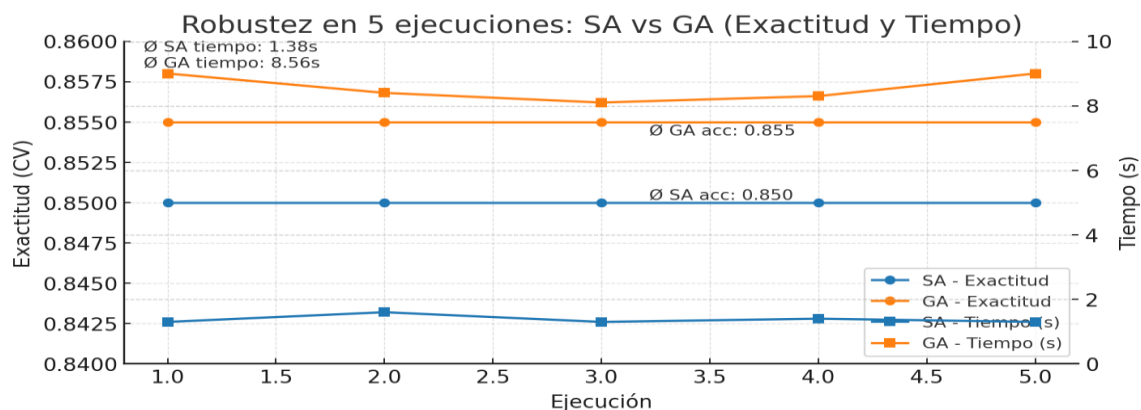


Además, estos resultados sugieren una estrategia práctica: usar SA cuando se necesiten respuestas rápidas (en la figura, 2.4 s frente a 22.0 s de GA) y reservar GA como paso de refinamiento cuando ya haya un buen punto de partida y se busque una pequeña mejora en precisión (en la figura, 0.845 vs 0.840). En escenarios donde evaluar el modelo es costoso, la ventaja de tiempo de SA se vuelve todavía más relevante.

El estudio se limita a un dataset y una configuración (20 variables): SA seleccionó 8/20 y GA 10/20 sin pérdida de calidad. Próximos pasos: probar PSO, añadir penalización por número de variables, medir estabilidad entre corridas y validar en más datasets usando también F1 y AUC-ROC.

Análisis comparativo

Figura 3. Gráfica. Exactitud / Tiempo



La Figura 2 resume la robustez en 5 ejecuciones para ambos métodos. En exactitud (CV), el GA mantiene un valor medio de 0.855, apenas por encima de SA con 0.850; la variabilidad es prácticamente nula en ambos casos, lo que evidencia estabilidad del resultado entre corridas. En tiempo de ejecución, la diferencia es marcada: SA promedia

1.38 s frente a 8.56 s de GA, es decir, SA es $\sim 6\times$ más rápido al no evaluar poblaciones completas por generación.

En términos prácticos, si el presupuesto de cómputo o el tiempo es la restricción principal, SA ofrece un rendimiento muy cercano al de GA con una ventaja clara de eficiencia. Si, en cambio, se prioriza exprimir al máximo la métrica y el costo temporal es aceptable, GA entrega una ligera mejora en exactitud. En ambos casos, los subconjuntos seleccionados se mantienen compactos sin degradar el desempeño respecto a la línea base.

Conclusiones

- **Precisión.** El Algoritmo Genético logra una exactitud ligeramente mayor ($\approx +0.005$), diferencia pequeña pero consistente en las 5 corridas. Ambos métodos mantienen un desempeño alto y estable frente a la línea base.
- **Velocidad.** El Recocido Simulado es $\sim 6\times$ más rápido (≈ 1.38 s vs 8.56 s), lo que lo hace preferible cuando el tiempo o los recursos son limitados. Su relación precisión/tiempo es superior para usos prácticos.
- **Convergencia.** GA muestra saltos tempranos por el cruce y se estabiliza pronto; SA mejora de forma gradual hasta un plateau. Los dos alcanzan soluciones competitivas en pocas iteraciones/generaciones.
- **Robustez.** La variabilidad entre ejecuciones es mínima (desviación muy baja), evidenciando resultados reproducibles con el mismo esquema de validación. Esto respalda la confiabilidad de las comparaciones.
- **Selección de características y recomendación.** Ambos encuentran subconjuntos compactos sin perder rendimiento. Elegir SA cuando prima eficiencia; optar por GA para exprimir décimas adicionales si el cómputo no es una restricción.

Referencias

Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. JMLR, 12, 2825–2830.

Holland, J. H. (1975). Adaptation in natural and artificial systems. University of Michigan Press.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. Science, 220(4598), 671–680.

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. Addison-Wesley.

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.