

**UNIVERSIDAD NACIONAL DEL ALTIPLANO**  
**FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y**  
**SISTEMAS**  
**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**APLICACIÓN DE LOS ALGORITMOS GENÉTICOS EN EL APRENDIZAJE DE**  
**MÁQUINA**

**CURSO:**  
**APRENDIZAJE DE MÁQUINA**

**DOCENTE:**  
**FERNANDEZ CHAMBI MAYENKA**

**SEMESTRE: NOVENO**

**INTEGRANTES:**  
**AQUINO SANDOVAL KENYA XIOMARA**  
**ARACAYO MAMANI JHON MARCO**  
**CANAZA PAUCARA JUAN DIEGO**  
**CASTRO CCALLO VANESSA YHOSELIN**

**PUNO - PERÚ**

**2025 – II**

# APLICACIÓN DE ALGORITMOS GENÉTICOS EN APRENDIZAJE AUTOMÁTICO

## Objetivo:

Demostrar mediante ejemplos ejecutables cómo los Algoritmos Genéticos (AG) pueden aplicarse en tres áreas: (1) selección de variables, (2) optimización de hiperparámetros y (3) diseño de arquitecturas de redes neuronales. El trabajo se estructura en tres módulos independientes.

## Enfoque general:

Se implementó un AG que mantiene una población de soluciones, evalúa su aptitud y aplica operadores de selección, cruce, mutación y elitismo. El ciclo se adaptó a cada problema modificando la codificación del cromosoma y la función de fitness.

- **Representación:**
  - FS: Vector binario (1 = variable seleccionada, 0 = descartada)
  - HPO: Vector de índices asociados a dominios discretos o mixtos
  - NE: Representación mixta para capas, neuronas, dropout y tasa de aprendizaje
- **Inicialización:** Población aleatoria con semilla fija para reproducibilidad
- **Fitness:** Precisión en validación cruzada con penalización por complejidad
- **Operadores:** Selección por torneo, cruce de un punto, mutación y elitismo
- **Terminación:** Número predefinido de generaciones

## 1) Selección de Características (FS)

Se codificó cada solución como vector binario. El fitness combina precisión en validación cruzada de regresión logística con penalización por número de variables.

## Evidencia generada:

- Lista de variables seleccionadas (selected\_features.csv)
- Evolución del fitness (fitness\_curve.png)
- Máscara booleana y métricas (bloque1\_selection.joblib)

**Conclusión:** El AG reduce dimensionalidad sin comprometer rendimiento, simplificando el modelo y facilitando su interpretación.

## 2) Optimización de Hiperparámetros (HPO) — Random Forest

Se optimizaron hiperparámetros de RandomForestClassifier: número de árboles, profundidad máxima, muestras mínimas y estrategia de max\_features.

### Evidencia generada:

- Hiperparámetros óptimos decodificados
- Precisión en validación cruzada del mejor individuo
- Curva de evolución del fitness

**Conclusión:** El AG explora eficientemente espacios mixtos sin búsquedas exhaustivas, encontrando combinaciones que equilibran sesgo y varianza.

## 3) Neuroevolución (NE) — MLP con Keras/TensorFlow

Se evolucionó la arquitectura de una MLP mediante cromosoma que define: número de capas ocultas (1-2), unidades por capa (32-128), dropout (0-0.5) y tasa de aprendizaje.

### Evidencia generada:

- Evolución del fitness (evolution\_progress.png)
- Resumen de la arquitectura ganadora

**Conclusión:** Emergen redes de 1-2 capas con 64-128 neuronas, dropout moderado y tasa de aprendizaje óptima, balanceando calidad y costo computacional.

### Conclusiones:

1. Un mismo marco evolutivo resolvió problemas de naturaleza distinta
2. Ventajas: Búsqueda global, manejo de espacios mixtos y fácil incorporación de penalizaciones
3. Coste computacional manejado mediante ejecuciones controladas
4. Total, reproducibilidad mediante requirements.txt y notebooks autoejecutables

**GITHUB:** <https://github.com/tamoil-0/genetic-ml.git>