

# Sistemas de Inteligencia Artificial

## Algoritmos Genéticos

Esteban Pintos (51048) - Cristian Pereyra (51190) - Matías De Santi (51051)

10 de Junio de 2013

### 1. Resumen

En el presente informe se detalla la resolución del trabajo práctico número 3 de la materia. El mismo consistió en el desarrollo de un algoritmo genético para encontrar los mejores pesos para la red neuronal construida en el trabajo práctico anterior. Dicha red neuronal tenía por objetivo aproximar una serie de tiempo.

### 2. Introducción

Para poder obtener los pesos para una red neuronal multicapa, se desarrolló un algoritmo genético cuyos parámetros son fácilmente configurables. Estos parámetros son:

Tamaño de la población

Brecha Generacional

Número máximo de generaciones

Probabilidad de permutación

Probabilidad de cruza

Probabilidad de entrenar un individuo con back-propagation

Método de selección

Método de reemplazo

Método de cruce

### 3. Modelado del problema

#### 3.1. Representación de un individuo

Dado que para el trabajo práctico anterior se desarrolló una red neuronal con todas las funciones necesarias, para este trabajo se decidió representar a

la **población** como un arreglo de redes neuronales. Cada red es inicializada con pesos aleatorios.

Debido a que el algoritmo genético fue desarrollado para encontrar los pesos correctos para la red neuronal, a la hora de realizar las operaciones de selección, reemplazo, mutación, etc. se extrajo de cada individuo la matriz de pesos y se la transformó en un arreglo. De esta manera, un **cromosoma** es el arreglo de pesos correspondiente a una red neuronal y un **locus** es un peso puntual de la red.

### 4. Implementación del Algoritmo Genético

El algoritmo implementado sigue el siguiente flujo:

- Inicializar población
- Mientras no se haya alcanzado la condición de corte:
  - Evaluar la función de fitness
  - Seleccionar individuos y aparear
  - Recombinar y mutar
  - Reemplazar

A continuación se detalla la implementación de cada uno de los pasos del algoritmo.

#### 4.1. Función de fitness

En un principio se utilizó como función de fitness

$$\frac{1}{ECM} \quad (1)$$

donde ECM es el error cuadrático medio. Esta función tiene la propiedad de asignar un valor alto a aquellos valores cercanos al cero y a medida que dicho valor se aleja del cero, el valor asignado es cada vez más chico.

Luego, se probó con una función lineal

$$500 - ECM \quad (2)$$

Esta función asigna valores cercanos a 500 a aquellos individuos que tengan el ECM más pequeño. Sin embargo, la diferencia entre esta función y la mencionada en el párrafo anterior es evidente. La función 1 es más sensible ante pequeños cambios. Por ejemplo, si el ECM disminuye de 0.2 a 0.1, la función 1 duplicará su valor (de 5 a 10), mientras que la ecuación 3 cambiará su valor en 0.1. De esta manera, la ecuación 1 premia a aquellos individuos que han logrado disminuir su error cuadrático medio, aunque el cambio haya sido pequeño.

Finalmente, se optó por la función

$$\frac{1}{ECM^2} \quad (3)$$

como función de fitness ideal, ya que empíricamente notamos una aún mejor valoración para los cambios entre el error que la función inicial, la cual si bien asignaba valores buenos, normalmente perdía a los mejores individuos apenas eran introducidos a la población.

## 4.2. Métodos de reemplazo

Independientemente de cómo realiza su propósito, cada uno de los algoritmos de reemplazo que se explicarán a continuación respetan una interfaz: reciben los individuos de la población, la evaluación del fitness de cada individuo y retornan una nueva población.

### Método 1

Utilizando alguno de los métodos de selección que se explicarán luego, este método selecciona de a 2 individuos. Una vez seleccionados dichos individuos los aparea, recombina, muta y los entrena con back-propagation. Esta secuencia de acciones retorna dos individuos que pertenecerán a la nueva generación. Esta secuencia se repite hasta generar una nueva población con la misma cantidad de individuos que tenía la población original.

### Método 2

Este método es similar al anterior, con la diferencia que se seleccionan  $k$  individuos que pasarán directamente a la siguiente generación sin sufrir modificación alguna.

Los  $N-k$  individuos restantes generan  $N-k$  hijos que sufriran los mismos cambios que los individuos en el método anterior.

### Método 3

Este método funciona de la misma manera que el método 1. Se seleccionan 2 padres que darán lugar a 2 hijos (luego de sufrir el apareamiento, recombinación, mutación, etc.). Una vez que se generan  $k$  hijos, se juntan con los  $N$  y de los  $N+k$  se seleccionan  $N$  que formarán la siguiente generación.

## 4.3. Métodos de selección

Los métodos de reemplazo mencionados en la sección anterior utilizan métodos de selección para, valga la redundancia, seleccionar individuos de la población. Todos los métodos de selección implementados cumplen la misma interfaz: reciben un arreglo con las evaluaciones de los individuos y retornan un arreglo con los índices de los individuos seleccionados.

El algoritmo implementado puede ser configurado para utilizar cualquiera de los siguientes métodos de selección.

### Elite

Este método es el más sencillo de todos. Dado el arreglo con las evaluaciones, retorna los índices de las dos evaluaciones más altas.

### Ruleta

Este método de selección tiene en cuenta la evaluación que cada individuo obtuvo. La probabilidad  $p$  de elegir al individuo  $i$  se calcula de la siguiente manera:

$$\frac{f_i}{\sum_{j=1}^N f_j} \quad (4)$$

En consecuencia, aquellos individuos que obtuvieron un fitness más alto tienen más probabilidad de ser elegidos. Sin embargo, los individuos que peor fitness obtuvieron tienen posibilidad de ser elegidos y de esta manera el método difiere de elite.

Si se desean seleccionar  $m$  individuos, se generan  $m$  números aleatorios y por cada uno de ellos se elige a un individuo.

## Selección universal estocástica

Este método sigue la misma idea que ruleta, con la diferencia que se genera un único número al azar para elegir a los  $m$  individuos.

### Boltzman

El método de selección de Boltzman estipula que la probabilidad de ser elegido depende de una función no lineal del fitness y una "temperatura". De esta manera, en el inicio la temperatura es alta y por lo tanto todos los individuos tienen aproximadamente la misma posibilidad de ser elegidos, ofreciendo diversidad a la selección. A medida que el algoritmo avanza la probabilidad de elegir a cada individuo empieza a depender cada vez más de la evaluación del fitness, eligiendo a los individuos más aptos.

La función de probabilidad es la siguiente:

$$\frac{e^{\frac{f_i}{T}}}{\sum_{j=1}^m e^{\frac{f_j}{T}}} \quad (5)$$

Para valores de  $T$  grandes, es fácil ver que todos los términos valen 1 y por lo tanto la probabilidad de ser elegido es la misma para todos los individuos. A medida que  $T$  disminuye, la probabilidad de cada individuo empieza a depender cada vez más del valor de su fitness.

### Torneo

Este método toma dos individuos al azar y un número aleatorio. Si dicho número es menor a 0.75, entonces se selecciona al individuo más apto de los dos. Caso contrario, se selecciona al menos apto. Luego de dicha selección, ambos individuos son devueltos a la población y pueden volver a ser elegidos.

### Rank

Para este método se deben ordenar a los individuos según su fitness. Una vez ordenados, la probabilidad de elegir a uno de los individuos es proporcional a su posición en la lista.

En este método no importa si dos individuos con fitness consecutivo tienen una diferencia pequeña o grande en su fitness a la hora de calcular la probabilidad de ser seleccionados.

### Método mixto

Existe la posibilidad de configurar al programa para que utilice un método mixto de selección. Este

método mixto utiliza dos métodos de selección: selecciona una cantidad  $k$  (configurable) de individuos con un método y la cantidad restante con otro.

## 4.4. Métodos de cruce

Se desarrollaron cuatro métodos de cruce:

El método de **cruce de un punto** toma un locus al azar y a partir de ese locus, intercambia los siguientes entre los dos cromosomas. Análogamente, el método de cruce **de dos puntos** toma dos puntos al azar del cromosoma y entre dichos puntos intercambia los locus de los cromosomas.

Por otra parte, el cruce **anular** toma también dos puntos pero en este caso el segundo punto puede estar antes que el primero. En este caso se intercambian los locus desde el primer punto hasta el final del y luego desde el principio hasta el segundo punto.

Por último, el cruce **uniforme parametrizado** intercambia un locus en particular entre cromosomas con una probabilidad  $p$ .

## 4.5. Criterios de Corte

Los criterios de corte implementados fueron los siguientes:

- Máxima cantidad de generaciones: El algoritmo finaliza cuando se llega a una cantidad de generaciones  $n$ , siendo  $n$  parámetro del sistema.
- Estructura: El algoritmo finaliza cuando se detecta que un gran porcentaje  $p$  de la población no muta de generación en generación, es decir, que la cantidad de individuos iguales en una generación es mayor a  $p$ .
- Contenido: El algoritmo finaliza cuando se detecta un estancamiento en la mejor función de fitness al pasar las generaciones.
- Entorno al óptimo: Cuando se llega a una solución óptima, es decir, cuando la función de fitness supera un valor determinado.

### 4.5.1. Mutación

Una vez que se realiza el cruce entre individuos, existe una probabilidad de que cada individuo mute. Esto es importante dentro del algoritmo genético ya que introduce variaciones en la población. De todas maneras, la mutación no debe ocurrir tan frecuentemente de manera que afecte a todos los individuos de la población y así arruine los avances logrados.

Hay dos maneras de realizar la mutación: o bien se decide o no mutar un cromosoma y luego se elige un locus al azar para ser modificado, o se itera por todos los locus del cromosoma decidiendo en cada paso si mutar o no dicho locus. Lógicamente, la probabilidad de que mute un locus (en el segundo método de mutación) es considerablemente menor que la probabilidad de mutar un cromosoma (si se utiliza el primer método de mutación).

Nuestra implementación de mutación utiliza el segundo método, y en caso de tener que mutar un locus, se genera un peso al azar y se reemplaza al locus a mutar.

## 5. Resultados y análisis

Dada la cantidad de métodos de selección, reemplazo y crossover, existen una gran cantidad de combinaciones posibles para probar el algoritmo genético. A esto hay que agregarle que los parámetros de tamaño de población, brecha generacional, etc. son todos modificables, lo cual agranda considerablemente la cantidad de opciones.

Es por esto que para probar los resultados del algoritmo, se decidió probar inicialmente los 3 métodos de reemplazo con un único método de selección. De esta manera se fueron descartando algunas combinaciones, hasta llegar a tener ciertos candidatos. A partir de este punto se realizaron nuevas pruebas y éstas son las que se muestran en esta sección.

Si bien en un principio se pensó que utilizar únicamente el método elite de selección era un candidato a presentar los mejores resultados, luego de algunas pruebas fue descartado. Si bien en un principio el algoritmo avanzaba velozmente, llegaba un momento en el cual se veía estancado ya que la población tendía a parecerse al individuo más apto y no se introducían variaciones.

Otra cuestión a destacar es que el uso de los métodos de reemplazo, si bien probamos una amplia variante de métodos, el método de reemplazo 1 no nos brindó resultados favorables, ya que los individuos más aptos se perdían en el reemplazo. Y el reemplazo 3, en cambio, solía favorecer a largo plazo a solamente un tipo de individuo, ya que no favorecía mucho a las mutaciones o crossovers. Creemos que tal vez no hayamos encontrado los parámetros justos para estos tipos de reemplazos, y que puede ser necesario calibrar los parámetros acorde a cada uno.

Finalmente, tampoco notamos una gran diferencia

en los métodos de crossover, lo cual nos dejó en claro que puede ser necesario calibrar también la probabilidad de crossover para cada método, lo cual podría dar resultados distinguibles en cada caso particular.

### 5.1. Resultados Generales

Los resultados que se pueden observar en la tabla 1 fueron obtenidos utilizando una configuración 2 9 6 1 en la red neuronal (2 puntos de entrada, 2 capas ocultas con 9 y 6 neuronas respectivamente, y 1 neurona en la capa de salida) sobre una población de 30 individuos.

Como se puede observar el resultado obtenido por el algoritmo genético es bueno. El error es del orden de  $1,7 \times 10^{-2}$  y la generalización con respecto a la serie con la cual no se entrenó la red es del mismo orden, lo cual habla de una buena generalización. Esto había quedado pendiente en el trabajo práctico anterior ya que la red había sido sobreentrenada y por lo tanto la generalización era mala.

Otro resultado que se puede observar es que 2 de los 3 mejores resultados tienen a elite como método de selección. Si bien el método de elite sólo no trae buenos resultados, combinado con otros métodos produjo los mejores resultados.

## 6. Conclusiones

A partir de la experiencia obtenida en este trabajo consideramos a los algoritmos genéticos una buena forma de aprender a optimizar soluciones a problemas u obtener soluciones a problemas nuevos en sí.

Consideramos también que es necesario calibrar los algoritmos genéticos, ya que un sólo número puede hacer que todo el algoritmo varíe ampliamente en sus resultados.

Por otro lado consideramos que es necesario considerar a la función de fitness como un parámetro más dentro del algoritmo, ya que puede hacer que distintos métodos se comporten de forma distinta respecto al mismo problema.

Y finalmente, creemos que nuestros resultados podrían haber sido mejores, ya que sólo exploramos las mejores variantes que encontramos durante el desarrollo, pero dado la amplia cantidad de posibilidades que habían para ajustar la red, puede que no hayamos conseguido la mejor.

Método de reemplazo	Probab. de mutación	Método de selección 1	Método de selección 2	Crossover	k	k mixto	E.C.M serie no utilizada	E.C.M cjto de testeo	E.C.M cjto testeo
Reemplazo 2	0.001	Ruleta	Mixto	1point	20	10	0.019557	0.017298	0.018241
Reemplazo 2	0.003	Elite	Mixto	1point	20	10	0.031663	0.032657	0.022719
Reemplazo 2	0.003	Elite	Mixto 2	1point	20	10	0.027122	0.027984	0.027551

Tabla 1: Comparación de resultados

**Método de selección 1** refiere al primer método de selección utilizado. Análogamente, **Método de selección 2** refiere al segundo método de selección utilizado (si es que el método de reemplazo requiere utilizar 2 métodos de selección). La columna titulada con **k** refiere a la brecha generacional, es decir, cuántos individuos de la población son seleccionados para ser recombinados. La columna titulada con **k mixto** refiere a la cantidad de elementos seleccionados con el primer método de selección cuando el método de selección es mixto. **E.C.M. serie no utilizada** refiere al error cuadrático medio obtenido con una serie que no fue utilizada para evaluar a las redes neuronales obtenidas durante la ejecución del algoritmo.

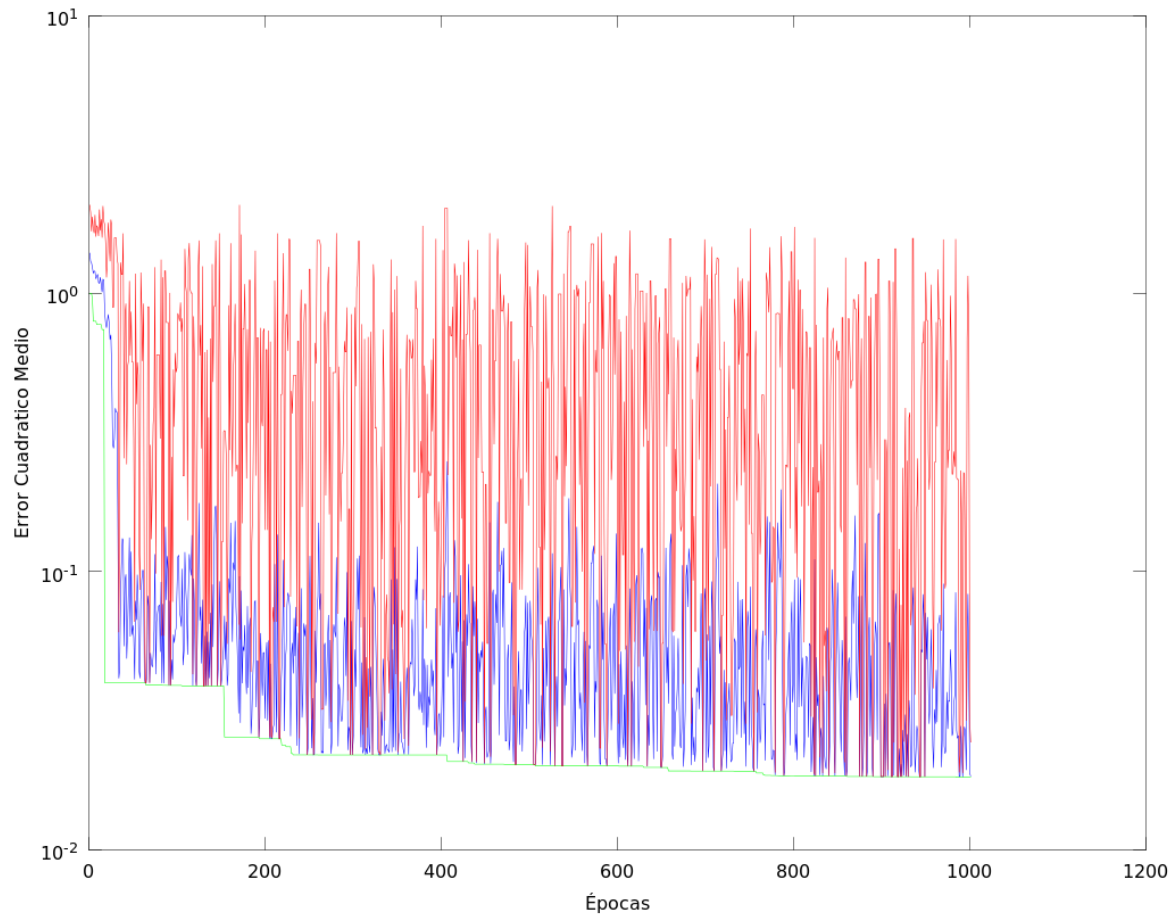


Figura 1: Error cuadrático medio

La línea inferior en la gráfica representa el error cuadrático medio mínimo de la población. La línea del medio representa el promedio de la población y la superior, el mayor error cometido.