

Evolúciós Algoritmusok (EA): Az evolúciós algoritmusok populáció alapú heurisztikus optimalizációs algoritmusok. A biológiai értelemben vett evolúció alapján működnek, egyszerűsített modellezései az elolúciónak. A mesterséges intelligencia témakörében az evolúciós algoritmusok négy fajtáját különböztetjük meg:

1. Genetikus Algoritmus
2. Genetikus Programozás
3. Evolúciós Programozás
4. Evolúciós Stratégia

Genetikus Algoritmusok: A genetikus algoritmusok optimalizációs eljárások, ahol a lehetséges hipotéziseket az egyedek a hipotézis tér valamely részét pedig a populációk reprezentálják. A genetikus algoritmusok minden egyes lépésben heurisztikák segítségével módosítják, vagy bővítik az aktuális populációt. Minden egyes lépésben képesek adni egy közelítést a megoldásra, amelynek a pontossága az idő teltével nő. Az egyedeken értelmezve van egy úgy nevezett fitness érték ($f(x) \rightarrow R$), amely a valós számok halmazába képez le. Az egyedeket reprezentáló vektor bináris értékeket tartalmaz, míg az evolúciós algoritmusoknál valós vektor.

Algoritmus:

- P_0 legyen egy kezdeti (random) populáció
- $t := 1$
- **repeat**
 - $P'_t := \text{NewPop}(P_t)$
 - $P_{t+1} := \text{PopGen}(P'_t, P_t)$
 - $t := t + 1$
- **until** (megállási feltétel)

Populáció kiválasztása:

1. generikus (teljes lecserélés)
2. elitizmus (legjobbak megmaradnak)
3. helybeni ($|P_i| = 1$)

Operátorok (utódok létrehozása):

1. szelekció

- párverseny: vegyük c_1 és c_2 egyedeket. Ha a $\frac{f(c_1)}{f(c_1)+f(c_2)}$ érték egy megfelelő véletlen értéknél nagyobb, akkor a c_1 egyed nyer.
- ranking: az összes egyed fitness értékét elhelyezzük egymás után egy intervallumon, majd vesszük az intervallum egy véletlen pontját. A pont alatt található fitness érték tulajdonosa lesz kiválasztva.

2. rekombináció

A rekombináció során a szülők valamilyen kombinációjából alakul ki az új egyed.

- egypontos
- kétpontos
- OX (ordered crossover)

3. mutáció: A mutáció során a szülő egyedet reprezentáló vektor értékeit módosítjuk valamekkora valószínűséggel.

Megállási feltételek: A következő megállási feltételeket lehet alkalmazni a genetikus algoritmusoknál:

- futási idő (ha költséges a kiértékelés)
- iteráció szám
- az optimum javulásának sebessége
- a populáció homogenitása

Genetikus algoritmusoknál alkalmazott fogalmak:

- genotípus (reprezentációs tér)
- fenotípus (megoldáskezdemény)
- gének (a reprezentáló vektor pozíciói)
- allélok (a pozíciók lehetséges értékei)

A genetikus algoritmusok nagy előnye, hogy a globális optimumhoz konvergálnak, általában egyszerűen kódolhatók, széles körben használhatók, valamint zajos fitness függvény esetén is jól működnek.

Hátránya viszont, hogy véges lépésben csak az optimum egy közelítése garantálható. Bizonyos esetekben lehézkés lehet az egyed reprezentáció. Nagy futásidő.

Szimulált Hűtés (Simulated Annealing): A szimulált hűtés algoritmus is életközi elveken alapul és globális optimum keresésre használatos, a genetikus algoritmusok egy speciális esete. A hőmérséklettől függő valószínűséggel az algoritmus nem akad el egy lokális optimumban, hanem tovább folytatódik. A megfelelő hűtési stratégia mellett az algoritmus véges lépésben megközelíti az optimum értékét.

Algoritmus:

- legyen $s \leftarrow$ kiindulási állapot
- legyen $T_0 \leftarrow$ kezdőhőmérséklet
- $k := 1$
- **repeat**
 - $r := N(s)$
 - **if** $f(r) < f(s)$ **then** $s := r$
 else if $e^{\frac{f(s)-f(r)}{T_k}} > rand(0, 1)$ **then** $s := r$
 - $T_{k+1} := H(t_k, k)$
 - $k := k+1$
- until** (megállási feltétel)

Az algoritmusban az $N(s)$ az s pont egy szomszédját jelöli, ahova az aktuális ciklusban lépni szeretnénk. Az $f(x)$ az optimalizálandó függvény x helyen felvett értékét adja meg. A $H(T_k, k)$ pedig a hűtési ütemterv, azaz az aktuális hűtési stratégia.

Egy lehetséges hűtési stratégia: $T_{k+1} := T_k * \alpha$, ahol $\alpha \in (0.5, 1)$ értéket vesz fel.

A megállási feltétel lehet többek között a hőmérséklet, a konvergencia sebessége, iteráció száma.