**Genetinis Algoritmas**

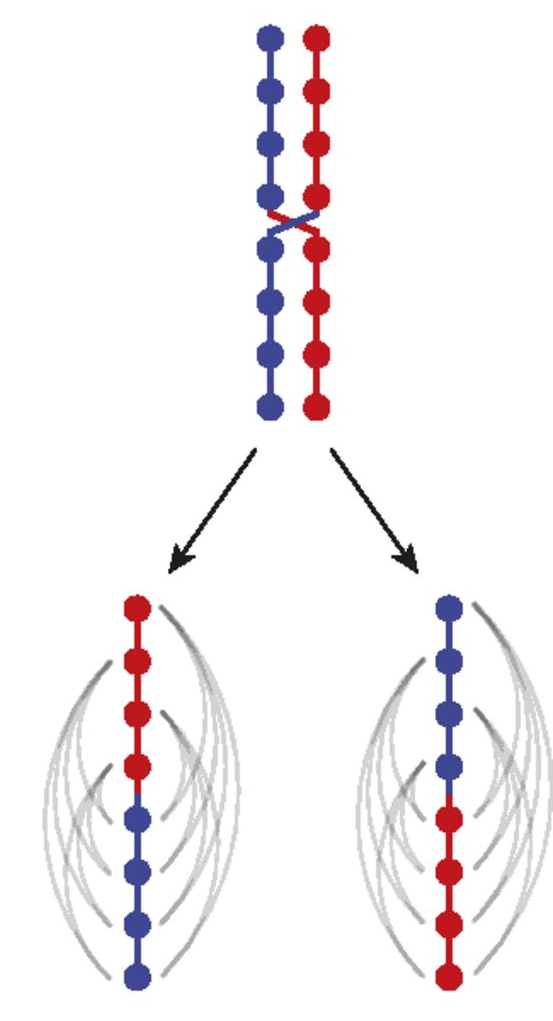
Genetinis algoritmas - algoritmas, paremtas gyvybės bruožų pokyčiais per kartas (Evoliucija). \*Genetinis algoritmas pasirėmia evoliucija trejais būdais. \*

* Gamtinė atranka - \*gyvybės, kurios yra geriausiai prisitaiko prie aplinkos išgyvena\*.
* Persikryžiavimas - genų mainai tarp tėvų gimdant naują palikuonį.
* Mutacija – palikuonio genetinės medžiagos pokytis.

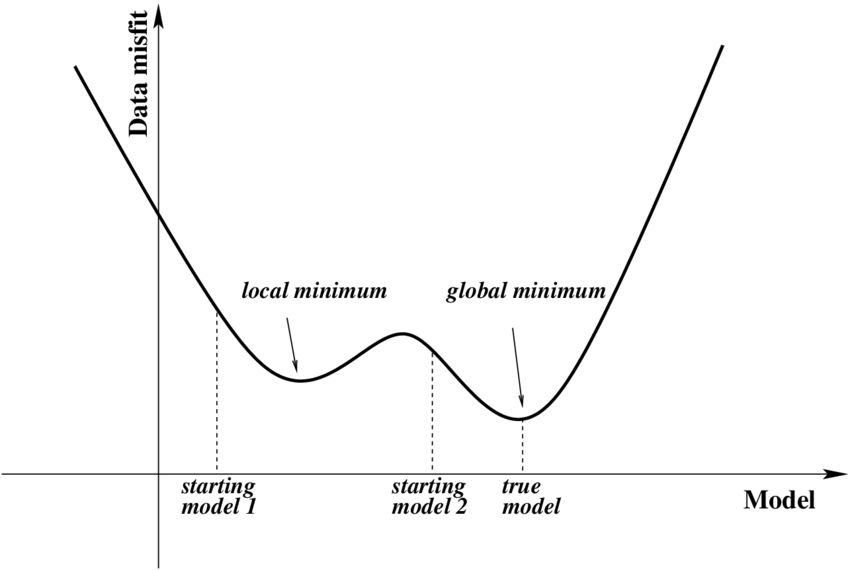
**Gamtinė atranka**

Simuliuoti gamtiniai atrankai programoje naudojamos skaitinės vertės vadinamos \*fitness\*. Kiekvienas objektas populiacijoj atsiranda su atsitiktinai išrinktais genais ir bando reaguoti į aplinką. Kuo arčiau objektas yra tikslo tuo labiau jo \*fitness\* didėja. Kiekvienam kartos objektui pasibandžius prisitaikyti prie aplinkos algoritmas juos surūšiuoja pagal jų \*fitness\*. Tie, kurie pasirodė geriausiai tarpusavyje palieka palikuonius persikryžiavimo būdu kitai kartai.

**Persikryžiavimas**

Persikryžiavimo metu objektas A ir objektas B sukuria naujų palikuonių. Tai daroma maišant jų abiejų genus gaunant 2 naujus palikuonius.

**Mutacija**

Naujų palikuonių genų atsitiktinis pakitimas. Mutacija yra paskirtis algoritme yra vengti \*local minimum\*. Be mutacijos vienintelės genų galimos vertės būtų tos, kurias gauna pirmoje kartoje tik sumaišytos naujose kartose. Mutacija leistų potencialiai išbandyti kiekvieną galimą vertę, radus labiau tinkančią vertę užduočiai objektas gebės surinkt daugiau \*fitness\* ir perduos savo genus kitom kartom. \*Global minimum\* - visos vertės yra optimalios užduočiai atlikti.

**Genetinio algoritmo sąsaja su neuroniniu tinklu**

Neuroniniam tinkle gyvybę sudarantys genai yra pačio tinklo sudarantys elementai: svoriai, aktyvacijos funkcija, sluoksniai, neuronu kiekis. Genetiniui algoritmui privaloma yra svoriai, kitu elementu modifikavimas yra pasirinktinas.

**Genetinio algoritmo pritaikymas mūsų programoje**

\*Fitness\* yra priklausomas nuo atstumo tarp raketos ir planetos kurią bando pasiekti, į ją atsitrenkus raketos \*fitness\* padidinamas 10 kartų, kad užtikrinti, kad raketa paliks palikuonių kitai kartai. Visom raketom pabandžius skristi naudojamas Šelo rikiavimo algoritmas, kad greit surūšiuot raketas pagal jų pasirodymą. Vėliau naudojamas persikryžiavimas, kur naudoja geriausiai pasirodžiusių 50% raketų ir sumaišo jų genus (svorius) tarpusavyje, kad gauti naujų palikuoniu kitai kartai. Kiekvienas naujai sukurto palikuonio svoris gali atsitiktinai mutuoti, tam, kad toliau ieškotų efektyvesnio būdo spręsti užduočiai.

**\*Įvesta informacija\***

Iš raketos yra brėžiamos 7 linijos kas 30°, kai linija atsitrenkia į kita objektą (planetą) tas atstumas tarp raketos ir atsitrenkto taško tampa neuroninio tinklo įvesta informacija.