

Genetiniai algoritmai I dalis

Parengė: lekt. Neringa Dubauskienė

Šioje paskaitoje sužinosite apie tai:

- Kas evoliucijos teorijoje svarbu genetiniams algoritmams
- Kas genetiniuose algoritmuose vadinama genais
- Kas yra chromosomos
- Kaip sudaromos populiacijos
- Kaip ir kodėl vykdoma atranka

Evoliucinių algoritmų sukūrimą įkvėpė evoliucijos teorija. Gamtoje matome daug labai skirtingų „geriausių sprendimų“ – įvairių organizmų, kurie klesti skirtingose aplinkose. Nepaisant organizmų įvairovės, jie visi egzistuoja evoliucijos dėka. Taip pat ir evoliuciniai algoritmai gali pasiūlyti įvairių sprendinių skirtingoms problemoms spręsti. Evoliuciniai algoritmai gali būti labai įvairūs, bet jie visi įkvėpimo semiasi iš gamtos, kur įvairūs organizmai reaguoja į aplinką ir stengiasi joje išgyventi ir klestėti. Evoliucinių algoritmų pagalba, pavyzdžiui, galima generuoti ir programinį kodą, įvairiai keičiant atskiras jo dalis, tai yra vykdant kodo mutacijas, kad galiausiai gautume optimalią programą norimam uždaviniui spręsti. Tai panašu į bandymą sukurti dirbtinę gyvybę. Tai atsispindi ir garsios simuliacijos pavadinime „The game of life“, arba „Gyvenimo žaidimas“. Tai paprastas žaidimas, simuliuojantis ląstelių gebėjimą išgyventi ir daugintis. Nepaisant savo paprastumo, šis žaidimas suteikia įžvalgų apie, pavyzdžiui, kelių robotų tarpusavio sąveiką ir panašaus tipo problemas.

Bene garsiausia evoliucinių algoritmų grupė yra vadinamieji genetiniai algoritmai, kurie remiasi pagrindiniais evoliucijos teorijos principais ir jų pagalba siekia rasti optimalų sprendimą.

Kiek supaprastinus, galima teigti, kad evoliucijos teorija remiasi keturiais pagrindiniais principais:

1. Atskiri kiekvienos rūšies individai skiriasi vienas nuo kito. Tos pačios rūšies gyvūnai ir augalai, nors ir panašūs vienas į kitą, didžiąja dalimi nėra visiškai identiški – skiriasi jų dydis, spalva, raštai ir kita.
2. Dalis šių variacijų yra perduodamos palikuonims. Vaikai paveldi savo tėvų savybes, ir kuo labiau tos savybės padeda išgyventi – tuo didesnė tikimybė, kad jos bus paveldėtos.
3. Kiekvienoje kartoje gimsta daugiau individų, nei gali išgyventi.
4. Perteklius reiškia „kovą už būvį“ – tie individai, kurie labiausiai prisitaikę gyventi toje aplinkoje, tai yra, tam tikra prasme geriausi, turi didžiausią tikimybę susilaukti palikuonių, o mažiau prisitaikę – tiesiog miršta. Jei palikuonių susilaukia gerai aplinkoje prisitaikę tėvai – tikėtina, kad tie palikuonys taip pat bus gerai prisitaikę. Tokiu būdu kiekviena nauja karta vis geriau prisitaiko prie esamos aplinkos.

Žinoma, tai labai supaprastintas evoliucijos teorijos aprašymas. Vis dar yra nemažai dalykų, kurių mes nesuprantame ir kurie gali tapti nauju įkvėpimo šaltiniu dar geresniems algoritmams. Tačiau net ir šis supaprastintas modelis padeda kurti naujus algoritmus. Vertinant genetinių algoritmų veikimą, gauname ne įvairius organizmus, bet daug tinkamų ar blogų uždavinio sprendimų. Genetinių algoritmų tikslas – atmesti nevykusius sprendimus ir rasti tokius, kurie geriausiai tinka kiekvienam uždaviniui.

Prieš pradėdami spręsti uždavinius genetinių algoritmų pagalba, pirmiausia turime tinkamai aprašyti turimą problemą. Įsivaizduokime, kad turime paskirstyti darbus didelės gamyklos darbuotojams – joje dirba kelios darbuotojų komandos, kurioms skirtingu metu reikia skirtingų įrenginių ir jie gali atlikti įvairias užduotis. Jei komanda per ilgai dirbs su ta pačia užduotimi – darbuotojams atsibos ir jie praras motyvaciją, o jei dalis įrengimų liks tušti, kol prie kitų darbuotojai lauks savo eilės – neefektyviai išnaudosime išteklius. Tokiu būdu radome keturis kintamuosius – darbuotojų komandos, turimi įrenginiai, atliekamos užduotys ir laikas, kada kuri komanda kurią užduotį atlieka. Kiekvienas iš šių kintamųjų turi būti aprašytas. Šiuo atveju, įkvėpimo vėl semiamės iš gamtos – įvairios skirtingų organizmų savybės gamtoje yra koduojamos genų sekomis. Taip pat ir genetiniuose algoritmuose – skirtingi kintamieji aprašomi genų sekomis. Biologiniai genai aprašomi raidžių sekomis ACTG... . Skaitiniuose sprendimuose populiariausias būdas kurti genų rinkinius – naudoti binarinį kodą, tai yra nuliukus ir vienetukus, nors galimi ir kiti variantai. Jei turime tris komandas, jas galime aprašyti tokiais genų rinkiniais:

001 – pirma darbuotojų komanda

010 – antra darbuotojų komanda

100 – trečia darbuotojų komanda

Kiekvienas iš nuliukų ir vienetukų rinkinyje bus vadinamas genu. Taip pat aprašysime ir kitus kintamuosius ir galiausiai turėsime pradinį savo turimos problemos aprašymą. Kiti galimi aprašymo variantai – naudoti skaitmenis nuo 0 iki 9 ar bet kurias abėcėlės raides. Šis pasirinkimas priklausys nuo sprendžiamos problemos tipo.

Taigi – dirbtinio intelekto srityje kuriant genetinius algoritmus genais vadinsime informacinius vienetus, kurie naudojami kintamiesiems koduoti. Visas kintamasis gali būti užkoduotas bet kokio ilgio genų rinkiniu. Binarinis kodavimas yra dažniausiai pasitaikantis, bet ne vienintelis būdas koduoti kintamuosius genetiniuose algoritmuose.

Turėdami genų rinkinius, galime sudarinėti chromosomas. Chromosoma genetiniuose algoritmuose vadinami galimi sprendimai. Mums reikalingas sprendimas turi apimti visus keturis kintamuosius – komandas, užduotis, įrenginius ir laiką, taigi šiuos kintamuosius aprašančius genus sudedame į vieną chromosomą.

Taip gauname pradinę populiaciją – keletą galimų sprendimų, kurie greičiausiai neatitinka jokių geriams sprendimams keliamų reikalavimų. Genetinio algoritmo tikslas – iš atsitiktinai sudarytų pradinių sprendimų sudaryti naujus, labiau tinkančius sprendimus. Tai panašu į būrį įvairių atsitiktinių gyvūnų rūšių, kurios atsidūrė naujoje aplinkoje ir galiausiai išliks tik tos rūšys, kurios prisitaikys prie naujos aplinkos.

Sudaryti naujas populiacijas – tik pusė darbo. Jei tik sudarytume naujus sprendinius, bet neatmestume nei vieno jų, mūsų algoritmas užimtų visą turimą kompiuterio atmintį ir nepajėgtų atlikti tolimesnių skaičiavimų. Senuosius sprendinius, kurie neatitinka iškeltų sąlygų reikia atmesti, užtikrinant, kad nebus sunaikinti galimai tinkami sprendimai. Sprendimai, kurie panašiausi į mums tinkamus – pavyzdžiui, tokie, kurie paskiria mažiausią skaičių komandų dirbti prie to paties įrenginio, paliekami, o likę sunaikinami, ištrinami iš atminties. Galima palikti tik pačius geriausius individus, bet tai rizikinga – tokiu būdu gausime mažą sprendimų įvairovę ir galime užstrigti ties vidutiniškais sprendimais, niekad taip ir nerandant pačių geriausių. Net ir labai prastuose sprendimuose gali slypėti nedidelė dalis genų rinkinio, kuri bus svarbi geriausiam sprendimui rasti. Tai panašu į veislinius šunis – jei veisiami tik griežtai veislės standartus atitinkantys šunys, galiausiai genetinė įvairovė tampa tokia maža, kad gimstantys šuniukai susiduria su įvairiomis sveikatos problemomis. Tas pats nutinka ir su sprendimais, todėl keletas ne pačių geriausių sprendimų paliekama, tokiu būdu išlaikant įvairovę ir paspartinant geriausio sprendimo radimą. Patys tinkamiausi sprendimai vadinami elitiniais, o jų pagrindu sudaryti sprendimai – elitiniais vaikais. Atrankos metu svarbu nustatyti balansą tarp elitinių ir likusių sprendimų. Vienas iš būdų tą padaryti – skaičiavimų pradžioje siekti didesnės įvairovės ir palikti daugiau ne elitinių sprendimų, o algoritmui artėjant prie pabaigos likusius skaičiavimus grįsti tik elitiniais sprendimais. Tokiu būdu subalansuojamas algoritmo paieškos laikas išvengiant pavojaus, kad bus apsistota ties vidutiniškais sprendimais.