

Projekat iz predmeta: „Mašinsko učenje“

Kratko uputstvo:

Kod se sastoji od dva modula: **NeuralNetwork.py** i **main.py**. Prvi modul (NeuralNetwork.py) sadrži implementaciju klase NeuralNetwork, dok main.py služi za testiranje programa, učitavanja dataseta i sl.

Projekat je pisan u Pythonu, te za njegovo pokretanje i korištenje je, osim samog Pythona, još jedino potrebno instalirati biblioteku numpy (ako već niste). Pokretati možete pomoću bilo kojeg Python IDE-a ili kroz komandnu liniju.

U modulu main.py već postoje tri dummy seta podataka sa odgovarajućim neuralnim mrežama za njihovo učenje. Potrebno je i dovoljno ukloniti komentare nad tim datasetom i pokrenuti program. Za testiranje svojih datasetova, potrebno je unijeti dataset (s tim da je potrebno da i input i output budu u matričnom zapisu, inače može doći do pada programa zbog problema sa dimenzijama¹). **Nakon što ste učitali svoj dataset, potrebno je da instancirate objekat NeuralNetwork kao NN.NeuralNetwork(), koji prima tri parametra:** broj neurona input layera, broj neurona hidden layera, broj neurona output layera, te learning rate.

Nakon što smo instancirali objekat NeuralNetwork, sve što je potrebno uraditi je „istrenirati“ našu neuralnu mrežu nad tim datasetom, što radite pozivajući metodu **NeuralNetwork.train()**, koja **prima 4 parametra: Input dataset, output dataset, broj iteracija, bool vrijednost true ili false.** Broj iteracija predstavlja broj prolaska kroz dataset, odnosno efektivnu duljinu treninga. Sa druge strane, bool vrijednost omogućava da Neuralna mreža, prilikom treninga, ispisuje i greške, nakon svake iteracije. Default-ni izbor je false (dakle nema ispisa greške).

Sve specifičnosti koda su posebno zakomentarisane. Projekat je implementiran na što nižem nivou, tj. Koristeći ništa drugo do najosnovnijih jednostrukih i dvostrukih for petlji. Razlog ovakve implementacije je taj što se zbog proizvoljnosti broja neurona u svakom layer-u, kroz matrični zapis, teško može shvatiti što program radi. Iz tog razloga, smatram da ovakav način implementacije puno bolje pokazuje „step-by-step“ pristup.

Reference radi, uz svoj projekat prilažem i dio koda koji ima istu funkcionalnost kao i osnovni kod, s tim da koristi matrice umjesto nizova, a sam račun je implementirat pomoću matričnih operacija. Taj kod je u potpunosti funkcionalan, osim djela gdje update-amo težinske koeficijente. Dio kod update-anja težinskih koeficijenata nije implementiran zbog problema sa dimenzijama mogućih layer-a (o čemu je bilo govora na početku). Ipak, prilažemo i taj dio koda, kako bi pokazali najbrži način implementacije potpuno funkcionalne neuralne mreže u svega 50-tak linija koda.

S poštovanjem,

Tarik Kreso

¹ Nije implementirana provjera dimenzija, s obzirom da radim sa proizvoljnim brojem neurona, pa bi eventualno pokrivanje svih „loših“ slučajeva uzelo daleko više vremena nego sam projekat, što svakako nije fokus.