Drugi domaći zadatak iz Tehnika i metoda analize podataka

Danilo Veljović, broj indeksa 1120

Opis skupa podataka

Funkcijom load_breast_cancer() se iz biblioteke sklearn učitava Winsconsin breast cancer dataset. Atributi ovog dataseta su dobijeni obradom digitalizovanih slika snimljenih dojki žena. Opisuju karakteristike jezgara ćelija koje su prisutne na slici. Link za preuzimanje dataseta je: https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/ . Atributi dataseta su:

1. ID
2. Dijagnoza (maligni/benigni)
3. - 32. Za tri vrste ćelija imamo po deset atributa za svaku od njih. Tih deset atributa opisuju stanje ćelije i njenog jezgra i oni su:
   a) poluprečnik jezgra
   b) tekstura
   c) obim jezgra
   d) površina jezgra
   e) glatkost
   f) kompaktnost
   g) konkavnost
   h) konkavne tačke
   i) simetrija
   j) fraktalna dimenzija (razuđenost jezgra)

Na osnovu ovog dataseta potrebno je predvideti da li je neka od ćelija raka maligna ili benigna. Dataset predstavlja problem koji se može rešiti binarnom klasifikacijom.

Opis metode

Mera za entropiju uzorka koja se ovde koristi je data formulom na slici 1. Ako je uzorak homogen, entropija će biti 0, a ako je uzorak 50 – 50, entropija je 1. Ova mera je implementirana u funkciji **entropy(y)**.

Metoda stabla odluke je ovde implementirana kroz dve klase, klasa Node i klasa DecisionTree. Klasa Node predstavlja čvor stabla odluke. Ako je u pitanju čvor koji nije list, tu se čuva atribut po kojem se najbolje vrši grananje, i vrednost tog atributa i čuvamo vezu ka levom i desnom elementu stabla. Ako je u pitanju list onda čuvamo najčešću klasu elementa tog čvora. Ova klasa takođe ima funkciju koja pokazuje da li je taj čvor list.

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Slika 1 – Formula za izračunavanje entropije uzorka

Klasa DecisionTree ima atribut zaustavljanja, označen kao *min_samples_split (ovo označava minimalni broj uzoraka koji mora da postoji u čvoru da bi se dalje delio i po defaultu je ovde postavljen na 2*, maksimalna dubina stabla koja se čuva u atributu max_depth i po defaultu je 100. Atribut n_features se ne koristi u ovom primeru, ali inače se može koristiti da se specificira broj atributa nad kojima će se formirati stablo, umesto nad svima (dobijeni podniz se formira random).

Funkcija koja formira stablo je nazvana po ugledu na sve ostale funkcije iz sklearn biblioteke, i zove se **fit()**. U toj funkciji se proverava da li će se stablo formirati po nekom podskupu atributa, ili nad svim atributima i poziva se pomoćna funkcija **grow_tree()** koja će formirati stablo. Stablo će se ovde formirati rekurzivno. Tako da se ova funkcija piše kao bilo koja druga rekurzivna funkcija. Prvo se piše uslov izlaska iz rekurzije koji ovde može da bude da je dostignuta maksimalna dubina, ili ako je n_labels = 1, tj ako u čvoru imamo samo jednu klasu ili ako je broj uzoraka u čvoru manji od zadatog minimalnog broja, onda se nalazimo u listu stabla.

Ako smo stigli do lista stabla, onda upisujemo kreiramo list i u njega upisujemo najčešću vrednost klase. Ako nismo u listu stabla, nego u nekom od čvorova, nalazimo atribut koji je najpogodniji za podelu uzorka u tom čvoru funkcijom best_criteria(). Nakon toga se stablo deli na levo i desno podstablo, tj za elemente koji pripadaju levom i desnom podstablu zove se funkcija grow_tree() i formiraju se elementi levog i desnog podstabla.

Funkcija best_thresh() vraća atribut koji je najbolji za podelu uzorka tog čvora, i vraća optimalnu vrednost koju taj atribut treba da ima. To radi tako što za svaki od atributa koji potencijalno služi da podeli uzorak u čvoru, nalazimo jedinstvene vrednosti i za svaku od tih vrednosti računamo information gain. Za onaj atribut čija neka vrednost daje najveći information gain, njegova će ta vrednost i ime, biti atribut podele i threshold granica za taj čvor.

Kada se pokušava da se klasifikuje instanca koja do tada nije viđena u treniranju, stablo se obilazi sve dok se ne dođe do nekog od listova stabla čime se instanca klasifikuje. Ovaj kod je jednostavan i dat je kroz funkcije predict() koja u sebi poziva traverse_tree() koja rekurzivno obilazi stablo.

U kodu je takođe ručno implementirana funkcija koja radi cross validaciju nad datasetom, i za tačnost se u 3 navrata dobijaju vrednosti od 90.78%, 96.05% i 93.33%.
Dobijena matrica konfuzije pokazuje da klasifikator prepoznaje 157 od ukupno 167 negativnih instanci, odnosno 269 od ukupno 288, što su dobri rezultati. Preciznost je izmerena na 96.4%, a odziv na 93.4%, a f1 mera 94.8%. Površina ispod ROC krive koja jako dobro aproksimira performanse modela je 0.937 što je jako dobar rezultat.