## Stablo odluke

### Način implementacije algoritma

1. Korak - Na samom početku definisane su klase Node i DecisionTree.

Klasa Node ima sledeće atribute:

feature\_index – ovaj atribut klase određuje na osnovu kog atributa iz skupa podataka se vrši podela na ovom čvoru

threshold – određuje granicu koja predstavlja vrednost na osnovu koje se u čvoru vrši podela na levu i desnu granu

left – čvor koji je levi potomak trenutnog čvora

right – čvor koji je desni potomak trenutnog čvora

info\_gain – redukcija entropije nakon što se skup podataka podeli na ovom čvoru

value - ako je čvor list, određuje klasnu pripadnost

Klasa DecisionTree ima sledeće atribute:

min\_split\_samples – minimalni broj uzoraka da bi se vršila dalja podela.

max\_tree\_depth – maksimalna dubina stabla

tree\_root – pokazivač na početni čvor

Unutar ove klase su implementirane sve funkcije potrebne za izgradnju stabla odluke.

1. Korak – Odabir metrike na osnovu koje se vrši podela.

Za svoju implementaciju algoritma stabla odluke sam odlučila da koristim entropiju kao meru. U kontekstu stabla odlučivanja, entropija se koristi za kvantifikovanje nesigurnosti u podacima. Ako su svi podaci iz iste klase, entropija je nula jer nema nesigurnosti.



1 Formula za računanje entropije

* Funkcija child\_dataset\_entropy(self, y) funkcija koja računa entropiju za jedan skup podataka koji se prosledi kao parametar y.
* Funkcija threshold\_entropy(self, x, y, feature\_idx, threshold) jeste funkcija koja računa entropy za jedan feature, za jedan threshold. U ovoj funkciji se skup podataka najpre podeli na levi i desni skup podataka na osnovu feature-a i threshold-a koji su prosleđeni kao parametri, a onda se izračuna konačna entropija za jednu takvu podelu.
* Funkcija feature\_entropy(self, x, y, feature\_idx) računa entropy za jedan feature, za sve threshold-ove. Ova funkcija za prosleđeni feature pronalazi sve vrednosti u skupu podataka i za svaku vrednost posmatra kao threshold određuje entropiju za svaku moguću podelu pozivanjem funkcije threshold\_entropy(self, x, y, feature\_idx, threshold), i na osnovu dobijene vrednosti određuje sa kojim threshold-om se najviše smanjuje entropija u skupu podataka.
* Funkcija calculate\_features\_entropies(self, x, y) za sve feature u skupu podataka određuje najbolji threshold na osnovu kog se može izvršiti podela. I kao povratnu vrednost vraća dictionary u kome se za svaki feature nalaze vrednosti za threshold i entropiju.

1. Korak – Pronalaženje najbolje podele

Pronalaženje najbolje podele se vrši u funkciji find\_best\_split(self, x, y). Na osnovu mere odabrane u prethodnom koraku, pronalazi se najbolji feature i threshold na osnovu kog će se podeliti podaci. Najbolji threshold je onaj koji najviše smanjuje entropiju. Ova funkcija poziva funkciju calculate\_features\_entropies koja računa entropiju i threshold za svaki feature i na osnovu povratne vrednosti bira onaj feature na osnovu čije podele se najviše smanjuje entropija.

1. Korak – Izgradnja stabla odluke

Izgradnja stabla odluke se vrši u funkciji building\_decision\_tree(self, x, y, depth=0). Koristi se rekurzija za izgradnju stabla odluke. Počinje se sa celim training set-om i pronalazi se najbolja karakteristika i prag za podelu. Ovo stvara dva nova skupa podataka, a proces se ponavlja nad svakim od njih. Pronalazi se najbolja karakteristika i prag za podelu stvarajući tako više skupova podataka i više čvorova. Rekurzija se zaustavlja kada je ispunjen određeni uslov:

* kada sadrži samo čvorove iste klase
* kada skup podataka za podelu sadrži manje uzoraka od postavljene minimalne vrednosti
* kada je dostignuta maksimalna dubina stabla

1. Korak – Predviđanje

Predviđanje se vrši funkcijom predict(self, node, x). Obilazi se drvo sa novim uzorkom počevši od korena. Na svakom čvoru se proverava feature na osnovu koga se vrši podela. Ako je vrednost feature-a manja od praga ide se na levo dete, a u drugom slučaju se ide na desno dete. Vrednost lista je predikcija za novi uzorak.

### Skup podataka

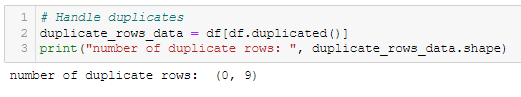
Ovaj skup podataka je izvorno iz Nacionalnog instituta za dijabetes i bolesti probave i bubrega. Cilj je da se na osnovu dijagnostičkih merenja predvidi da li pacijent ima dijabetes. Svi pacijenti ovde su žene koje imaju najmanje 21 godinu indijanskog porekla Pima. Značenje atributa je sledeće:

* Pregnancies: broj trudnoća
* Glucose: koncentracija glukoze u krvnoj plazmi
* BloodPressure: dijastolni krvni pritisak (mm Hg)
* SkinThickness: debljina kožnog nabora tricepsa (mm)
* Insulin: 2-satni serumski insulin (mU/ml)
* BMI: Indeks telesne mase (težina u kg/(visina u m)^2)
* DiabetesPedigreeFunction: funkcija pedigrea dijabetesa
* Age: starost (godine)
* Outcome: promenljiva klase (0 ili 1)

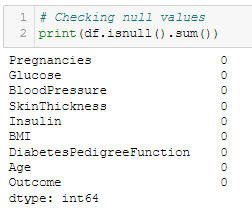
Broj instanci je 769.



Na osnovu pretprocesiranja je utvrđeno da u ovom skupu podataka nema duplikata.



Takođe, preprocesiranjem je utvrđeno da u ovom skupu podataka nema nedostajućih vrednosti.



### Zaključak

Preciznost modela je 0.69 (Figure 2), što znači da uspešno predviđeno oko 69% svih podataka u test skupu podataka.

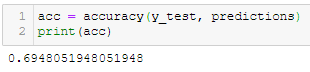


Figure 2

Model ima preciznost 0.79 za klasu 0, što znači da su za slučajeve gde je model predvideo da nije u pitanju dijabetes, 79% zaista nije bio dijabetes. Recall je 0.75 za klasu 0 što znači da je tačno identifikovano 72% među svim nedijabetskim slučajevima (Figure 3).

Model ima preciznost 0.56 za klasu 1, što znači da su za slučajeve gde je model predvideo da je u pitanju dijabetes, 79% zaista nije bio dijabetes. Recall je 0.65 za klasu 1 što znači da je tačno identifikovano 72% među svim dijabetskim slučajevima (Figure 3).

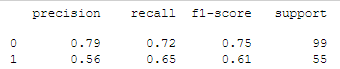


Figure 3

