Univerzitet u Beogradu, Fakultet organizacionih nauka - Mašinsko učenje

Caravana – Sell No Evil – Predviđanje kupovine automobila

Danilo Popović 100/18, Filip Jevtović 381/18

**1 Opis i razumevanje problema**

Kupovina automobila bi trebalo da bude jednostavan proces, međutim u stvarnosti to nije tako jednostavno.Auto spolja može da izgleda odlično dok “iznutra” može biti u znatno lošijem stanju.Neki od problema kupovine automobila sa aukcijskih auto placeva jesu promena pređene kilometraže, na oko nevidljivi mehanički problemi, loše slike automobila, nedovoljan opis stanja automobila...Kupovinom automobila sa nekim od gore navedenih problema može auto dilere da dovede do eventualnih gubitaka, jer zbog troškova transporta, zamene delova, fluktuacija cena na tržištu taj auto možda ne bude isplativ.Zbog toga glavni cilj jeste da pokušamo da predvidimo da li će automobil biti dobra kupovina, odnosno da auto neće imati neki skriveni problem.

**2 Opis i priprema podataka**

Set podataka se sastoji od 6798 observacija i 34 atributa.Izlazni atribut IsBadBuy se sastoji od nula koje znače da auto nije bio loša kupovina, dok jedinice znače suprotno.Proverom odnosa izlazne klase vidimo da 87,23% kupovina bilo dobro i to nam ukazuje na veliki disbalans u izlaznoj klasi.U setu ne postoje kolone AcquisitionType, KickDate koje su navedene u dokumentu sa opisom podataka.

**2.1 Nedostajuće vrednosti**

Uvidom u podatke vidimo da postoji veliki broj

nedostajućih vrednosti.Atributi PRIMEUNIT, AUCGUART i Auction sadrže veliki broj nedostajućih vrednosti pa nisu korišćene pri daljoj analizi.Nedostajuće vrednosti kao i kategorije sa manje od 20 ponavljanja kod promenljive Trim smo zamenili kategorijom Other. Za atribut WheelType NA vrednosti smo popunili tako sto smo tamo gde je IsBadBuy = 1 upisali vrednost 'Alloy',a tamo gde je IsBadBuy = 0, upisali vrednost 'Covers'. Kako za isti automobil nedostaju vrednosti promenljivih Nationality, Size i TopThreeAmericanName, uklonili smo taj red. Kako za 20 istih automobila nedostaju vrednosti promenljivih: MMRCurrentAuctionAveragePrice, MMRCurrentAuctionCleanPrice, MMRCurrentRetailAveragePrice i MMRCurrentRetailCleanPrice – uklonili smo tih 20 redova.

**2.2 Selekcija atributa**

Atribut VNST je suvišan jer imamo kolonu VNZIP1 na osnovu koje možemo za odredimo VNST, isto važi i za atribut WheelType i WheelTypeID.Atribut PurchDate predstavlja datum kupovine,a kako godinu kupovine možemo da dobijemo iz drugih atributa nismo koristili taj atribut.Atribut BYRNO ne utiče na predviđanje pa ga nismo koristili u daljoj analizi. Kako je atribut VehYear gotovo savršeno korelisana sa atributom VehicleAge, izbacićemo atribut VehYear iz dalje analize. Kako su svi "MMR" atributi visoko korelisani sa prvim "MMR", ostavićemo samo prvi u daljoj analizi.Primenom hi kvadrat testa izabrali smo 20 atributa koji naviše doprinose tačnosti predviđanja izlazne promenljive pa smo dalja predviđanja vršili na osnovu njih.

**3 Kreiranje prediktivnih modela sa originalnim skupom podataka**

Nakon pripreme podataka primenili smo neke osnovne algoritme mašinskog učenja koji će nam služiti kao osnova za dalja predviđanja i poboljšanja.Kao meru evaluacije modela korišćene su mere accuracy i auc.

**3.1 Logistička regresija**

Prvo smo testirali model logističke regresije iz paketa Scikit.Primenom desetostruke kros-validacije na trening skupu dobijamo preciznnost koja je ista kao raspodela izlazne klase, daljim ispitivanjem smo zaključili da ovaj model sve obzervacije klasifikuje kao dobre kupovine.

**3.2 Stablo odlučivanja**

Sledeći model koji smo testirali je stablo odlučivanja iz paketa Scikit.U poređenju sa logističkom regresijom dobijamo lošiju tačnost ali za razliku od logističke regresije model ne klasifikuje sve obzervacije kao dobre što objašnjava lošije rezultate.

**4 Optimizacija hiperparametara**

**4.1 Logistička regresija**

Optimizacija hiperparametra C nije dovela do poboljšanja modela zato što model još uvek klasifikuje sve obzervacije kao dobru kupovinu.

**4.2 Stablo odlučivanja**

Optimizacijom hiperparametara max\_depth, min\_samples\_leaf smo kao rezultat dobili da se naveća tačnost modela dobija za podrazumevane vrednosti parametra.

**5 Balansiranje podataka**

Kako smo u prethodnim modelima dobili loše rezultate odlučili smo da probamo da izbalansiramo podatke u odnosu na izlaznu klasu.Za rešavanje problema disbalansa klasa koristili smo tehnike uzorkovanja *oversampling* i *undersampling*.Poredili smo rezultate prethodnih modela na podacima dobijenim iz gore navedenih tehnika i došli do zaključka da tehnika *oversample* daje znatno bolje rezultate od *undersample* i od nebalansiranih podataka pa smo te podatke koristili u daljoj analizi.

**6 Ansambli**

Standardni modeli za predviđanje kao i njihova optimizacija nisu dali dobre rezultate, pa smo odlučili da probamo napredne modele predviđanja kao što su ansambli kako bi dobili bolje rezultate.Koristili smo RandomForest, AdaBoost i Bagging modele iz paketa Sklearn ensemble.Ove modele smo evaluirali na trening skupu primenom kors-validacije.

**6.1 Slučajna šuma**

Prvi model iz ensambl paketa koji smo testirali je slučajna šuma.Testirali smo model na *oversample* skupu podataka sa standardnim vrednostima i dobili jako dobre rezultate za accuracy i za auc, dok se za undersample i standardne podatke mere evaluacije bolje od svih standardnih modela koje smo probali.Optimizacijom hiperparametra min\_samples\_leaf, min\_samples\_split uz pomoć GridSearch došli smo do zaključka da su optimalne vrednosti ovih hiperparametra ustvari standardne vrednosti.Postavljanjem n\_estimators na vrednost 100 dobijamo izuzetno dobre rezultate gde nam je accuracy čak 0.974 a auc 0.999.

**6.2 Adaboost sa slučajnom šumom**

Kako smo za slučajnu šumu dobili do sada najbolje rezultate, odlučili smo da kao estimator modela Adaboost iskoristimo upravo slučajnu šumu.Adaboost sa standardnim parametrima daje veoma visok accuracy 0.974 koji je identičan slučajnoj šumi.Isprobavanjem različiih kombinacija hiperparametara zaključili smo da n\_estimators postavljamo na 10, a learning\_rate na 0.2,što je zapravo manje od standardnih vrednosti.

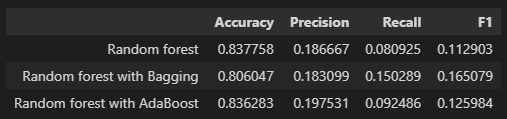
**6.3 Bagging sa slučajnom šumom**

Isto kao i za prethodni model kao estimator smo uzeli slučajnu šumu.Model sa standardnim hiperparametrima daje veoma dobre rezultate,za accuracy 0.949 a za auc 0.998.Optimizacijom hiperparametara nismo uspeli da dobijemo bolje rezultate.

**7 Poređenje rezultata**

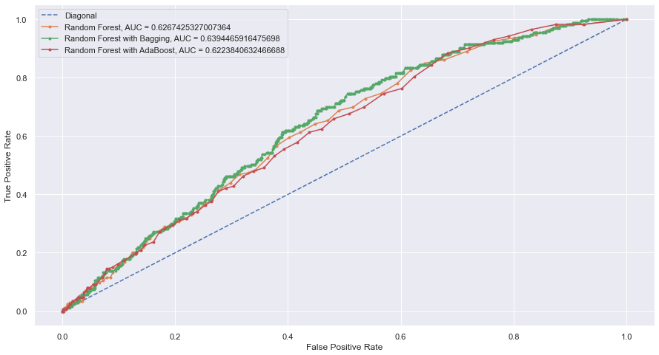
Iz prethodnih poglavlja može se zaljučiti da smo najbolje rezultate na treningu dobili za anasmbl algoritme.U nastavku ćemo uporediti rezultate dobijene na test skupu podataka i konačno izabrati najbolji model.

**7.1 Uporedni prikaz evaluacionih metrika**

****

U tabeli iznad se nalaze sve evaluacione metrike za 3 najbolja modela primenjena na test skupu podataka.Kao ciljanu klasu smo izabrali kada je IsBadBuy = Yes.Kao što možemo videti slučajna šuma ima za nijansu veću preciznost od Adaboost, a malo veću od Bagging modela.Bagging model ima malo veću vrednost F1 statistike, pre svega zbog veće recall mere.

**7.1 Roc kriva**

****

Kada posmatramo auc vrednosti na test skupu dolazimo da zaključka da su male razlike u istim, najveću vrednost ima Bagging model 0.6394, a najgoru Adaboost 0.6222.

**8 Zaključak**

Ako sagledamo evaluacione metrike za accuracy i auc kao najbolji modeli nameću se slučajna šuma i bagging.Slučajna šuma ima bolji accuracy dok bagging ima bolji auc.Kao konačni model izabrali smo slučajnu šumu, najviše zbog skoro 4% veće preciznosti.Najveći doprinos u kreiranju ovih modela je definitivno balansiranje podataka.*Oversampling* je uspeo da poboljša sve modele koje smo testirali, pa čak i one standardne. Selekcija k naboljih atributa nije imala velikog uticaja, pre svega jer anambl algoritam slučajna šuma već sadrži embedded metode izbora atributa. Važno je i napomenuti da je i izbacivanje visoko korelisanih numeričkih promenljivih takođe dosta uticalo na rezultate ovih modela.