Pregled gradiva

1. **Koje vrste mašinskog učenja postoje?**

   - Nadgledano učenje
   - Nenadgledano učenje
   - Učenje potkrepljivanjem (primer je autonomna vožnja automobila, ili igranje šaha)

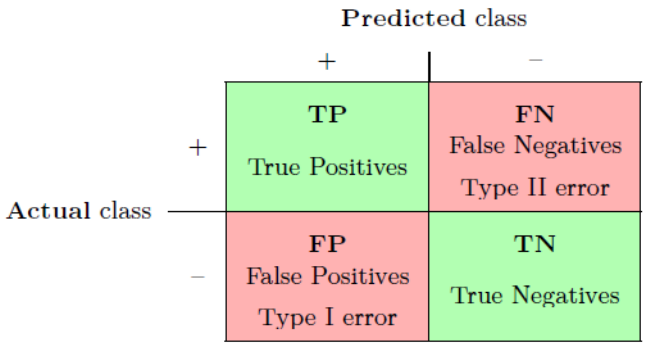
2. **Koji su koraci u izgradnji modela?**

   Najpre se odlučujemo koju ćemo metodu mašinskog učenja da koristimo. To radimo na osnovu toga da li je u pitanju problem regresije ili klasifikacije, kao i na osnovu toga koje veličine je dataset. Naravno, možemo da izaberemo više različitih metoda i da napravimo više modela, pa da ih uporedimo na test skupu i izaberemo najbolji. Prvi korak u izgradnji modela je izbor atributa koje ćemo koristiti. Ako imamo sirove podatke, možda će biti neophodno da konstruišemo atribute od sirovih podataka (neuronska mreža sama radi taj korak). Takodje, neophodno je da kategoričke atribute transformišemo u numeričke (ovaj korak je nepotreban kod modela zasnovanih na stablima). Zatim se najčešće radi standardizacija podataka i nakon toga su podaci spremni za upotrebnu i prelazi se na treniranje modela.

3. **U kom odnosu se dele podaci na skupove za trening, validaciju i testiranje? I čemu služi svaki od skupova?**

   Ako je ukupan broj podataka mali, onda ih najčešće delimo u odnosu 70:15:15. A ako imamo veliki skup podatka (npr. preko 15000 instanci), onda je dovoljno smestiti po otprilike 2000 instanci u skupove za validaciju i testiranje, jer nam ti skupovi služe samo za ispitivanje precizosti modela, a već sa 2000 instanci ćemo dobiti verodostojan rezultat. Trening skup služi za pravljenje modela, odnosno za dobijanje optimalnih parametara modela (za fiksirane hiperparametre), validacioni skup služi za dobijanje optimalnih hiperparametara modela, a test skup za konačnu evaluaciju modela.

4. **Kako izgleda matrica konfuzije?**



5. **Koje mere preciznosti koristimo u slučaju klasifikacije?**

   accuracy $= \frac{TP+TN}{TP+TN+FP+FN}$

   precision $= \frac{TP}{TP+FP}$

   sensitivity (true positive rate) $= \frac{TP}{TP+FN}$

   specificity (true negative rate) $= \frac{TN}{TN+FP}$

   F_1 score $= \frac{2 \cdot precision \cdot sensitivity}{precision+sensitivity}$

   AUC (površina ispod ROC krive).

6. **Koje mere preciznosti koristimo ako imamo nebalansirane podatke?**

   F_1 score i AUC.

7. **Koja je razlika izmedju regresije i klasifikacije?**

   Klasifikacija se koristi kada je zavisna promenljiva kategoričkog tipa, na primer kada hoćemo da odredimo da li je na slici mačka ili pas, a regresija se koristi kada je zavisna promenljiva neprekidnog tipa, npr. kada hoćemo da predvidimo temperaturu vazduha ili broj prodatih proizvoda.

8. **Šta je standardizacija podataka i čemu služi?**

   Standardizacija podataka je proces koji se sastoji u sledećim koracima. Posmatramo neki fiksirani prediktor, odnosno neku odabranu kolonu u podacima. Prvo računamo uzoračku sredinu i standardnu devijaciju te kolone na trening skupu. Zatim od vrednosti u toj koloni na trening, validacionom i test skupu oduzimamo izračunatu uzoračku sredinu i delimo sa izračunatom uzoračkom standardnom devijacijom. Na taj način dobijamo da su vrednosti tog prediktora raspodeljene oko nule, sa disperzijom 1. Ovaj proces je neophodan kada nam je bitno da svi prediktori budu na istoj skali. A to nam je bitno npr. ako koristimo euklidsku metriku ili ako koristimo *lasso* ili *ridge* regularizaciju. Standardizacija svakako ne može da odmogne, pa je poželjno uraditi je. Npr. ispostavilo se da optimizacija parametara neuronske mreže traje kraće ako su podaci standardizovani.

9. **Šta je overfitting, kako ga prepoznati i kako ga izbeći?**

   Overfitting je pojava kada se model previše prilagodi podacima u trening skupu. Prepoznajemo ga tako što izračunamo preciznost modela i na trening i na test skupu. Ako je preciznost na trening skupu značajno veća nego na test skupu, to je znak da je došlo do overfittinga. Primer je Lagranžov interpolacioni polinom koji prolazi kroz sve tačke trening skupa, tj. greška na trening skupu je nula, ali će greška na test skupu biti velika. Postoji više načina za borbu sa overfittingom:

   - Regularizacija, koja služi da ne damo pojedinim parametrima da budu previše veliki.
   - Pojednostavljivanje modela, odnosno smanjivanje broja parametara.

10. **Kada najčešće koristimo neuronske mreže?**

    Neuronske mreže se koriste kada imamo dovoljno veliki skup podataka. Ako imamo mali skup podataka, onda druge metode postižu bolje rezultate. Takodje, ako su podaci sirovi, tj. neobradjeni, neuronske mreže postižu mnogo bolje rezultate nego ostale metode, jer one u suštini same kreiraju atribute na osnovu kojih vrše klasifikaciju.

11. **Kako biste napravili model koji na osnovu teksta mejla predvidja da li je mejl spam?**

    Pretpostavimo da imamo označenu bazu podataka, tj. da imamo tekstove mejlova i da li su spam ili nisu. Prvi zadatak je odabrati relevantne atribute. Posmatraćemo reči koje se često pojavljuju u spam mejlovima, a skoro nikad u regularnim mejlovima, kao i reči koje se retko pojavljuju u spam mejlovima, a često u regularnim mejlovima, i na osnovu tih reči ćemo napraviti kategoričke prediktore. Verovatno će nam prva analiza biti korisnija. Tom analizom ćemo ustanoviti da se reči kao što su lutrija, novac,

poklon, nasledstvo, popust, akcija, često pojavljuju u spam mejlovima, a retko u regularnim mejlovima, pa ćemo za svaku od tih reči napraviti prediktor. Dakle, kada dobijemo neki novi mejl, analiziraćemo njegove reči i ako sadrži neku od ovih reči (a pogotovo ako sadrži više njih), možemo taj mejl da označimo kao spam.

12. **Kako funkcionišu sistemi za preporuku, npr. kod Netflixa?**

Netflix čuva podatke o svim serijama i filmovima koje je neki korisnik pogledao. Te informacije su prediktori koje koristi za predvidjanje. Na osnovu toga nam predlaže sadržaje koji su slični pogledanim, na primer na osnovu žanra kom pripadaju ti sadržaji. Takodje uporedjuje nas sa drugim korisnicima, i nalazi korisnike koji su pogledali dosta istih sadržaja kao mi, i onda nam predlaže sadržaje koje su ti korisnici pogledali, a mi nismo.

13. **Kako izgleda funkcija greške u slučaju regresije, a kako u slučaju klasifikacije?**

U slučaju regresije najčešća funkcija greške je Mean squared error (MSE) i definisana je sa

$$MSE = \sum_{i=1}^{n} (y_i - \hat{y_i})^2$$

U slučaju regresije funkcija greške je unakrsna entropija (Cross entropy loss) i jednaka je negativnoj vrednosti logaritma funkcije verodostojnosti, odnosno:

$$Crossentropyloss = -\sum_{j=1}^{n}\sum_{i=1}^{k} p_{ji} log p_{ji},$$

gde je k broj mogućih kategorija, a p_{ji} verovatnoća da j-ta instanca pripada i-toj kategoriji.

14. **Koje su prednosti konvolutivnih mreža u odnosu na potpuno povezane?**

Prva prednost je specijalizovanost za topologiju signala. Naime, i potpuno povezane mreže bi mogle da se primenjuju na probleme vezane za zvuk, slike i slično, ali kod njih bi raspored piksela (ako radimo obradu slike) bio potpuno proizvoljan, pošto ne uzimaju u obzir susednost piksela na slici, dok su konvolutivne mreže konstruisane imajući u vidu da to što su pikseli jedni u okolini drugih ima poseban značaj. Druga prednost su proredjene interakcije, pod čime se podrazumeva da je svaka jedinica povezana samo sa malim brojem jedinica iz prethodnog sloja, umesto sa svim, kao u slučaju potpuno povezanih mreža. Na taj način se smanjuje broj parametara modela. Treća prednost je deljenje parametara, tj. sve jedinice jednog kanala imaju iste parametre – definisane filterom tog kanala. Kada parametri ne bi bili deljeni, kao kod potpuno povezane mreže, učeći

različite parametre za različite delove slike, mreža bi učila da nekim delovima ulaza pridaje posebnu semantiku. To može zvučati dobro, ali npr. u slučaju detekcije lica na slici, ne bi bilo dobro da se nauči da nos treba da bude baš na sredini slike, jer bi mogao biti i na nekom drugom mestu. Deljenje parametara omogućava da se nauči filter koji traži nos bilo gde na slici.

15. **Koji su osnovni tipovi neuronskih mreža i za šta se koriste?**

    Potpuno povezane neuronske mreže - najčešće se ne koriste samostalno, već kao sastavni deo drugih neuronskih mreža.
    Konvolutivne neuronske mreže - koriste se za obradu signala, poput zvuka i slike, ali takodje i teksta.
    Rekurentne neuronske mreže - koriste se za obradu sekvencijalnih podataka, kao što su obrada prirodnog jezika (eng *natural language processing* - NLP) i obrada vremenskih serija. Rekurentne mreže se sve manje koriste, jer ih u poslednje vreme zamenjuju transformeri koji postižu značajno bolje rezultate.

16. **Navesti primer kada je bitnije minimizovati false negatives u odnosu na false positives?**

    Na primer ako treba da na osnovu slike klasifikujemo da li je tumor benigni ili maligni. Neka benigni predstavlja negativnu kategoriju, a maligni pozitivnu. Mnogo gore posledice će imati ako maligni klasifikujemo kao benigni, nego da benigni klasifikujemo kao maligni. Dakle, mnogo nam je bitnije da minimizujemo false negatives.

17. **Koji algoritmi se koriste za smanjivanje dimenzija podataka?**

    Razlikujemo algoritme zasnovane na kovarijacionoj matrici i algoritme zasnovane na grafovskoj analizi podataka. Najpoznatiji predstavnik prve grupe je PCA, a predstavnici druge grupe su t-SNE i UMAP.
    UMAP je trenutno najpopularniji algoritam i više o njemu možete pogledati na sledećem linku.

18. **Ukratko objasniti logističku regresiju?**

    Logistička regresija se koristi za binarnu klasifikaciju (tj. kada zavisna promenljiva ima samo dve moguće klase). Ovaj metod se sastoji u tome da se izračuna vrednost logističke funkcije za ulaz X:

$$\frac{1}{1 + e^{-(1+\beta X)}}$$

    gde je β vektor parametara modela dobijen minimizovanjem funkcije gubitaka. Ako je vrednost logističke funkcije veća od threshold-a, dodelju se

klasa 1, a u suprotnom klasa 0. Threshold je najčešće postavljen na 0.5, ali može biti i neki drugi broj.

19. **Ukratko objasniti kNN?**

kNN je metoda nadgledanog učenja, koja može da se koristi i za regresiju i za klasifikaciju. Sastoji se u tome što se za neko fiksirano k posmatra k najbližih suseda (u odnosu na neku metriku) instance za koju želimo da izračunamo vrednost zavisne promenljive. Ako je u pitanju regresija, dodelićemo joj aritmetičku sredinu vrednosti zavisne promenljive posmatranih najbližih k suseda, a ako je u pitanju klasifikacija, dodelićemo joj onu klasu koja je najzastupljenija medju posmatranih k najbližih suseda.