Seminar 1

OBJAŠNJIVA UMJETNA INTELIGENCIJA (XAI)

Antonia Šarčević

Sadržaj

Uvod

Razvoj računalne snage, napredak algoritama učenja i dostupnost velike količine podataka omogućili su razvoj modela dubokog učenja koji daju izvrsne rezultate čak i za vrlo složene probleme. U današnje vrijeme takvi modeli umjetne inteligencije pronalaze široku primjenu u raznim područjima ljudskih djelatnosti, ali su i dalje neshvatljivi prosječnom korisniku što nerijetko uzrokuje nepovjerenje te se modeli čine manje pouzdanim od dobro proučenih i inherentno objašnjivih modela poput linearne i logističke regresije ili stabala odluke. Zbog toga bi mogućnost objašnjavanja modela dubokog učenja znatno doprinijela prihvaćanju dobivenih rezultata te omogućila korištenje umjetne inteligencije i za donošenje osjetljivijih odluka. Osim toga objašnjavanjem modela možemo osigurati da model nije pristran, ali i olakšati razvoj i otklanjanje grešaka u modelu. [1][2]

Zbog toga se razvila posebna grana umjetne inteligencije, XAI (eXplainable Artificial Intelligence), koja se bavi objašnjavanjem modela strojnog učenja. U nastavku će biti detaljnije objašnjeno što je XAI, vrste metoda objašnjavanja, pregled modela koji su objašnjivi sami po sebi i nekoliko popularnijih metoda za interpretaciju.

1. XAI

XAI (eXplainable Artificial Intelligence) se odnosi na skup metoda i tehnika koje omogućuju razumljivost i transparentnost odluka koje donose modeli strojnog učenja. [3] Time se osiguravaju pravednost odluke, zaštita osjetljivih informacija, robusnost modela i uzročno-posljedične veze te doprinosi povjerenju u sustav. No također treba razumjeti da nije uvijek potrebno i poželjno objašnjavati modele strojnog učenja.

Metode interpretacije mogu se klasificirati na temelju nekoliko kriterija navedenih u nastavku.

1.1. Inherentna i naknadna objašnjivost

Inherentna objašnjivost odnosi se na modele koji su lako shvatljivi zahvaljujući njihovoj jednostavnoj strukturi poput manjih stabala odluke i rijetkih linearnih modela. S druge strane, naknadna objašnjivost podrazumijeva primjenu interpretacijske metode nakon što je model već istreniran. Takve metode mogu se primijeniti i na inherentno objašnjive modele.[4]

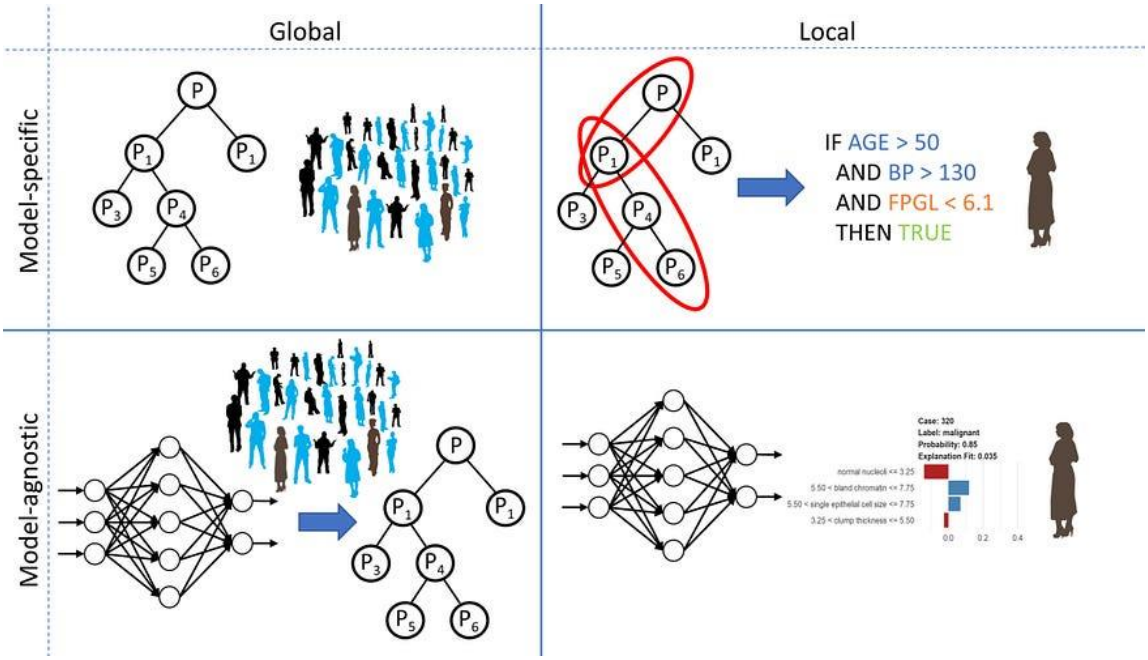
1.2. Specifične i opće metode

Specifične metode objašnjavanja ograničene su na određenu klasu modela. Primjer takve metode bila bi interpretacija težina linearnog modela koja je ograničena samo na taj skup modela ili metode specifične za neuronske mreže. S druge strane, opće metode mogu se primijeniti na bilo koji model dubokog učenja, ali naknadno, nakon treniranja modela. Takve metode se metode uglavnom baziraju na proučavanju izlaznih i ulaznih parova podataka, te nemaju pristup unutarnjim parametrima modela. [4]

1.3. Lokalne i globalne metode

Lokalne metode objašnjavaju određeni, individualni izlaz modela. Prednost lokalnog pristupa je što se čak i za složene modele lokalno predviđanje može monotono ili linearno ovisiti o određenim značajkama. Globalne metode, s druge strane, opisuju prosječno

ponašanje modela. One se primjenjuju na grupu instanci, koja se promatra kao čitav skup podataka ili primjenom lokalnih metoda na svakoj instanci. [4]



Metode objašnjavanja [5]

1.4. Rezultati interpretacijskih metoda

Interpretacijske metode mogu se podijeliti prema vrsti rezultata koje pružaju:

- **Sažeta statistika značajki** može biti sažetak za svaku značajku, poput jedne vrijednosti koja prikazuje važnost značajke, ili složenijih prikaza, kao što su interakcije između parova značajki
- **Vizualizacija sažetaka značajki** često je nužna za ispravnu interpretaciju
- **Interni parametri modela**. Interpretacija modela temelji se na lako razumljivim parametrima, poput težina u linearnim modelima ili strukture stabla odluke. Ovi parametri su specifični za model te se ponekad smatraju i sažetim statistikama značajki.
- **Točke podataka** objašnjavaju model kroz konkretne podatke, korisno za slike i tekst, ali manje za podatke s velikim brojem značajki.
- **Inherentno objašnjivi modeli.** Crne kutije mogu se objasniti pomoću aproksimativnih modela koji sami po sebi imaju jednostavnije, razumljive parametre. [4]

2. Inherentno objašnjivi modeli

Najjednostavniji način za postići objašnjivost je korištenjem modela koji su inherentno objašnjivi. Primjeri takvih modela bili bi linearna i logistička regresija te stabla odluke koji će biti detaljnije objašnjeni u nastavku.

ALGORITAM	Linearnost	Monotonost	Interakcija	Zadatak
Linearna regresija	DA	DA	NE	regr
Logistička regresija	NE	DA	NE	class
Stabla odluke	NE	Ponekad	DA	class,regr
RuleFit	DA	NE	DA	class,regr
Naivni Bayes	NE	DA	NE	class
k-najbližih susjeda	NE	NE	NE	class,regr

Tablica inherentno objašnjivih modela [4]

2.1. Linearna regresija

Linearna regresija izlaz modela računa kao sumu značajki pomnoženih s težinama što možemo zapisati kao:

$$h(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 + \ldots + w_nx_n$$

Iz ovakve hipoteze možemo jednostavno i intuitivno odrediti utjecaj pojedinih značajki na izlaz modela. Generalno možemo reći da značajke čije težine imaju veću apsolutnu vrijednost više doprinose predikciji modela. Naime važnost značajke računa se kao t-statistika, odnosno apsolutna vrijednost pripadajuće težine skalirana standardnom pogreškom.

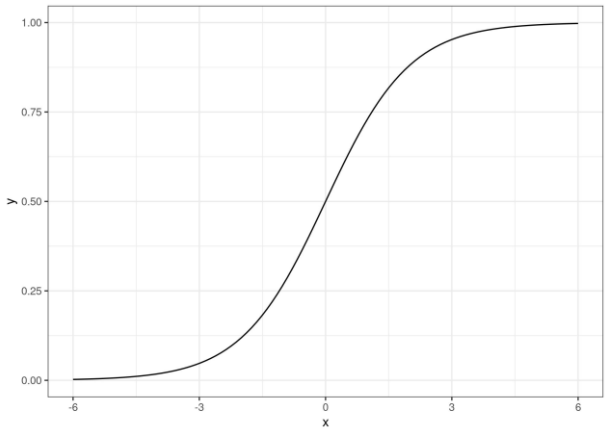
$$t_{w_i} = w_i / SE(w_i)$$

No kako bi model bio primjenjiv podaci moraju zadovoljavati određene zahtjeve poput: linearnosti, normalnosti, nezavisnosti, imati konstantne varijance i fiksirane oznake te ne smiju biti multikolinearni. Zbog relativno strogih zahtjeva ovakav model često nije dovoljno dobar za složenije probleme i skupove podataka. [4]

2.2. Logistička regresija

Model logističke regresije koristi logističku funkciju kako bi dobili izlaz između 0 i 1 koji interpretiramo kao vjerojatnost pripadnosti primjera određenoj klasi.

$$\text{logistic}(\eta) = 1 \,/\, (1+\exp(-\eta))$$



Logistička funkcija [4]

Model logističke regresije:

$$P(y_{(i)}=1) = 1/\,(1+\exp(-(\beta_0+\beta_1 x_{(i)1}+\ldots+\beta_p x_{(i)p})))$$

Zbog primjene logističke funkcije težine više ne utječu na izlaz linearno kao što je to bio slučaj za linearnu regresiju već promjena značajke za mjeru veličine mijenja omjer vjerojatnosti s faktorom $\exp(\beta)$ po izrazu:

$$\text{odds}_{x_j+1} \,/\, \text{odds}_{x_j} = \exp(\beta_j(x_j+1) - \beta_j x_j) = \exp(\beta_j)$$

koji slijedi iz:

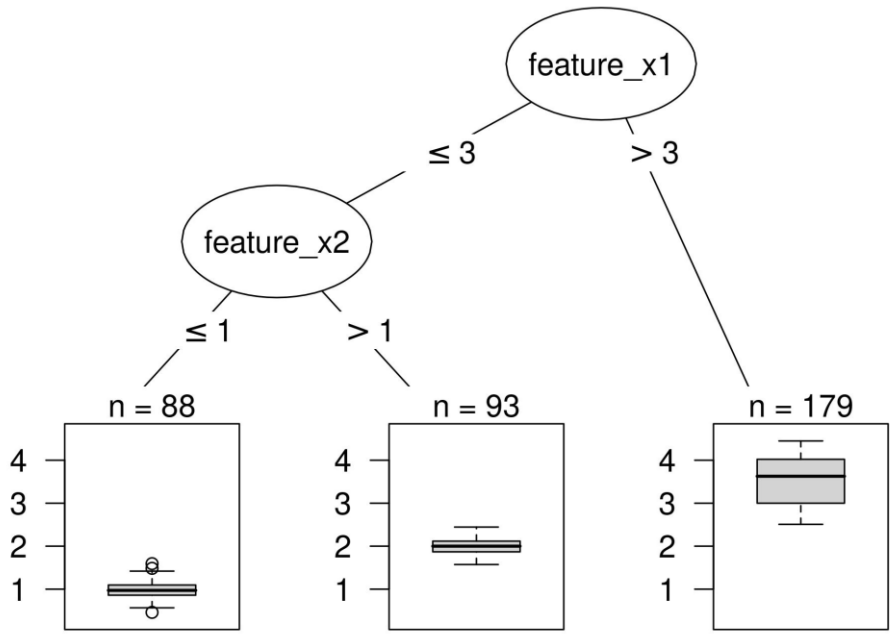
$$P(y=1) \,/\, (1-P(y=1)) = \text{odds} = \exp(\beta_0+\beta_1 x_1+\ldots+\beta_p x_p)$$

Iz toga je vidljivo da je interpretacija logističkih modela nešto zahtjevnija od interpretacije linearnih modela. [4]

2.3. Stabla odluke

Velika prednost stabala odluke je što ne zahtijevaju linearnost podataka već se baziraju na podjeli skupa podataka na maje skupove te se mogu koristiti iza klasifikaciju i za probleme regresije. Odnos izlaza y i ulaza x prikazan je formulom:

$$y = f(x) = \sum_{m=1}^{M} c_m I\{x \in R_m\}$$



Primjer stabla odluke [4]

Ovakva stabla vrlo su jednostavna za interpretaciju, počevši od korijena spušta se rema određenom podskupu podataka prema listovima koji određuju izlaz modela. Svi uvjeti na putu povezani su I operatorom.

Općenito važnost značajki u stablu odluke računa se tako da se prolaskom po putu uzima u obzir koliko je značajka smanjila entropiju u odnosu na roditeljski čvor.

Prednosti ovakvog pristupa su to što su stabla odluke idealna za opisivanje interakcija između značajki te stvara jasne grupe koje je jednostavno shvatiti i vizualizirati. No s druge strane nije dovoljno dobro rješenje za linearne podatke, te mu nedostaje zaglađenosti (oštre granice/podjele grupa) i poprilično je nestabilan model. [4]

3. Metode objašnjavanja

Jednostavniji modeli poput stabla odluke i linearnih modela mogu se jednostavno objasniti (white-box modeli), ali ne daju dobre rezultate na složenim skupovima podataka i kompleksnijim problemima. S druge strane modeli poput dubokih neuronskih mreža, dobro predviđaju i za složene probleme, ali ne znamo zašto donose pojedine odluke to jest ne daju zadovoljavajuća objašnjenja odluka, sama po sebi (black-box modeli). Zbog toga se za objašnjavanje ovakvih modela koriste ad-hoc metode koje interpretiraju rezultate neovisno o modelu. U nastavku je prikazano nekoliko popularnijih metoda za objašnjavanje modela. Sve navedene metode pridodaju određene vrijednosti značajkama ulaza zbog čega ih nazivamo atributnim metodama objašnjavanja.

3.1. LIME

LIME (Local Interpretable Model-agnostic Explanations) je kao što joj ime kaže opća metoda primjenjiva na sve vrste modela strojnog učenja, a objašnjava ponašanje modela lokalno, na nekom primjeru.

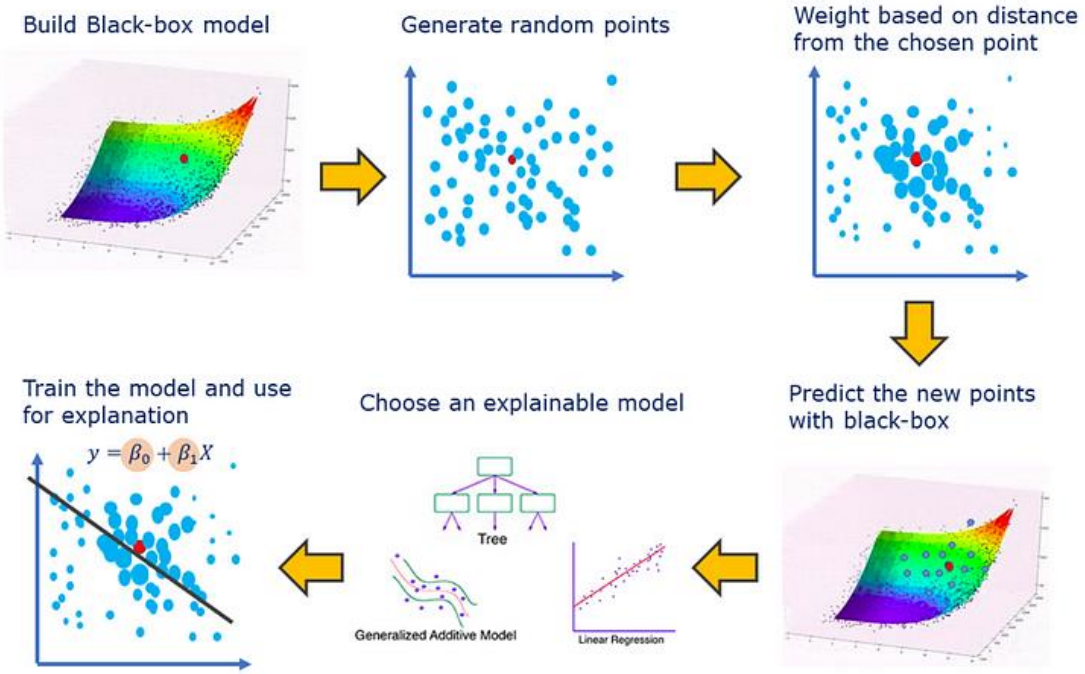
3.1.1. Algoritam

- Odabir modela i referentne točke
- Generiranje novog seta ulaznih podataka uzorkovanjem iz normalne distribucije odgovarajuće setu za treniranje
- Generiranje predviđanja korištenjem modela koji želimo objasniti
- Dodjeljivanje težina temeljenih na udaljenosti od referentne točke (koristi se RBF Kernel koji dodjeljuje veće težine bližim točkama)

$$RBF(x^{(i)}) = \exp\left(-\frac{||x^{(i)} - x^{(ref)}||^2}{kw}\right)$$

Gaussian Kernel formula, kw parametar određuje koliko je velik značajni krug značajki

- Treniranje surogat modela na generiranom skupu s težinama koji aproksimira ponašanje početnog modela u referentnoj točki i onda se on koristi za objašnjavanje. Surogat model može biti bilo koji inherentno objašnjivi model, a kao podrazumijevani model u Pythonu koristi se Ridge regresija. [6]

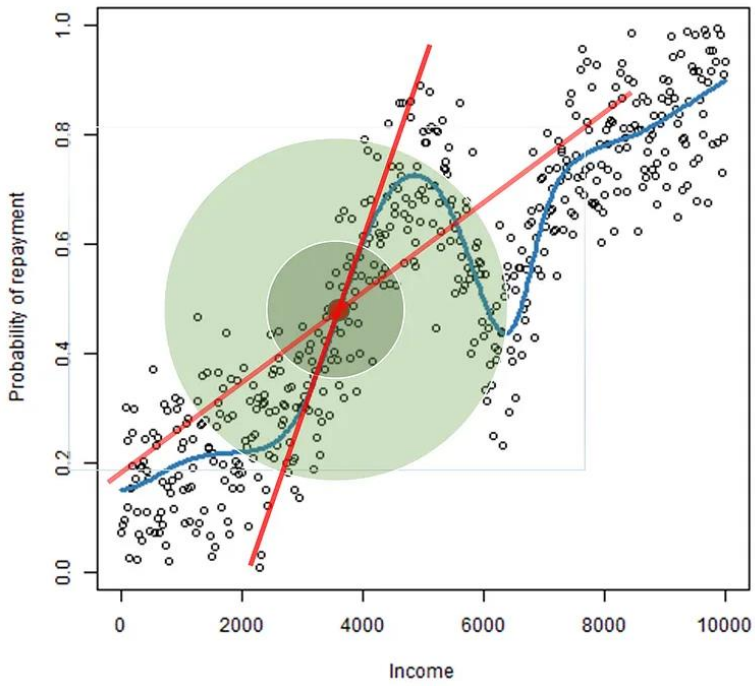


Koraci LIME algoritma [6]

3.1.2. Prednosti i nedostaci

Prednost LIME metode je njena jednostavnost i primjenjivost, naime LIME se može koristiti na različitim modelima i skupovima podataka poput tabličnih, tekstualnih i slikovnih podataka.

S druge strane problem generiranja podataka i dalje se raspravlja. Naime algoritam generira nasumične točke u prostoru, ali težine se primjenjuju samo na točke koje su dovoljno blizu referenci. Zašto onda ne generirati samo točke koje su blizu i kako odrediti na koje točke treba primijeniti težine. Ukoliko je maksimalna udaljenost prevelika, problem se neće moći uvijek dobro aproksimirati, s druge strane ograničavanje na premalu udaljenost rezultira nestabilnim modelima. [6]



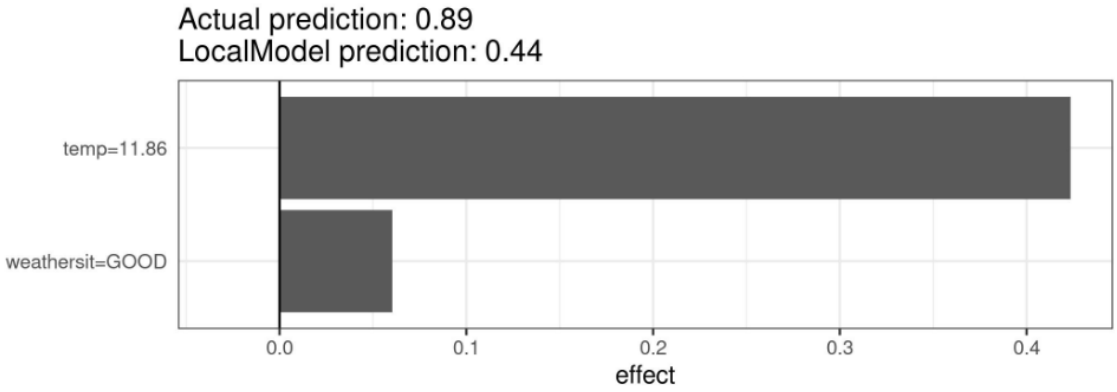
Ovisnost interpretacije o širini kernela [6]

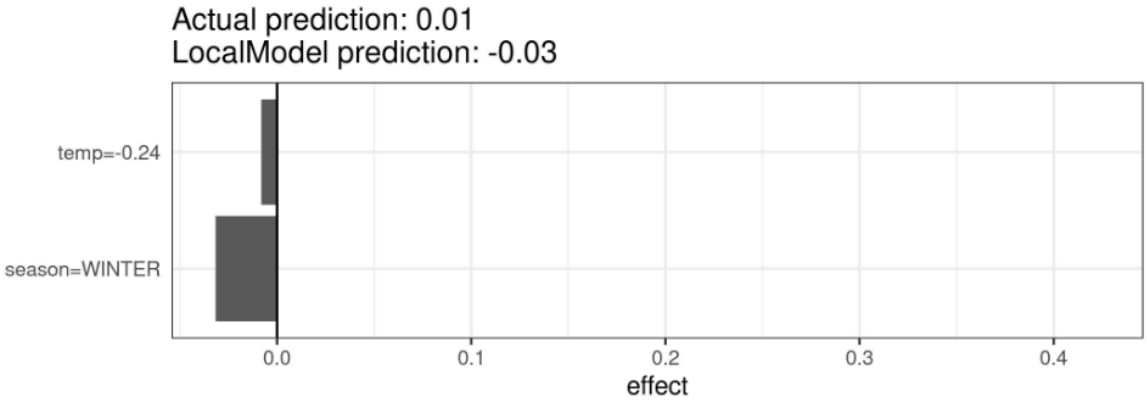
3.1.3. Primjeri

LIME podržava različite tipove podataka, a u nastavku će biti pokazani primjeri za tablične, tekstualne i slikovne podatke.

- **Tablični podaci:**

  Problem klasifikacije s podacima o najmu bicikala na temelju kojih se predviđa hoće li broj iznajmljivanja premašiti prosjek. Trenirana je slučajna šuma sa 100 stabala, te se pomoću LIME metode objašnjava predviđanje za dva primjera na temelju informacija o vremenu i temperaturi, pokazujući kako specifične značajke utječu na rezultate.





Visoka temperatura i dobro vrijeme imaju pozitivan utjecaj na predikciju. Efekt predstavlja težinu pomnoženu s vrijednošću značajke.

- **Tekstualni podaci:**

LIME za tekst funkcionira stvaranjem varijacija izvornog teksta uklanjanjem nasumičnih riječi te predstavlja svaku riječ kao binarnu (1 ako je uključena, 0 ako nije). Primjer uključuje klasifikaciju komentara na YouTubeu kao spam ili normalne. Vjerojatnost predikcije modela mijenja se ovisno o prisutnosti ili odsutnosti specifičnih riječi, pri čemu je "channel" prepoznat kao snažan indikator spama.

	SADRŽAJ	**RAZRED**
267	PSY is a good guy	0
173	For Christmas Song visit my channel! ;)	1

For	**Christmas**	**Song**	**visit**	**my**	**channel!**	**;)**	**VJV.**	**TEŽINA**
1	0	1	1	0	0	1	0.17	0.57
0	1	1	1	1	0	1	0.17	0.71
1	0	0	1	1	1	1	0.99	0.71
1	0	1	1	1	1	1	0.99	0.86
0	1	1	1	0	0	1	0.17	0.57

Slučaj	Vjerojatnost labele	Značajka	Težina značajki
1	0.1701170	is	0.000000
1	0.1701170	good	0.000000
1	0.1701170	a	0.000000
2	0.9939024	channel!	6.180747
2	0.9939024	;)	0.000000
2	0.9939024	visit	0.000000

- **Slikovni podaci:**

Za slike LIME ne mijenja pojedinačne piksele, već segmentira slike u takozvane superpiksele (grupe sličnih piksela). Primjer uključuje klasifikaciju slike kruha koristeći Googleov Inception V3 model, s objašnjenjima za oznake "Bagel" i "Strawberry." Zeleni segmenti povećavaju vjerojatnost klase, dok crveni smanjuju, pružajući vizualni uvid u to kako LIME ističe ključna područja na slici.

** Svi primjeri preuzeti su iz [4]

3.2. Shapely values

Ova metoda koristi se za prikaz važnosti pojedinih značajki za donošenje odluke, a bazira se na izračunu Shapelyjeve vrijednosti. Ideja potječe iz teorije kooperativnih igara i temelji se na podjeli zajedničkog dobitka ovisno o zaslugama pojedinih igrača. Ukoliko zadatak modela promatramo kao igru, a izlaz kao dobitak možemo igrače povezati s značajkama.

Shapelyjeve vrijednosti računaju se kao prosjek marginalnih doprinosa pojedinih značajki koji odgovaraju razlici u izlazu modela za ulaz sa i bez promatrane značajke za svaki podskup skupa preostalih značajki. Svakom doprinosu pridodaju se težine ovisno o udjelu značajki koje obuhvaćaju.

Formula za Shapelyjeve vrijednost za značajku i s funkcijom doprinosa v:

$$\phi_i(v) = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [v(S \cup i) - v(S)]$$

$$v_x(S) = \int f(x_1, ..., x_F) d\mathbb{P}_{x \notin S} - \mathbb{E}_X[f]$$

Pri čemu S označava broj značajki u određenom podskupu, a F ukupan broj značajki. Iz izraza za težinu (multinomijalni koeficijent) da metoda više kažnjava podskupove koji čiji je broj elemenata udaljeniji od 0 ili F. [9]

3.2.1. Algoritam

- Enumerirati sve moguće podskupove značajki za svaki primjer
- Izračunati predikcije za svaki podskup
- Pronaći marginalni doprinos značajke unutar svake predikcije podskupa
- Prosječno vrednovanje tih doprinosa kako bi se dobila Shapleyjeve vrijednost za tu značajku

3.2.2. Prednosti i nedostaci

Jedna od najvećih prednosti ove metode je njena pravednost u smislu da garantira teoretski pravedan način za raspodjelu zasluga svakoj značajki. Osmi toga ova metoda ne ovisi o modelu što ju čini široko primjenjivom.

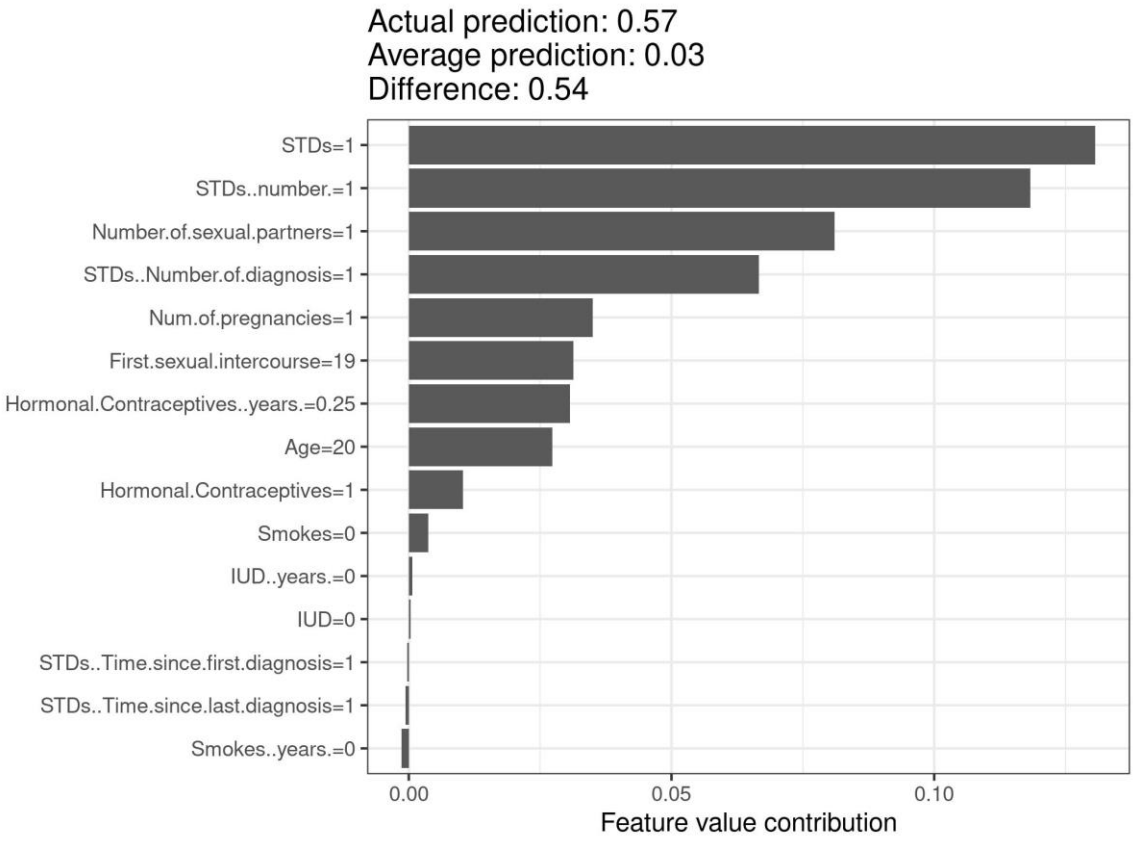
S druge strane složenost ovakvog postupka raste eksponencijalno ovisno o broju značajki, što metodu čini nepraktičnom za modele s velikom dimenzijom ulaza.

U praksi se često koriste aproksimativne metode poput SHAP (SHapley Additive exPlanations) koje ubrzavaju izračun Shapleyjeve vrijednosti, a zadržavaju većinu interpretabilnosti.

3.2.3. Primjeri

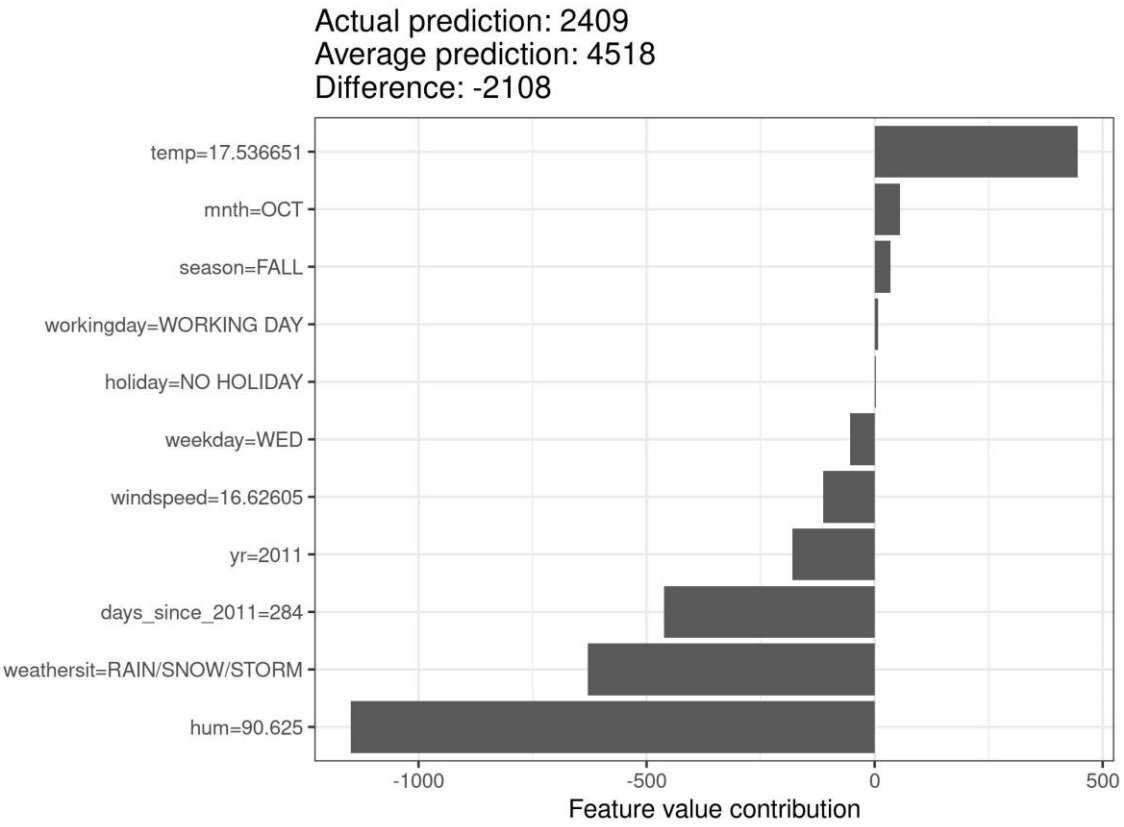
Interpretacija Shapleyjeve vrijednosti za značajku j je sljedeća: Vrijednost j-te značajke pridonijela je predikciji za određeni primjer s vrijednošću ϕ_j u usporedbi s prosječnom predikcijom za cijeli skup podataka. Shapleyjeva vrijednost primjenjiva je i na klasifikaciju (kada radimo s vjerojatnostima) i na regresiju.

Analiza predikcije pomoću modela slučajne šume za procjenu rizika raka vrata maternice:



Za ženu iz skupa podataka s predikcijom od 0,57, rizik od raka je 0,54 iznad prosječne predikcije (0,03). Najveći doprinos povećanju vjerojatnosti imala je dijagnosticirana spolno prenosiva bolest (STD). Zbroj doprinosa odgovara razlici između stvarne i prosječne predikcije (0,54).

Kod skupa podataka o najmu bicikala:



Za određeni dan (dan 285), predviđena je vrijednost od 2409 bicikala, što je -2108 ispod prosjeka (4518). Vremenski uvjeti i vlažnost dali su najveći negativni doprinos, dok je temperatura imala pozitivan utjecaj. Zbroj Shapleyjevih vrijednosti daje razliku između stvarne i prosječne predikcije (-2108).

Važno je točno interpretirati Shapleyjevu vrijednost: ona predstavlja prosječni doprinos određene značajke predikciji unutar različitih kombinacija značajki. Shapleyjeva vrijednost nije jednostavno razlika u predikciji ako bismo značajku izostavili iz modela.

** Svi primjeri preuzeti su iz [4]

3.3. Saliency maps (Pixel Attribution)

Saliency mape su tehnika objašnjavanja strojnog učenja koja se koristi u računalnom vidu kako bi se vizualiziralo koji dijelovi slike najviše doprinose odluci neuronske mreže. Ove mape pokazuju relevantne dijelove slike ističući piksele koji imaju najveći utjecaj na odluku modela. To omogućuje razumijevanje onoga što model smatra važnim za donošenje određene odluke. [11]

Metode poput LIME i Shapely values manipuliraju djelom ulaza kako bi generirali rješenja. S druge strane neke metode poput Sailency mapa računaju gradijent izlaza u odnosu na ulazne značajke. To jest izračunava se kako se promjena pojedinog piksela odražava na promjenu izlaza modela. Rezultat se uglavnom prikazuje u obliku mapi, gdje su važnija područja označena svjetlijim ili istaknutijim bojama. [4]

Sailency mape [11]

3.3.1. Algoritam

- Primijeniti prolaz unaprijed na zadanu sliku to jest izračun izlaza istreniranog modela
- Unatražnom propagacijom kroz mrežu računa se gradijent izlazne vrijednosti u odnosu na svaki piksel.

$$E_{grad}(I_0) = \frac{\delta S_c}{\delta I}|_{I=I_0}$$

- Vizualizacija gradijenata. Moguće je prikazati apsolutne vrijednosti gradijenta za pojedini piksel ili naglasiti negativne i pozitivne doprinose odvojeno [4]

3.3.2. Prednosti i nedostaci

Velika prednost Sailency mapa je vizualizacija objašnjenja što ga čini lakšim za shvatiti. Osim toga metode bazirane na gradijentima su brže od metoda koje manipuliraju ulaz kao LIME i Shapely values.

S druge strane teško je raspoznati je li objašnjenje doista točno te se pokazalo da je metoda nestabilna. Naime za male promjene u slici koje ne utječu na klasifikaciju došlo je do velikih promjena u objašnjenju modela. Sve to ovu metodu čini vrlo nepouzdanom. [4]

Literatura

[1] https://www.sciencedirect.com/science/article/pii/S1566253523001148 (29. 10. 2024)

[2] Linardatos, P.; Papastefanopoulos, V.; Kotsiantis, S. Explainable AI: A Review ofMachine Learning Interpretability Methods. Entropy 2021, 23, 18 https://dx.doi.org/0.3390/e23010018

[3] https://www.geeksforgeeks.org/explainable-artificial-intelligencexai/ (29. 10. 2024)

[4] https://christophm.github.io/interpretable-ml-book/ (29. 10. 2024)

[5] https://towardsdatascience.com/explainable-machine-learning-9d1ca0547ae0 (29. 10. 2024)

[6] https://towardsdatascience.com/lime-explain-machine-learning-predictions-af8f18189bfe (29. 10. 2024)

[7] https://www.geeksforgeeks.org/introduction-to-explainable-aixai-using-lime/ (29. 10. 2024)

[8] https://www.kaggle.com/code/prashant111/explain-your-model-predictions-with-lime (29. 10. 2024)

[9] https://medium.com/data-reply-it-datatech/explainable-ai-shapley-values-and-lime-5f14d42147b3 (29. 10. 2024)

[10] https://www.kaggle.com/code/prashant111/explain-your-model-predictions-with-shapley-values?scriptVersionId=28578470 (29. 10. 2024)

[11] https://medium.com/@bijil.subhash/explainable-ai-saliency-maps-89098e230100 (30. 10. 2024)