Detekcija I otklanjanje autlajera primenom modela robusne linearne regresije

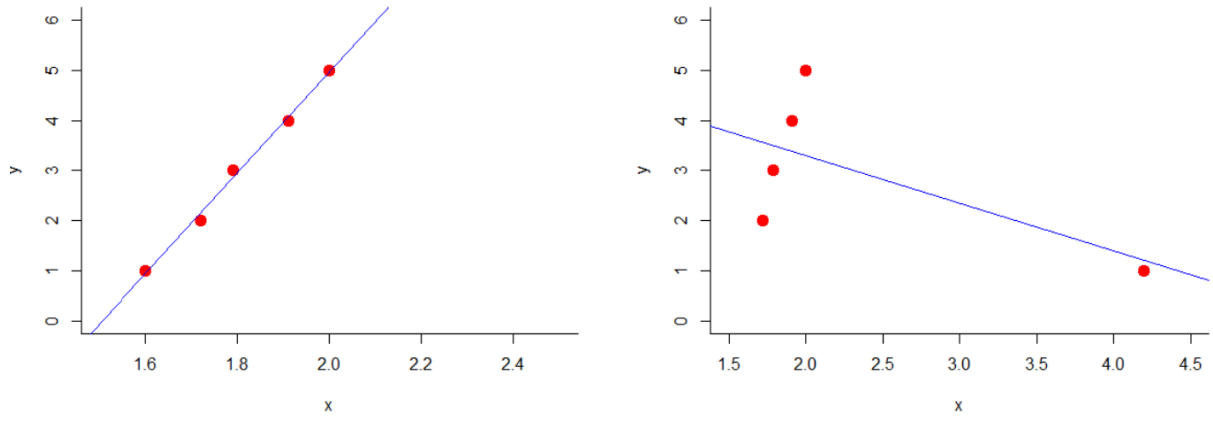
1. Opis problema

Kako bismo definisali naš problem potrebno je prvo da objasnimo šta su autlajeri. To su tačke čije vrednosti odstupaju od šablona.

Uklanjanje autlajera ima za cilj poboljšanje tačnosti i stabilnosti modela. Eliminišući ih, model postaje pouzdaniji jer se fokusira na uopštenije obrasce u podacima što omogućava bolje predviđanje novih vrednosti na temelju stvarnih trendova u skupu podataka. Ovo sve na kraju rezultira modelom koji bolje opisuje stvarni odnos između nezavisnih i zavisne promenljive i čini ga korisnijim za praktičnu primenu.

Kao što možemo da vidimo na slici 1, parametri naše regresione prave u mnogome zavise od jedne tačke. Potrebne su nam metode koje su otporne na ovakve tačke(autlajere). Takve metode zovu se *robusne metode* i za njih je definisana *breakdown* vrednost.

Breakdown vrednost predstavlja broj tačaka koje možemo zameniti proizvoljno velikim vrednostima u našem skupu podataka a da parametri regresione prave ostanu nepromenjeni. Za OLS(*Ordinary Least Squares*) metodu korišćenu u višestrukoj linearnoj regresiji vrednost ovog parametra iznosi 1/n (čim zamenimo jednu tačku izgled naše regresione prave menja se u potpunosti). Dok kod robusnih metoda *breakdown* vrednost može iznositi i 50%.



Slika 1.

2. Skup podataka

Skup podataka nad kojim ćemo primeniti robusni model linearne regresije je *Electic Vehicle Prices* sa *Kaggle.com* (https://www.kaggle.com/datasets/fatihilhan/electric-vehicle-specifications-and-prices). Dataset sadrži 9 kolona(atributa) i 360 redova. Kolona čiju vrednost želimo da predvidimo u dataset-u je *price* i kao što joj samo ime kaže predstavlja cenu automobila. Najznačajnije nezavisne promenljive su:

- *Battery* - predstavlja kapacitet baterije
- *Efficiency* - predstavlja efikasnost vozila(Wh/km)
- *Fast_charge* - predstavlja dužinu punjenja u minutima
- *Range* - predstavlja doseg automobila sa punom baterijom
- *Top_speed* - predstavlja maksimalnu brzinu
- *Acceleration..0.100* - predstavlja vreme potrebno da dostigne brzinu od 100 km/h počevši u stajaćoj poziciji.

3. Metodologija

Prvi korak predstavlja pretprocesiranje podataka iz našeg dataset-a. Primenićemo interpolaciju splajnom kako bi popunili nedostajuće vrednosti. Nakon što smo pripremili podatke za rad nad njima podelićemo ih na *train/test/val* segmente u odnosu 60/20/20.

Kako nam je cilj da prikažemo uticaj autlajera na parametre regresione prave, fitovaćemo model višestruke linearne regresije koristeći OLS(*Ordinary Least Squares*) metod nad trening podacima, osigurati što bolju meru nad validacionim podacima i na kraju prijaviti meru koju je model postigao nad test podacima.

Nakon toga ćemo, istim postupkom, fitovati modele robusne linearne regresije koristeći metode: LMS(*Least Median Squares*), LTS(*Least Trimmed Squares*), WLS(*Weighted Least Squares*), *Theil-Sen estimator*, RANSAC(*Random Sample Consensus*) i *Huber regression*.

Pored standardnih metoda robusne linearne regresije koristićemo i *DB-Scan*. Tako što ćemo sa DB-Scan-om preprocesirati podatke pa nad njima primeniti OLS, a potom nastaviti istim postupkom kao i kod pređašnjih metoda.

U nastavku sledi objašnjenje datih metoda.

LMS – je regresioni model koji minimizuje medijan grešaka. Kako je definicija medijana vrednost koja se nalazi u sredini nekog skupa elemenata, autlajeri uopšte ne utiču na njega jer su to vrednosti koje se nalaze na jednom ili drugom kraju ekstrema.

LTS - koristi takozvani *bootstrap approach*. Prvo ćemo iz skupa uzeti određeni deo podataka, fitovati model primenom OLS-a, pa potom vratiti naše podatke u skup i to ponoviti M puta. Nakon toga ćemo izabrati parametre koji su najbolje modelovali skup

podataka. Ovim smo imitirali čitavu populaciju podataka i na taj način probali da eliminišemo autlajere.

WLS – je regresioni model koji pridodaje težinu svakoj grešci, gde težina predstavlja množenje greške sa realnim brojem. Nakon toga, novodobijene vrednosti grešaka minimizuje kao OLS.

Theil-Sen estimator – je regreesioni model koji za svaku tačku posebno računa parametre linearne regresije po svakoj dimenziji i na kraju uzmemo median od svih vrednosti za svaki parametar respektivno.

RANSAC – slično kao LTS, M puta uzimamo deo podataka našeg skupa i nad njima računamo parametre modela. Razlika je u tome što RANSAC prilikom računanja parametara zanemari tačke koje su na rastojanju većem od unapred definisanog. Nakon toga za vrednost parametara biramo one koji su najbolje modelovali skup podataka.

Huber regression – koristi drugačiju loss funkciju na osnovu udaljenosti tačke od prave i pridodaje težinu greškama. Za tačke blizu prave minimizujemo kvadrate grešaka, a za tačke koje su od prave udaljene više od unapred definisanog parametra *delta* minimizujemo apsolutnu grešku.

DB-Scan – služi kao algoritam za preprocesiranje podataka grupišući ih u klastere. Podaci su grupisani u klastere na osnovu dva paramterta, *epsilon* što predstavlja maksimalnu distancu između dve tačke u istom kalsteru i *min_samples* što predstavlja minimalan broj tačaka u jednom klasteru. Tačke koje nisu grupisane u klastere izbacujemo iz našeg skupa podataka. Nakon pretprocesiranja skupa podataka, primenićemo OLS i fitovati model linearne regresije.

4. Način evaluacije

Metrika koju ćemo koristiti kao mera kvaliteta svih naših modela biće MSE(Mean Squared Errors). Nakon evaluacije modela, uporedićemo model višestruke linearne regresije sa modelima robusne linearne regresije, a potom i međusobno modele robusne linearne regresije i diskutovati o rešenju, u smislu prednosti i mana od svakog modela respektivno.

$$MSE = \left(\frac{1}{n}\right)\sum(i = 1\ to\ n)(y_i - \hat{y}_i)^2$$

5. Tehnologije u izradi

Koristićemo programski jezik Python(verzija 3.12).

Od biblioteka koristićemo Numpy, Matplotlib, Sklearn i Pandas.

6. Podela rada

SV8/2022 Nikola Velemir – Fitovanje modela koristeći OLS, LTS, Theil-Sen estimator i Huber regression

SV13/2022 Mihajlo Orlović – Fitovanje modela koristeći LMS, WLS, RANSAC i DB-Scan

7. Literatura

- https://repozitorij.pmf.unizg.hr/islandora/object/pmf%3A10145/datastream/PDF/view
- https://www.baeldung.com/cs/ransac
- https://en.wikipedia.org/wiki/Weighted_least_squares
- https://home.olemiss.edu/~xdang/papers/MTSE.pdf
- https://en.wikipedia.org/wiki/Huber_loss
- https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html
- https://www.youtube.com/watch?v=AN3UkzE3HMg&list=PLqzoL9-eJTNAB5st3mtP_bmXafGSH1Dtz