Detectarea anomaliilor într-un set de date

Bontaș Cezar - Octavian

Ioniță Cosmin – Ștefan

332AC

Introducere

În cadrul acestui proiect vrem să studiem cum putem descoperi anomaliile într-un set arbitrar de date, analizând distribuția lor, apoi selectând o metodă de modelare nesupervizată care se potrivește cu cazul de față, vom încerca să ajustăm eventualii parametrii ai funcției astfel încât să obținem cel mai bun scor.

Analiza setului de date

Proiectul începe cu o scurtă analiză a setului de date în care sunt evidențiate minimul, maximul și în special media și deviația standard pe fiecare coloană pentru a le folosi în a centraliza datele.

	id	feature_0	feature_1	feature_2	feature_3	feature_4	feature_5	is_anomaly
count	144.000000	144.000000	144.000000	144.000000	144.000000	144.000000	144.000000	144.000000
mean	111.326389	64.226751	19.561420	54.079310	44.703726	117.250586	35.702038	0.097222
std	66.748335	18.145381	10.436813	18.725270	14.851784	13.590508	45.957506	0.297294
min	1.000000	30.577565	-3.653628	18.995085	12.791948	70.660124	-9.957228	0.000000
25%	55.750000	48.680970	12.794194	39.482445	33.876683	110.247439	5.182902	0.000000
50%	106.500000	65.071287	17.890558	51.512030	43.373902	117.117564	27.227696	0.000000
75%	164.750000	76.047032	24.860083	66.524102	55.151957	124.157933	54.205813	0.000000
max	239.000000	128.573395	48.715990	94.829946	122.478511	163.339675	419.262574	1.000000

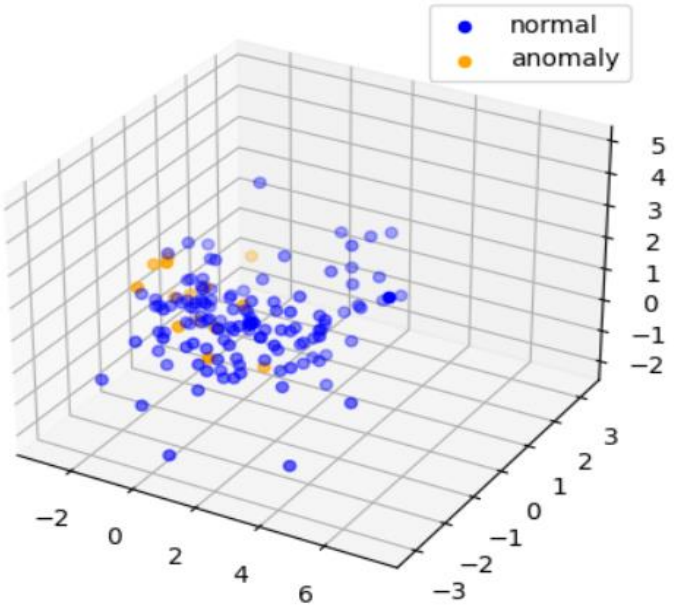
train.csv

	id	feature_0	feature_1	feature_2	feature_3	feature_4	feature_5
count	192.000000	192.000000	192.000000	192.000000	192.000000	192.000000	192.000000
mean	209.630208	62.250398	17.926074	55.008894	44.286810	114.165483	30.097154
std	95.137452	16.097708	10.451460	19.883281	12.812600	13.572711	28.192111
min	0.000000	26.997378	-7.650561	14.221737	11.204973	82.934180	-12.665816
25%	149.250000	51.828889	11.609624	40.304586	34.351519	105.094410	3.563398
50%	239.000000	61.485335	17.054633	53.299112	45.727571	114.652847	29.109324
75%	287.250000	74.644358	23.415473	64.569508	53.024391	123.144293	46.633370
max	335.000000	95.901798	51.912726	126.299809	70.361605	159.572384	117.367656

test.csv

Observăm atât pentru setul de date de antrenament cât și pentru cel de test că anumite caracteristici, aici fiind 4 și 6, că există o diferență considerabilă dintre valoarea maximă și valorile aparținând celor mai mari 75%. Acestea adaugă zgomot în distribuție, și cresc greutatea cu care modelul poate prezice cu precizie.

Pentru analiza statistică vom avea nevoie de ajutorul librăriilor matplotlib și sklearn pentru a ne centraliza datele din fișierul train.csv și a le pune într-un grafic 3D pentru a determina tipul de anomalii. Metodele folosite sunt calcularea lui Z-Score, implementat în „sklearn.scaler”, și metodei PCA pentru a reduce dimensiunea datelor astfel încât să le reprezentăm într-un spațiu 3D.



Putem observa că distribuția datelor pe modelul de antrenament este una unde cele mai multe puncte, atât normale cât și anomalii, se concentrează într-un cluster. Detectarea anomaliilor este dificilă încât anomaliile sunt distribuite local, apropiate de punctele normale. De asemenea, față de o problemă simplă de detectare a anomaliilor, unde datele anormale se regăsesc pe marginea clusterului sau în afară, aici majoritatea anomaliilor sunt „ascunse” printre punctele normale.

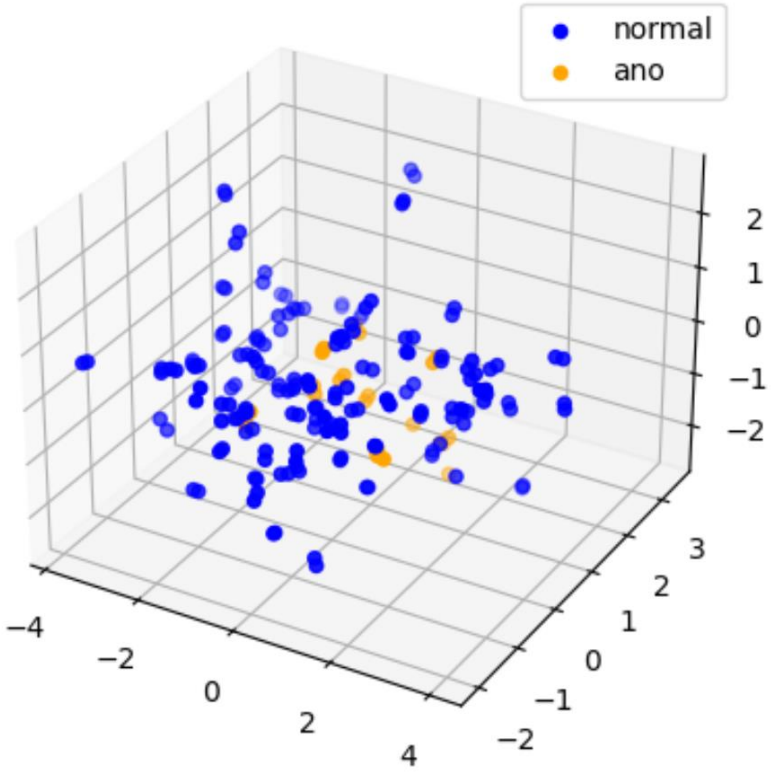
Metoda folosită și procesul de antrenare

Din datele studiate anterior, putem să excludem anumite seturi modalități de antrenare a modelului, întrucât nu se potrivesc cu setul nostru:

- LOF: se bazează pe distanțele pe o vecinătate de puncte. Ca să meargă, ar trebui ca aceste puncte să fie mult mai îndepărtate de punctele normale, altfel riscăm să detectăm multe falsuri pozitive;
- Angle-based Outlier Detection: anomaliile sunt distribuite strâns față de alte puncte normale, așadar rata de fals pozitiv va crește mult;
- Isolation Forest: deși se potrivește mai bine pe cazul nostru (anomaliile sunt puține și răsfirate între ele), faptul că distanța între anomalii si punctele normale este mică poate induce probleme de detecție în cazuri nișă.

Așadar alegerea noastră în cazul de față este să ne construim un SVC, unde nucleul funcției va fi RBF (radial basis function). Acesta ne ajută să construim arii de detecție pentru punctele noastre de anomalii, cu hiperparametri ajustabili astfel încât să ajungem la o rată de detecție mai bună.

Aplicând un model SVC cu parametrii C = 25 si gamma = 0.4, modelul nostru are o rată de predicție de 72.5%. Mai jos avem graficul care reprezintă etichetarea datelor de test.



După cum observăm, anomaliile au o distribuție similară cu cele din setul de antrenament.

Concluzii

Rata de detecție ar putea fi îmbunătățită în felul următor:

- Găsirea unui model pentru cazul de față;
- Ajusarea hiperparametriilor pentru SVC ca să aducem modelul mai apropiat de realitate; este posibil ca parametrul C, fiind ridicat, să fii creat overfitting pe datele noastre;
- Combinarea modelelor, fie după aceeași metodă, fie combinând metode diferite, prin tehnici diferite (de exemplu bagging, ensembling, stacking, etc.) pentru a compara anumite modele și hiperparametrii lor spre a aduce predicția către cel mai apropiat rezultat;

Bibliografie

1. https://scikit-learn.org/stable/modules/svm.html
2. https://www.analyticsvidhya.com/blog/2021/06/support-vector-machine-better-understanding/
3. https://aryanbajaj13.medium.com/ensemble-models-how-to-make-better-predictions-by-combining-multiple-models-with-python-codes-6ac54403414e