Kategorizácia syntetických zemetrasení pomocou strojového učenia

Vypracoval : Jakub Parada
Vedúci projektu : doc. RNDr. František Gallovič, Ph.D

Pracovisko : Katedra geofyziky
Ukončenie projektu : 30.6.2022

1 Abstrakt

Tektonické zemetrasenia predstavujú dominantný a často aj deštruktívny zdroj seizmických vĺn. Z toho plynie potreba dobre pochopiť procesy odohrávajúce sa v seizmickom ohnisku, kde dochádza k vzájomnému posuvu (sklzu) horninových blokov pozdĺž aktívnych zlomov. Tento proces je z fyzikálneho hľadiska riadený trením. Úloha, kedy sa pre predpokladaný model trenia snažíme určiť jeho parametre a predpätie na zlome pomocou modelovania nameraných seizmogramov sa nazýva dynamická sklzná inverzia.

Spojitosť medzi parametrami a seizmogramami je silne nelineárna, čo robí z dynamických inverzií výpočetne enormne náročnú úlohu, obzvlášť pokiaľ je formulovaná rigorózne v bayesovkom formalizme (Gallovič a kol., 2019). Štandardne používané metody typu Markov Chain Monte Carlo nemôžu štartovať z čisto náhodných modelov, ale iba z takých, ktoré minimálne vedú k šíriacej sa trhline po zlomovej ploche, prípadne k základnému vystihnutiu nameraných seismogramov. Je teda nutné pred samotnou inverziou roztriediť (kategorizovať) možné modely na vhodné a nevyhovujúce, a to napríklad pomocou metód strojového učenia a neurónových sietí. Práca by sa inšpirovala nedávno publikovaným článkom Ahamed a Daub (2021).

Metódy strojového učenia vedia veľmi dobre modelovať nelineárne závislosti, vďaka čomu vedia veľmi dobre identifikovať komplexné závislosti v množine dát. V tejto práci skúšame viaceré metódy strojového: neurónové siete a kombinácie rozhodovacích stromov (Gradient boosted decision trees, Random forest classifier).

V tejto práci boli využité metódy strojového učenia z knižnice scikit-learn, verzie 1.0.1 pre jazyk Python 3.9+.

2 Dataset

Náš dataset obsahuje 3600 modelov zemetrasení opísaných "slip-weakening"zákonom trenia s maximálnou veľkosťou danou elipsou s veľkosťami hlavných poloos 20 km a 5 km na rovinnom zlome v homogénnom elastickom prostredí. Ako počiatočné parametre berieme:

```
#Nucl_X Initial_stress Strength_excess Dc_0 Dc_Rate
```

kde Nucl_X je poloha nukleačného bodu od ľavej strany zlomu (vo vertikálnom smere je v strede), Initial_stress je pomer počiatočného a normálového napätia, Strength_excess je pomer rozdielu mezi pevnosťou a počiatočným napätím voči normálovému napätiu. Dc_0 a Dc_Rate popisujú lineárny rast parametru Dc v zmysle minimálnej hodnoty a gradientu.
Výsledky dynamických simulacií šírenia trhliny:

```
Seismic_Moment Ruptured_Area Rupture_velocity Stress_drop E_g E_r
```

V tejto práci sa zameriavame len na klasifikáciu zemetrasení v závislosti od `Ruptured_Area`, konkrétne za úspešné považujeme len tie modely prasknutia zlomu, ktoré majú veľkosť `Ruptured_Area` aspoň 95% maxima (tj. obsahu celej elipsy).
Ukážka reálnych dát z datasetu:

#Nucl_X (m)	Initial_stress	Strength_excess	Dc_0 (m)	Dc_Rate (m/km)	Seismic_Moment (Nm)
1000.	0.05	0.01	0.1	0.0	0.99035E+18
1000.	0.05	0.01	0.1	0.03	0.12659E+18
1000.	0.05	0.01	0.1	0.06	0.39177E+17
1000.	0.05	0.01	0.1	0.09	0.31950E+17

Ruptured_Area (m^2)	Rupture_velocity (m/s)	Stress_drop (Pa)	E_g (J)	E_r (J)
0.80780E+08	0.32727E+04	0.56912E+07	0.53267E+14	0.33595E+14
0.16810E+08	0.27505E+04	0.87240E+07	0.12186E+14	0.48337E+13
0.50900E+07	0.34148E+04	0.19129E+08	0.69627E+13	0.45870E+13
0.38300E+07	0.35088E+04	0.22469E+08	0.66940E+13	0.43700E+13

Pre potreby strojového učenia sú hodnoty v datasete normalizované pomocou `sklearn.preprocessing.MinMaxScaler()`, pre klasifikačné metódy sú hodnoty v stĺpci `Ruptured_Area` nahradené 0, ak daný model šírenia považujeme za neúspešný alebo 1, ak ho považujeme za úspešný.

3 Metódy strojového učenia

3.1 MLP Classifier

Multi-layer Perceptron (odteraz MLP) je typ feed forward neural networku pozostávajúceho zo vstupnej, skrytých a výstupnej vrstiev. Vrstvy sa skladajú z neurónouv, ktoré sú napojené na všetky neuróny v predošlej a nasledujúcej vrstve. Výpočet prebieha dosadením vstupných dát do neurónov vstupnej vrstvy a potom postupným rátaním hodnôt v ďalších vrstvách. Pri tréningu sme použili nasledovné parametre:

```
hidden_layer_sizes    (32, 32, 32, 32)
activation            'tanh'
solver                'adam'
alpha                 0.01
learning_rate_init    0.001
max_iter              600
random_state          69
tol                   0.0001
max_fun               15000
```

Zvyšné parametre boli ponechané ako default z knižnice scikit-learn.

3.2 Random Forest Classifier

Random Forest Classifier je Classifier, ktorí používa viacej slabých Classifierov, v tomto prípade Decision trees, ktoré skombinuje, a tým vznikne lepší Classifier. Pri tréningu sme použili nasledovné parametre:

```
n_estimators          200
criterion             'gini'
max_depth=10          10
min_samples_split     2
min_samples_leaf      1
random\_state         69
```

Zvyšné parametre boli ponechané ako default z knižnice scikit-learn.

3.3 Gradient Boosting Classifier

Gradient Boosting Classifier podobne ako Random Forest kombinuje viacero slabých Classifierov, ale v aditívnom štýle, teda pri trénongu sú slabé Classifiere trénovaná postupne s tým, že neskoršie sa učia na chybách predošlých. Pri tréningu sme použili nasledovné parametre:

```
loss                  'exponential'
learning_rate         0.1
n_estimators          200
criterion             'friedman\_mse'
min_samples_split     2
max_depth             3
min_samples_leaf      1
random_state          69
tol                   1e-4
```

Zvyšné parametre boli ponechané ako default z knižnice scikit-learn.

3.4 MLP regressor

MLP regressor funguje rovnako ako Classifier, ale na rozdiel od Classifiera ktorý predpovedá pravdepodobnosť že model zemetrasenia je úspešný, predpovedá regresor jedno reálne číslo - naškálovanú veľkosť praskliny. Pri tréningu sme použili nasledovné parametre:

hidden_layer_sizes	(64, 64, 64, 64)
activation	'tanh'
solver	'lbfgs'
alpha	0.01
learning_rate_init	0.001
max_iter	10000
random_state	69
tol	1e-7
max_fun	50000

Zvyšné parametre boli ponechané ako default z knižnice scikit-learn.

4 Výsledky

V nasledovnej tabuľke sú zhrnuté výsledky tréningu. Vstupný dataset sme rozdelili na trénovaciu množinu s 2880 modelmi zemetrasení a testovaciu s 720 modelmi. V prípade regresoru sme číselný výstup klasifikovali ručne porovnaním s 0.95 násobku najväčšieho vydeného prasknutia a v prípade Classfierov sme zvolili threshold na binarizáciu ako 0.5 (každý classifier predpovedá pravdepodobnosť že daný model zemetrasenia je úspešný, ak je táto pravdepodobnosť väčšia ako threshold tak model zemetrasenia prehlásime za úspešný, v opačnom prípade za neúspešný). Všetky modely sa trénovaly niekoľko sekúnd až minút a predikcie sú skoro okamžité.

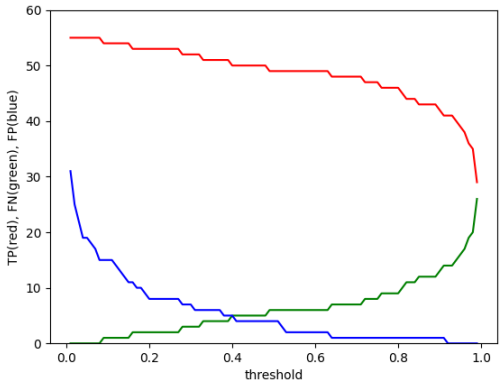
V nasledovnej tabuľke TP znamená true positive, FN false negative, FP false positive a TN je true negative. Precision sa počíta ako $\frac{TP}{TP+FP}$, recall ako $\frac{TP}{TP+FN}$, f1 ako $\left(\frac{Recall^{-1}+Precision^{-1}}{2}\right)^{-1}$ a accuracy ako $\frac{TP+TN}{TP+FN+FP+TN}$.

MLP Classfier		Random forest Classifier		Gradient Boosting Classifier		MLP regressor	
TP:	49	TP:	45	TP:	48	TP:	50
FN:	6	FN:	10	FN:	7	FN:	5
FP:	4	FP:	1	FP:	1	FP:	14
TN:	661	TN:	664	TN:	664	TN:	651
precision:	0.9245	precision:	0.9783	precision:	0.9796	precision:	0.7813
recall:	0.8909	recall:	0.8182	recall:	0.8727	recall:	0.9091
f1:	0.9074	f1:	0.8911	f1:	0.9231	f1:	0.8403
accuracy:	0.9861	accuracy:	0.9847	accuracy:	0.9889	accuracy:	0.9736

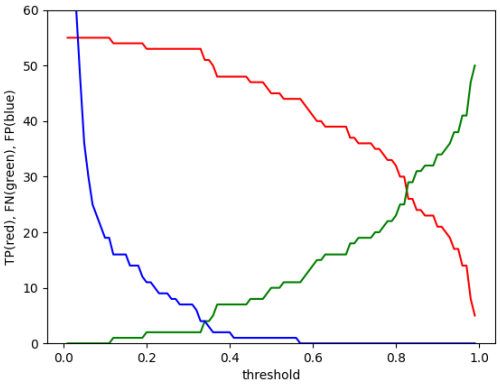
Thresholdy sa dajú voliť aj iné v závislosti od toho, ktorý typ chýb chceme minimalizovať. Na Obr. 1 je znázornené, ako sa menia kvantity TP, FN, FP v závislosti od thresholdu. V grafe (d) MLP regresor používame ako threshold priamo veľkosť prasknutej plochy, teda ukazujeme ako sa menia počty TP, FN, FP ak by sme za úspešné považovali modely zemetrasení veľkosťou prasknutej plochy aspoň threshold namiesto 95% maxima (tj. obsahu celej elipsy).

Skúmaním rôznych techník strojového učenia sme vyvodili nasledovné závery:

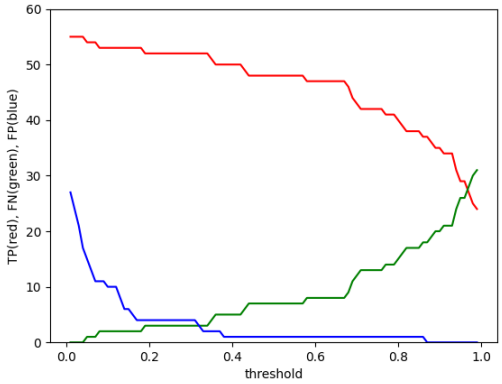
1. Metódy strojového učenia sa dajú použiť na rýchlu klasifikáciu veľkosti prasknutej plochy zemetrasení či na predpovedanie jej veľkosti z malého počtu parametrov.
2. Stačí používať malé sady modelov na dosiahnutie relatívne vysokej presnosti.



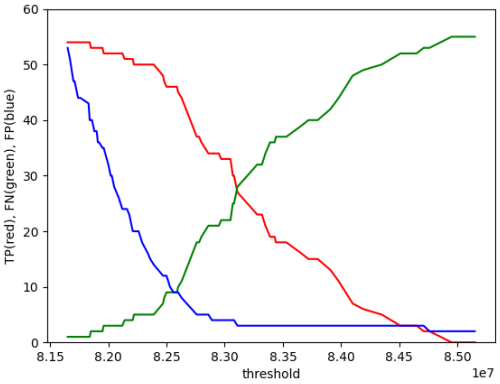
(a) MLP Classfier



(b) Random forest Classifier



(c) Gradient Boosting Classifier



(d) MLP regressor

Obr. 1: Porovnanie rôznych thresholdov.

5 Zdrojový kód

Všetok zdrojový python kód tohto projektu sa nachádza v nasledovnom github repozitári:
`https://github.com/Qwedux/grant/blob/main/code.py`

6 Zdroje

[1] https://scikit-learn.org/stable
[2] Ahamed, S., and E. G. Daub (2021). Application of machine learning techniques to predict rupture propagation and arrest in 2-D dynamic earthquake simulations, Geophys. J. Int. 224, 1918–1929
[3] Gallovič, F., Valentová, Ľ., Ampuero, J.-P., Gabriel, A.-A. (2019). Bayesian Dynamic Finite-Fault Inversion: 1. Method and Synthetic Test, J. Geophys. Res. Solid Earth 124, 6949-6969.
[4] Premus, J., Gallovič, F., Hanyk, L., Gabriel, A.-A. (2020). FD3D_TSN: Fast and simple code for dynamic rupture simulations with GPU acceleration, Seism. Res. Lett. 91, 2881-2889.