Aftershock Pattern Prediction Based On Earthquake Rupture Data For Improved Seismic Hazard Assessment

Rishabh Anand (19bcs4525)
AIT-CSE-IoT
Chandigarh University
Mohali, India
cse.19bcs4525@gmail.com

Shefali Yadav, 19BCS4524
AIT-CSE-IoT
Chandigarh University
Mohali, India
shefaliyd14@gmail.com

Abhishek Singh, 19BCS4508
AIT-CSE-IoT
Chandigarh University
Mohali, India
abhisheksinghxix@gmail.com

Abstract - In a world, divided by fear, of losing your loved ones, of losing your loved belongings,of losing your life, we hope to come up with a solution that should keep you and your dreams safe. Because that's what EarthQuake's take away... Even after the major tremor, what hurts more is the AfterShocks that follow. These are produced by the stress that was caused by the earthquake.

This project gives us a second chance at saving lives by using Artificial Intelligence to determine where the next tremor is going to be. So that you can move, and get to a safer place. Methods like Columnb's Stress Criterion are being used in current times to explain the spatial distributions of AfterShocks, but as the advent of science & technology is improving, we hope to introduce Machine Learning models that can find an undiscovered pattern which will be helpful in predicting the fair locations of AfterShocks.

Once we have our predictions, it is very important to display them in a good manner so that Uncle Bob can understand them and move himself to safety. We have created a React web-app just for this purpose so that it is easily acessible to people and move them from harm's way. Thereby, reducing the damage to both people and resources, thus, making this world a better place.

***Index Terms*—Bayesian inference, EarthQuake, Disster management, Aftershocks**

I. Introduction

The present project aims at forecasting the location of aftershocks that follow the occurrence of any large earthquake. While mainshocks are not (yet) predictable, aftershock statistics are sound enough for us to try predict them in reasonably small magnitude-space-time windows. For any mainshock of magnitude M occurring, Båth's law combined to the Gutenberg-Richter law tells us that there is a c. 10% chance that another earthquake (i.e. an aftershock) of the same magnitude or greater will occur (Båth, 1965; Gutenberg & Richter, 1944). Aftershocks are also most likely to occur just after the mainshock with their likelihood decreasing with time (Mignan, 2015 and references therein). Predicting where they are most likely to occur represents an important endeavour to improve time-dependent seismic hazard and risk assessment (e.g. Mignan et al., 2018). So far, the preferred model to predict aftershock spatial patterns is the Coulomb stress model (e.g. King et al., 1994). Recent results (De Vries et al., 2018) as well as the present project show that machine learning (ML) can significantly improve those predictions.

Our work is based on the recent article 'Deep learning of aftershock patterns following large earthquakes' by Phoebe M. R. DeVries, Fernanda Viégas, Martin Wattenberg & Brendan J. Meade, published in Nature in 2018 (https://www.nature.com/articles/s41586-018-0438-y). We will refer to that study as DeVries18 hereafter. To the best of our knowledge, this is the first study to use ML (here deep learning) to predict the spatial patterns of aftershocks. As such, DeVries18 represents a milestone in earthquake ML research. It also provides us a framework upon which we can test new models and develop new hypotheses and products.

Our results can be summarized as follows:

- The DeVries18 results are first reproduced (AUC = 85%). This model (deep neural network (DNN) 12 - 50 - 50 - 50 - 50 - 50 -50 - 1) is then used as baseline;
- Since DeVries only used 12 stress-related features, we test simpler DNN topologies. A new DNN 12 - 8 - 8 - 1 yields a similar performance (AUC = 85%);
- Since (1) stress computation (i.e. mapping of deformation to stress via geometric operations; King et al., 1994) is computationally intensive and cumbersome, and since (2) neural networks can be interpreted as geometric operations themselves, we bypass stress computation entirely and define features based on mainshock rupture geometry and kinematics instead;
- We find that a DNN (2 - 6 - 6 - 1) with only two features (minimum distance to rupture and mean rupture slip) yields a similar performance (AUC = 85%);
- Since aftershock patterns are in fact relatively simple at first order (Mignan, 2018), an artificial neural network (ANN, 2 - 30 - 1), based on the universal approximation theorem, yields a similar performance (AUC = 85
- This is an example of Occam's Razor in action. In view of the AI/earthquake-prediction buzz observed in the media (see inset below), one could have imagined that afterhosks represent a complex problem that only artificial intelligence can solve. Although ML helps us better predicting aftershock patterns, there is nothing magical about it. No deep learning was in fact needed with aftershock patterns already well described by only 2 features, which is in agreement with any visual inspection of the aftershock data.

The impact of these new results is twofold:

- By bypassing stress computation entirely, using a simple ANN topology, and only 2 mainshock-rupture-based features, we can streamline the process of aftershock pattern prediction for future post-mainshock crisis management (i.e. process faster and more transparent);
- In our in-depth analysis, we were not able to obtain performances higher than AUC = 86% despite using stress, or geometry and kinematics, and this for various hyperparameterizations. This infers that important features are missing. Observing past aftershock patterns, we suggest that performance could be further improved if the location of nearby faults were included. Testing this hypothesis is however out of the scope of the present project.

II. Proposed Approach

A. Probelm Definition

The present project aims at forecasting the location of aftershocks that follow the occurrence of any large earthquake. While main shocks are not (yet) predictable, aftershock statistics are sound enough for us to predict them in reasonably small magnitude-space- time windows.

Here we use a deep-learning approach to identify a static-stress-based criterion that forecasts aftershock locations without prior assumptions about fault orientation. We show that a neural network trained on more than 131,000 main shock–aftershock pairs can predict the locations of aftershocks in an independent test dataset of more than 30,000 main shock–aftershock pairs more accurately (area under curve of 0.849) than can classic Coulomb failure stress change (area under curve of 0.583). We find that the learned aftershock pattern is physically interpretable: the maximum change in shear stress, the von Mises yield criterion (a scaled version of the second invariant of the deviatoric stress-change tensor) and the sum of the absolute values of the independent components of the stress-change tensor each explain more than 98 per cent of the variance in the neural-network prediction. This machine-learning-driven insight provides improved forecasts of aftershock locations and identifies physical quantities that may control earthquake triggering during the most active part of the seismic cycle.

B. Existing System

1) Coulomb failure stress change :

The Coulomb failure stress (CFS) criterion is the most commonly used method for predicting spatial distributions of aftershocks following large earthquakes. However, large uncertainties are always associated with the calculation of Coulomb stress change. The uncertainties mainly arise due to nonunique slip inversions and unknown receiver faults; especially for the latter, results are highly dependent on the choice of the assumed receiver mechanism. Based on binary tests (aftershocks yes/no), recent studies suggest that alternative stress quantities, a distance-slip probabilistic model as well as deep neural network (DNN) approaches, all are superior to CFS with predefined receiver mechanism. To challenge this conclusion, which might have large implications, we use 289 slip inversions from SRCMOD database to calculate more realistic CFS values for a layered half-space and variable receiver mechanisms. We also analyze the effect of the magnitude cutoff, grid size variation, and aftershock duration to verify the use of receiver operating characteristic (ROC) analysis for the ranking of stress metrics. The observations suggest that introducing a layered half-space does not improve the stress maps and ROC curves. However, results significantly improve for larger aftershocks and shorter time periods but without changing the ranking. We also go beyond binary testing and apply alternative statistics to test the ability to estimate aftershock numbers, which confirm that simple stress metrics perform better than the classic Coulomb failure stress calculations and are also better than the distance-slip probabilistic model.

Despite its frequent application for several decades, Coulomb failure stress calculations have been questioned by recent studies and shown to be outperformed by other stress scalars and state-of-the-art methods like deep neural network in forecasting aftershocks. However, the recent results are also questionable because of an artificial DNN application (Mignan & Broccardo, 2019) as well as simplified CFS calculations. As this has broad implication for this research area, we performed a comprehensive reanalysis of the previous ROC-based study. Here we include CFS metrics accounting for the variability of aftershock mechanisms and additionally taking account of the incompleteness of catalogs as well as the occurrence of background activity. In addition to the previously conducted ROC analysis for binary forecasts, we also tested forecasts of aftershock numbers. To summarize, we find that the results of the ROC analysis are dependent on the magnitude cutoff, aftershock duration, and grid sizes and that more realistic CFS calculations (OOP and VM) can significantly improve the results. However, our analysis verifies that the stress scalars MS and VMS, and distance-slip probabilistic model (R), all of which do not rely on any specification of receiver mechanisms, outperform on average the CFS metrics in all test setups. While CFS might still be used for the evaluation of the stress changes on well-defined fault segments, our results indicate that spatial forecasts of the aftershock density might be generally improved by using von-Mises stress (VMS) instead of Coulomb stress.

2) Research by Harvad and Google

They started with a databse of information on more than 118 major earthquakes from around the world.From there, they applied a neural network to analyze the relationships between static stress changes caused by the mainshocks and aftershock locations. The algorithm was able to identify useful patterns.

The end result was an improved model to forecast aftershock locations and while this system is still imprecise, it’s a motivating step forward. Machine learning-based forecasts may one day help deploy emergency services and inform evacuation plans for areas at risk of an aftershock.When they applied neural networks to the data set, they were able to look under the hood at the specific combinations of factors that it found important and useful for that forecast, rather than just taking the forecasted results at face value. This opens up new possibilities for finding potential physical theories that may allow us to better understand natural phenomena.

C. Problem Formulation

We have applied a neural net to analyze the relationships between static stress changes caused by the mainshocks and aftershock locations. The algorithm was able to identify useful patterns.

The end result was an improved model to forecast aftershock locations and while this system is still imprecise, it's a motivating step forward. Machine learning-based forecasts may one day help deploy emergency services and inform evacuation plans for areas at risk of an aftershock.

III. Approach

The following methodology will be followed to achieve the objectives defined for proposed research work:

- The idea is to observe a volume which extends 100 km horizontally and 50 km vertically from the main shock. We then break that volume into 5kmx 5km x 5km small volumes and calculate the elastic stress change tensors at each of their centroid.
- Now using that information we need to predict whether there was an aftershock in that small volume or not. In order to know the ground truth we use International Seismological Center(ISC) event catalogue, in which for each main shock we looked up for its corresponding aftershocks from 1 sec to 1 year time and using that information we created our ground truth i.e. whether there was an aftershock in that small volume or not.
- So this whole problem is now a binary classification problem, in which our neural network has to predict was there an aftershock in that small 5km x 5km x 5km region or not. The model will use deep learning techniques to predict whether there can be aftershock at a particular locations or not. I'll be taking the data provided by the SRCMOD http://equake-rc.info/SRCMOD/searchmodels/allevents/.
- The format of the data will be FSP (finite-source rupture model). We'll not be working on how to process those SRCMOD files in order to create a csv. Rather we'll use already created CSVs.
- The data provided by the SCRMOD file contains the information about the hypocenter, latitude, longitude, magnitude, strike, dip, the inversion parameters and many other relevant data that is sufficient to get an insight of the earthquake.
- We'll be feeding these data to the neural network having several hidden layers (it depends on you, try and experiment with it), which will then try to extract the important features in order to predict the whether there is a chance of having an aftershock or not. we'll be using two activation functions tanh and ReLu (again experiment with it). The output layer will be a sigmoid layer, which will give us a probability between 0 - 1, as to how confident it is.

Step By Step

1) Generating Data

First download all labelled samples per mainshock rupture model, in csv format, into AllCSV folder from the DeVries18 Google Drive fromgiven link. Read the template files nd save to a handy format that can be used later.

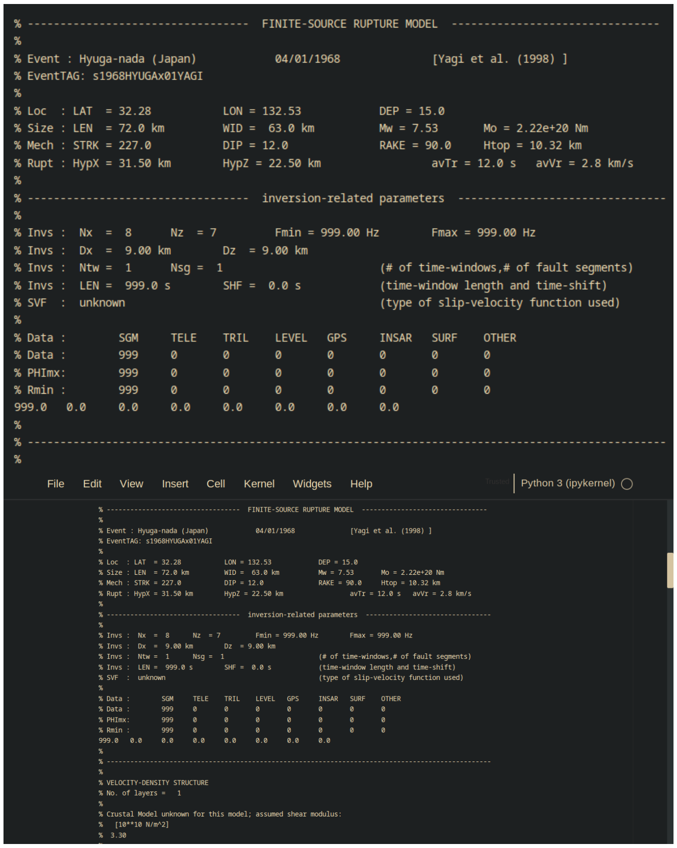
2) Initial Data Exploration

Initial data exploration for the project 'Aftershock pattern prediction based on earthquake rupture data for improved seismic hazard assessment'. DeVries18 will refer to the article 'Deep learning of aftershock patterns following large earthquakes' by Phoebe M. R. DeVries, Fernanda Viégas, Martin Wattenberg & Brendan J. Meade, and published in Nature in 2018 (https://www.nature.com/articles/s41586-018-0438-y). Data source : Finite-Source Rupture Model Database

Import the original mainshock rupture SRCMOD data files from the project's GitHub repository.

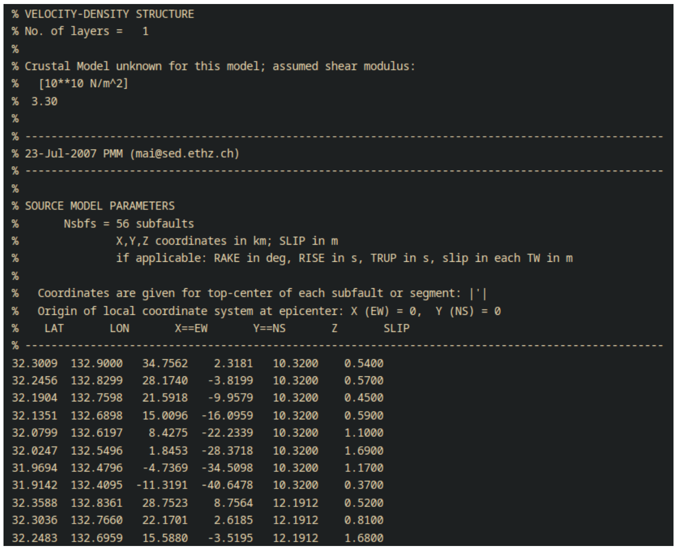
We first retrieved the same 199 SRCMOD.fsp files as used in DeVries18 from here

Sample FSP :



```
% ----------------------------------  FINITE-SOURCE RUPTURE MODEL  --------------------------------
%
% Event : Hyuga-nada (Japan)           04/01/1968             [Yagi et al. (1998) ]
% EventTAG: s1968HYUGAx01YAGI
%
% Loc  : LAT  = 32.28          LON = 132.53           DEP = 15.0
% Size : LEN  = 72.0 km        WID =  63.0 km         Mw = 7.53       Mo = 2.22e+20 Nm
% Mech : STRK = 227.0          DIP = 12.0             RAKE = 90.0     Htop = 10.32 km
% Rupt : HypX = 31.50 km       HypZ = 22.50 km                avTr = 12.0 s   avVr = 2.8 km/s
%
% ----------------------------------  inversion-related parameters  --------------------------------
%
% Invs :  Nx  =  8      Nz  = 7         Fmin = 999.00 Hz        Fmax = 999.00 Hz
% Invs :  Dx  =  9.00 km        Dz  = 9.00 km
% Invs :  Ntw =  1      Nsg =  1                        (# of time-windows,# of fault segments)
% Invs :  LEN =  999.0 s        SHF =  0.0 s            (time-window length and time-shift)
% SVF  :  unknown                                       (type of slip-velocity function used)
%
% Data :        SGM     TELE    TRIL    LEVEL   GPS     INSAR   SURF    OTHER
% Data :        999     0       0       0       0       0       0       0
% PHImx:        999     0       0       0       0       0       0       0
% Rmin :        999     0       0       0       0       0       0       0
999.0   0.0     0.0     0.0     0.0     0.0     0.0     0.0
%
% --------------------------------------------------------------------------------------------------
%
```

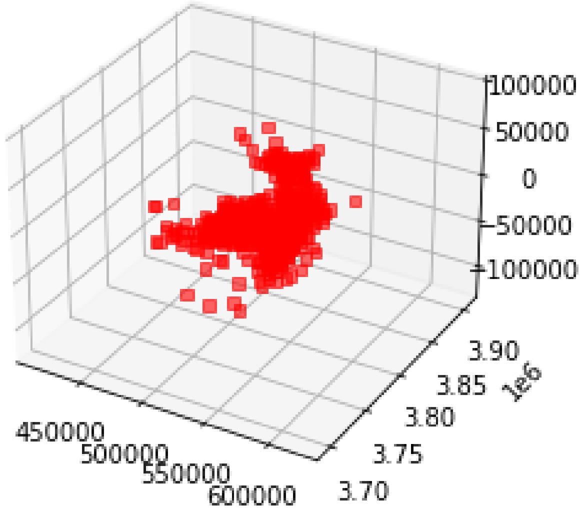
```
% ----------------------------------  FINITE-SOURCE RUPTURE MODEL  --------------------------------
%
% Event : Hyuga-nada (Japan)           04/01/1968             [Yagi et al. (1998) ]
% EventTAG: s1968HYUGAx01YAGI
%
% Loc  : LAT  = 32.28          LON = 132.53           DEP = 15.0
% Size : LEN  = 72.0 km        WID =  63.0 km         Mw = 7.53       Mo = 2.22e+20 Nm
% Mech : STRK = 227.0          DIP = 12.0             RAKE = 90.0     Htop = 10.32 km
% Rupt : HypX = 31.50 km       HypZ = 22.50 km                avTr = 12.0 s   avVr = 2.8 km/s
%
% ----------------------------------  inversion-related parameters  --------------------------------
%
% Invs :  Nx  =  8      Nz  = 7         Fmin = 999.00 Hz        Fmax = 999.00 Hz
% Invs :  Dx  =  9.00 km        Dz  = 9.00 km
% Invs :  Ntw =  1      Nsg =  1                        (# of time-windows,# of fault segments)
% Invs :  LEN =  999.0 s        SHF =  0.0 s            (time-window length and time-shift)
% SVF  :  unknown                                       (type of slip-velocity function used)
%
% Data :        SGM     TELE    TRIL    LEVEL   GPS     INSAR   SURF    OTHER
% Data :        999     0       0       0       0       0       0       0
% PHImx:        999     0       0       0       0       0       0       0
% Rmin :        999     0       0       0       0       0       0       0
999.0   0.0     0.0     0.0     0.0     0.0     0.0     0.0
%
% --------------------------------------------------------------------------------------------------
%
% VELOCITY-DENSITY STRUCTURE
% No. of layers =   1
%
% Crustal Model unknown for this model; assumed shear modulus:
%   [10**10 N/m^2]
%  3.30
```



```
% VELOCITY-DENSITY STRUCTURE
% No. of layers =   1
%
% Crustal Model unknown for this model; assumed shear modulus:
%   [10**10 N/m^2]
%  3.30
%
% ----------------------------------------------------------------------------------------------------
% 23-Jul-2007 PMM (mai@sed.ethz.ch)
% ----------------------------------------------------------------------------------------------------
%
% SOURCE MODEL PARAMETERS
%       Nsbfs = 56 subfaults
%               X,Y,Z coordinates in km; SLIP in m
%               if applicable: RAKE in deg, RISE in s, TRUP in s, slip in each TW in m
%
%   Coordinates are given for top-center of each subfault or segment: |'|
%   Origin of local coordinate system at epicenter: X (EW) = 0,  Y (NS) = 0
%    LAT       LON       X==EW       Y==NS       Z       SLIP
% ----------------------------------------------------------------------------------------------------
32.3009  132.9000   34.7562    2.3181   10.3200    0.5400
32.2456  132.8299   28.1740   -3.8199   10.3200    0.5700
32.1904  132.7598   21.5918   -9.9579   10.3200    0.4500
32.1351  132.6898   15.0096  -16.0959   10.3200    0.5900
32.0799  132.6197    8.4275  -22.2339   10.3200    1.1000
32.0247  132.5496    1.8453  -28.3718   10.3200    1.6900
31.9694  132.4796   -4.7369  -34.5098   10.3200    1.1700
31.9142  132.4095  -11.3191  -40.6478   10.3200    0.3700
32.3588  132.8361   28.7523    8.7564   12.1912    0.5200
32.3036  132.7660   22.1701    2.6185   12.1912    0.8100
32.2483  132.6959   15.5880   -3.5195   12.1912    1.6800
```

Visualize one labelled sample - let us consider the 1992 Landers case for one specific rupture model (event tag '1992LANDER01COHE').

We here want to reproduce an animation similar to the one shown in the this Google blog post to check the aftershock 3D pattern.



3) Extract-Transform-Load (ETL)

Extract::

- The data needed for this project were already extracted in the notebook $pred_seism_aftXYZ.data_exp$ for preliminary data exploration.
- Mainshock rupture files (SRCMOD) extracted from here, originally from here, then filtered manually, considering only the 199 mainshocks used in the DeVries18 publication
- Labelled data of the baseline model ($complete_dataset_labelled$) imported as pickle file, created from a list of csv files originally imported from the Google Drive
- The fsp format is quite complex and must be cleaned for feature engineering.

We here directly 'extract' all needed data from our working directory.

Transform ::

Fortunately, we can use the srcmod.py program which reads and transform fsp files to be (originally) used as input for Coulomb stress modelling. The original code is available on the following Google GitHub repository.

Although we won't do any Coulomb stress modelling, using the same input format will simplify feature engineering in the next step of the process model. Note that the original code was done with Python 2. For Python 3, we made the following modifications:

- $printx$ changed to $print(x)$,
- $.has_key$ changed to in
- We also removed the gcs module with $gcs.File(filename)$ changed to $open(filename,'r')$.
- Changed proj utm source.

The updated code srcmod.py is available in this project's GitHub repository under utils.

The original srcmod reader was a little fragile, failing to read in many of the .mat files. ReadSrcmodFile fixes this – it reads in the raw ascii from a .fsp file, and returns the data in the same format. There are some small differences in the data, mostly due to the fact that .mat files are 64-bit, and we read in the ascii, and covert to floats. As such, there's some small differences in the read in values, but this routine does work with all the data.

Far more manageable than the fsp file format, we will use defaultdict objects in the next step of the process model to define new features based on the SRCMOD rupture parameters (geometry and kinematics). The structure of the dictionary created by ReadSrcmodFile is first further simplified to use cell centers instead of 4 corners per cell.

No transformation needed for the labelled dataset of the baseline model, thus , we will directly use the dataframe $df_DeVries18$ for feature engineering.

Load::

Two objects will be used for feature definition:

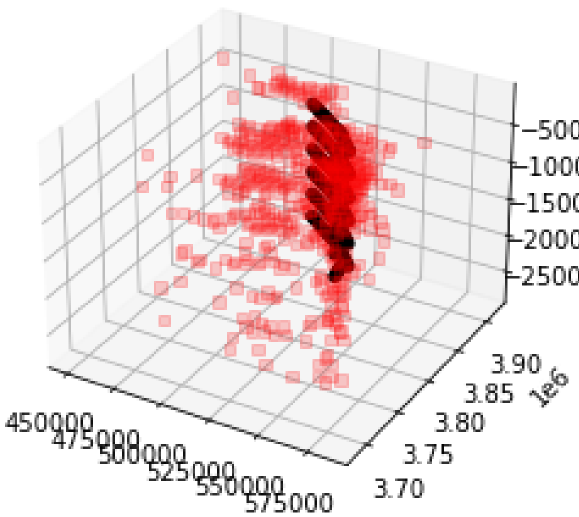
- $SRCMOD_dictList$, a list of dictionaries, which contains all required rupture parameters per mainshock 'ID' (saved as $SRCMOD_cleaned.pkl$);
- $df_DeVries18$, a dataframe, which contains the labelled dataset of the baseline model and where mainshocks are identified by 'ID' (already saved as $complete_dataset_labelled.pkl$).

4) Feature Engineering

- Make balanced dataset
- Unbalanced towards 0 values
- Subsample 0s - we use the same approach as in DeVries18 for consistency : Subsampling here reduces the number of samples from 6+ million to c. 200,000
- Merge 1s and 0s

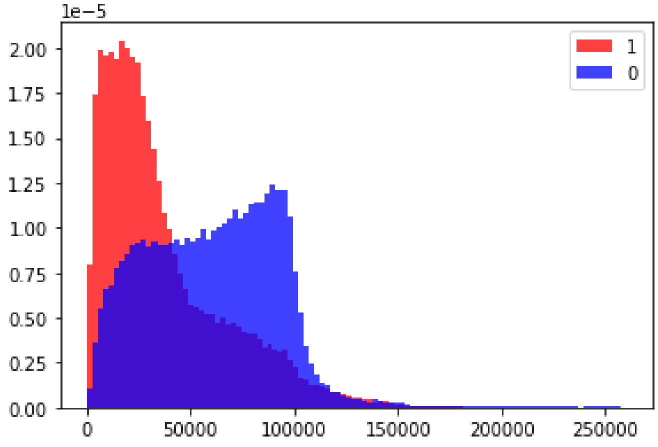
We do the same engineering of stress features as done in DeVries18, spliting the 6 stress components into 12 $(abs(stress) and - abs(stress))$ and normalizing the new features by 1e6.

Visualizes the geometric relationship between SRCMOD rupture and aftershock pattern. As example, let us consider the 1992 Landers case for one specific rupture model (event tag ’1992LANDER01COHE’). We here want to reproduce an animation similar to the one shown in the following Google blog post.

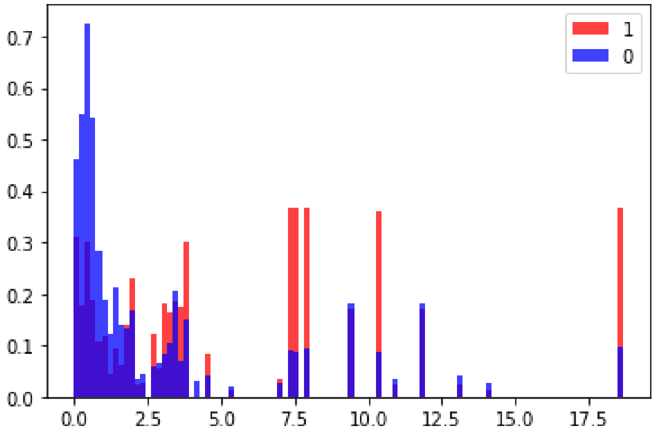


The simplest choice is to define as feature the nearest distance between aftershock cells and rupture (see animation above). Since DeVries18 found the absolute values of stress components to lead to AUC = 85%, we must identify some rupture parameters that will lead our ML algorithm to approximate, in part, those stress operations. Since stress depends on deformation, mainshock rupture slip should be a feature.

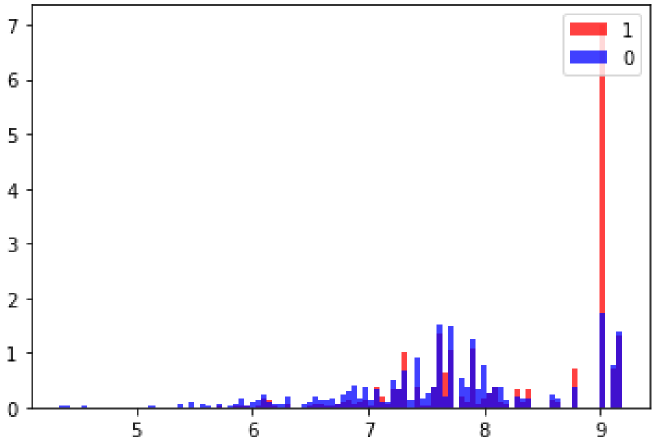
Various features are defined in the next step of the process model.



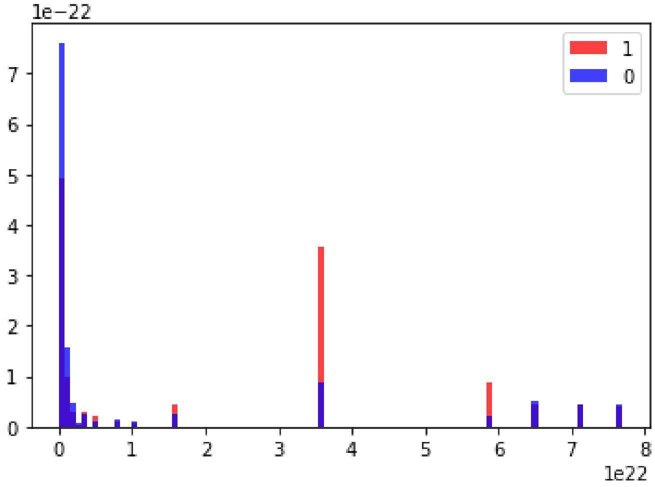
Mindist aftershock rupture



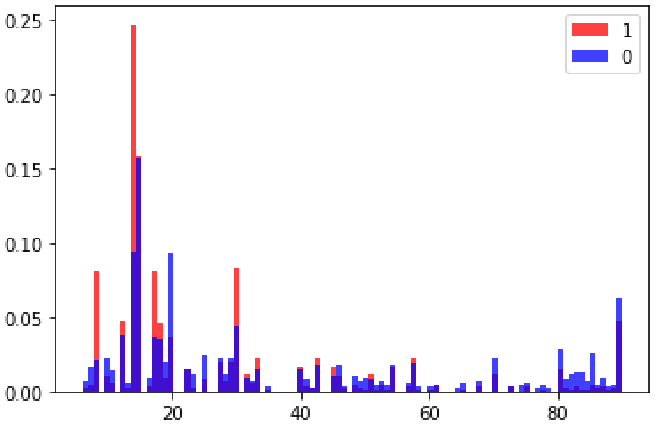
Slip-mean aftershock rupture



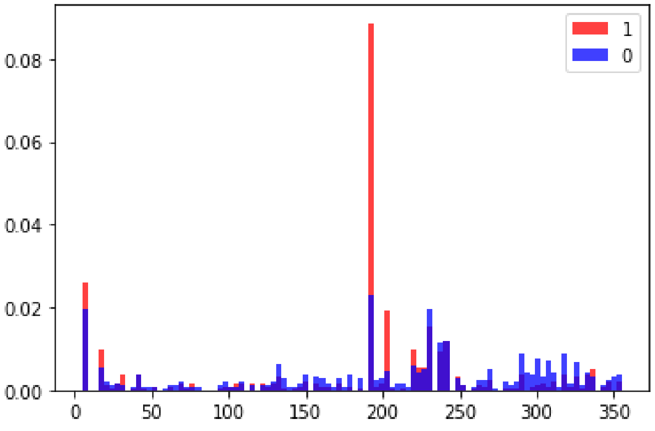
Magnitude aftershock rupture



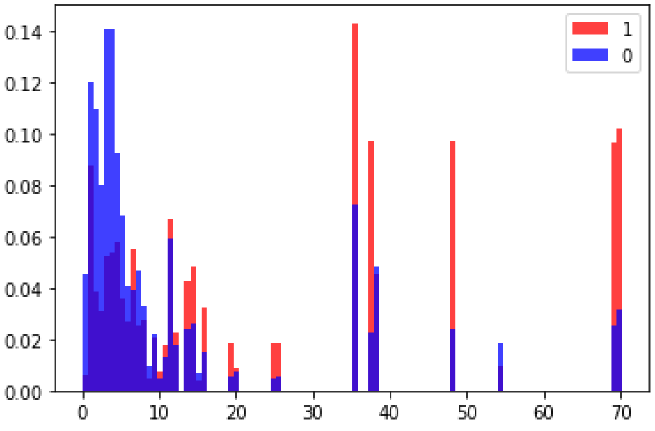
Moment aftershock rupture



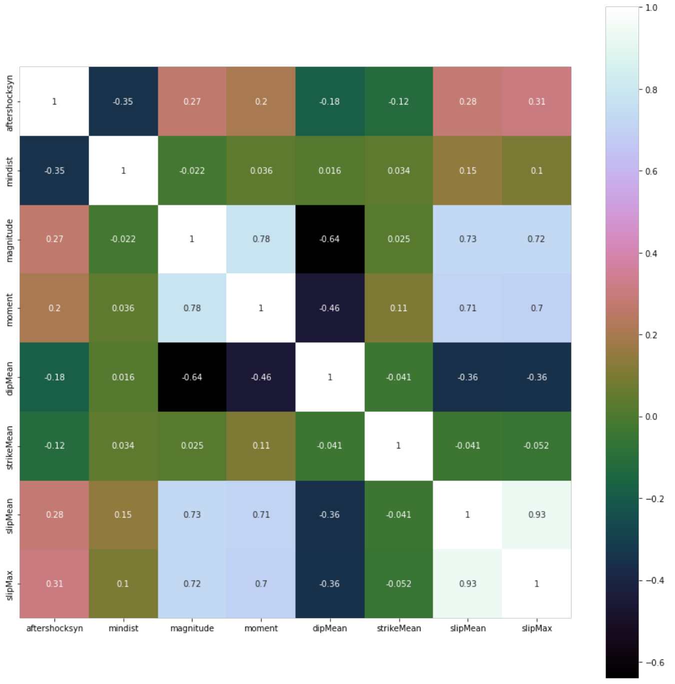
Dip-Mean aftershock rupture



Strike-Mean aftershock rupture

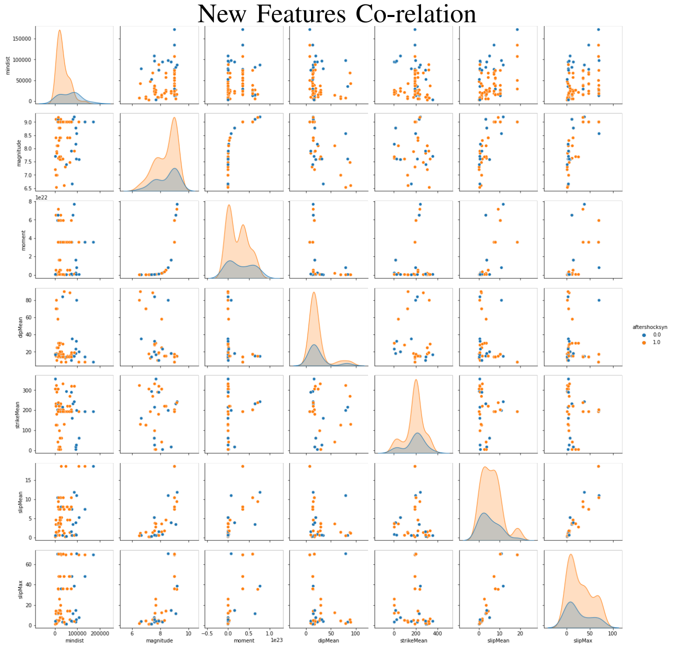


Slip-Max aftershock rupture

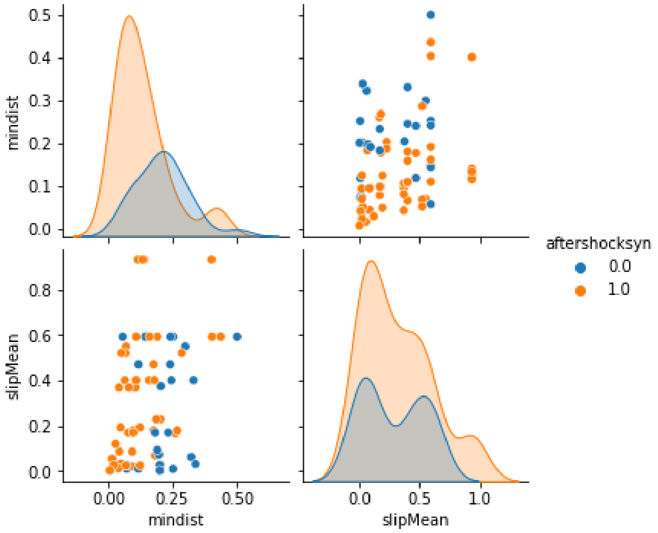


	aftershocksyn	mindist	magnitude	moment	dipMean	strikeMean	slipMean	slipMax
aftershocksyn	1	-0.35	0.27	0.2	-0.18	-0.12	0.28	0.31
mindist	-0.35	1	-0.022	0.036	0.016	0.034	0.15	0.1
magnitude	0.27	-0.022	1	0.78	-0.64	0.025	0.73	0.72
moment	0.2	0.036	0.78	1	-0.46	0.11	0.71	0.7
dipMean	-0.18	0.016	-0.64	-0.46	1	-0.041	-0.36	-0.36
strikeMean	-0.12	0.034	0.025	0.11	-0.041	1	-0.041	-0.052
slipMean	0.28	0.15	0.73	0.71	-0.36	-0.041	1	0.93
slipMax	0.31	0.1	0.72	0.7	-0.36	-0.052	0.93	1

New Features Co-relation



New features Sample



5) *Model Definition*

- We here reproduce the DeVries18 deep learning model = six hidden layers with 50 neurons each and hyperbolic tangent activation functions (13,451 weights and biases in total).
- Since the DeVries model has a 12-neuron input layer, we first verify whether a much simpler Deep Neural Network (DNN) topology can lead to similar results. We also reduce the dropout rate. We keep everything else the same (i.e. $kernel_initializer, activation, optimizer, loss$). This model is for illustration purposes only. A different model with different features is proposed in this project.
- We use more direct features than in DeVries18, here based on geometry/kinematics instead of stress. We further simplify the topology of the DNN compared to the baseline.
- We also test a shallow Artificial Neural Network (ANN) (i.e. no deep learning, only one hidden layer).
- NB: During training, we will also test another standard machine learning algorithm (XGBoost) for completeness.

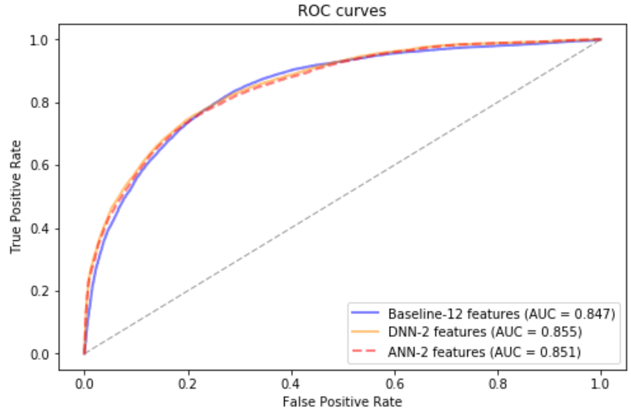
6) *Model Training*

This is the baseline model CV results - we should aim at achieving similar or greater accuracy (72%) for the new model with new set of features. This will however not assure a same or better generalization than the baseline model (AUC = 85%, see next step of the process model). Since the Devries18 model uses a relatively small input layer (12 nodes), a simpler DNN topology should do just fine. We show it, only for illustrative purpose. The model proposed in this project will use another set of features.

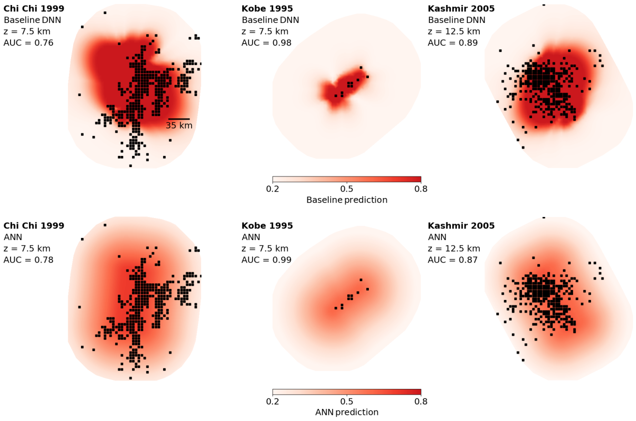
Training new models with the proposed set of features based on geometry and kinematics (minimum distance to mainshock rupture and mean slip on rupture). To not only test neural networks, we here test another standard machine learning algorithm. XGBoost has become a method of choice in recent Kaggle competitions (althought this does not mean that it will perform better than neural networks in the present case - ”No Free Lunch” theorem)

IV. Model Evaluation

We reproduced the DeVries18 result (baseline AUC = 85%). Similar results are obtained for a simpler DNN topology (12 features) but also for simpler features (2 parameters based on geometry and kinematics). Note that the values may slightly change for different training runs. We also tested the XGBoost classifier, which had shown good CV results during the training step. We here find that the model does not generalize well. For the rest of the project, only neural networks are considered. This does not mean that other ML algorithms or other XGBoost hyperparameterizations could not do better.



We finally produce plots similar to Fig. 1 of DeVries18 for a last check. Most of the plot formatting comes from PlotThreeTestCases.py of DeVries18.



V. References

- https://www.blog.google/technology/ai/forecasting-earthquake-aftershock-locations-ai-assisted-science/
- https://www.nature.com/articles/s41586-018-0438-y
- https://www.nature.com/articles/d41586-018-06091-z
- https://phys.org/news/2018-08-ai-quake-aftershocks.html
- Båth, M. Lateral inhomogeneities of the upper mantle. Tectonophysics 2, 483–514(1965).
- Utsu, T. A statistical study on the occurrence of aftershocks. Geophys. Mag. 30, 521–605 (1961).
- Parsons, T., Stein, R. S., Simpson, R. W. & Reasenberg, P. A. Stress sensitivity of fault seismicity: a comparison between limited-offset oblique and major strike-slip faults. J. Geophys. Res. 104, 20183–20202 (1999).