Project Title: Earthquake Induced Landslide Prediction Using CNN Model

Earthquake induced landslide prediction has been measured Convolutional Neural Network (CNN) supervised machine learning classification algorithm. Performance measures of these methods has been evaluated using three pillars of binary classification namely accuracy, precision and recall. It has been perceived that the deep learning model CNN (accuracy=0.981) performed well than Support Vector Machine SVM(accuracy=0.968) .

CNN perform well than other Neural Network architectures because of their unique process. CNNs grouping a number of pixels together so they can recognize a temporal pattern. Important steps involved in CNN models are Data pre-processing, creating a model and fitting the model. This model wants to know the input shape of accepted data. To this, the first layer in a sequential model receive information about its input shape. Different possible ways to do this are (i) Pass an input shape argument to the first layer (ii) for dense layer ,support the specification of their input shape via the argument input dimension (iii)specify a fixed batch size argument to layer. Here the dataset is CSV file which is a array. So firstly, we have to convert these into 3-D matrix of pixels image. Now re-shape the matrix into a 3-D matrix then separate the X(input) and Y(output) from data. Re-scaling the data in the range 0 and 1 ,which fasten the process.

This model is build with 2 Convolutional layer 2 Max pooling layer and 2 hidden layers. There is a flatten layer between the convolutional layer and the fully connected layer. The flatten mainly used to flatten the 3-D array of previous layer into a single layer, vector that can be fed into a fully connected neural network classifier. There are two labels so there will be 2 neurons in the output layer. There are 784 columns for pixel representation, which is reshaped as 28*28. Dropout function is also used to avoid over-fitting of model.

Model Creation, Compilation, Fitting and Evaluation.

The model creates a Neural Network that has three layers. There are two layers of 16 nodes each and one output node. The last node uses the sigmoid activation function that will keeps all the values between 0 and 1 into the form of a sigmoid curve. The other two layers use ReLU (Rectified Linear Units) as the activation function. ReLU is a half rectified function; that is, for all the inputs less than 0 the value is 0 while for anything positive the value is retained. One output unit is used since for each record values in X, a probability will be predicted. The outputs represent the *probability* of either outcome 0 or outcome 1. So the decision boundary is 0.5. if prediction > 0.5 , the prediction is 1. if prediction <= 0.5 , the prediction in 0.

The compiled the network uses stochastic gradient descent optimizer(sgd). It helps to find the minima, control the variance, then update the model's parameters and finally lead us to Convergence.

The loss function used in this model is binary cross entropy. For binary classification problems that give output in the form of probability, binary cross entropy is usually the optimizer of choice. Metrics used in this model is is accuracy. The model is trained for 50 epochs with a batch size of 100. Finally, the trained model was evaluated for the test set to check the accuracy.