

BTECH PROJECT 2020

PREDICTING SEISMIC HAZARD IN AN UNDERGROUND MINE USING MACHINE LEARNING AND DEEP LEARNING ALGORITHMS

Mentored by:
Dr. Nawal Kishore Sir

Performed By:
Nikhil Kumar





Outline

INTRODUCTION

OBJECTIVE

DATASET AND DESCRIPTION

DATA PREPROCESSING

OVERSAMPLING

FEATURE SELECTION

FEATURE CREATION AND SCALING

DATA DIVISION

CROSS VALIDATION

HYPERPARAMETER TUNING

LEARNING MODELS

EVALUATION

SOFTWARE AND FRAMEWORKS USED

CONCLUSION

INTRODUCTION

What is Machine Learning?

Application of AI which provides systems an ability to automatically learn and improve from experience without being explicitly programmed.

Mining activity was and is always connected with the occurrence of dangers which are commonly called mining hazards. A special case of such threat is a seismic hazard which frequently occurs in many underground mines.

Seismic hazard is the hardest detectable and predictable of natural hazards and in this respect it is comparable to an earthquake.

OBJECTIVE

The objective of this use-case is to develop an algorithm that will forecast if there will be generation of hazardous seismic in future shifts.

Seismic hazard is the hardest detectable and predictable natural hazard as compared to other hazards like hurricane, cyclone etc. There are many advanced seismic and seismoacoustic monitoring systems that help us better understand the rock mass process and hazards but their accuracy is very low. Complexity of seismic processes and big differences between the number of low-energy and high energy (greater than 10000 joules) seismic events causes statistical techniques to be insufficient to predict seismic hazard.

DATASET AND DESCRIPTION

The dataset used for this project is Seismic-Bumps Dataset and it describes the problem of high energy (higher than 10^4 J) seismic bumps forecasting in a coal mine. Data come from two of longwalls located in a Polish coal mine. This data was generated by Silesian University of Technology, Poland and Institute of Innovative Technologies EMAG, Poland. The multivariate dataset here contains 19 different features and 2584 instances of this features. The available features are real and categorical forms.

	genergy	gpul	gdenergy	gdpuls	nbumps	nbumps2	nbumps3	nbumps4	nbumps5	nbumps6	nbumps7	nbumps89	energy	maxenergy	class
count	2.584000e+03	2584.000000	2584.000000	2584.000000	2584.000000	2584.000000	2584.000000	2584.000000	2584.000000	2584.0	2584.0	2584.0	2584.000000	2584.000000	2584.000000
mean	9.024252e+04	538.579334	12.375774	4.508901	0.859520	0.393576	0.392802	0.067724	0.004644	0.0	0.0	0.0	4975.270898	4278.850619	0.065789
std	2.292005e+05	562.652536	80.319051	63.166556	1.364616	0.783772	0.769710	0.279059	0.068001	0.0	0.0	0.0	20450.833222	19357.454882	0.247962
min	1.000000e+02	2.000000	-96.000000	-96.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.000000	0.000000	0.000000
25%	1.166000e+04	190.000000	-37.000000	-36.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.000000	0.000000	0.000000
50%	2.548500e+04	379.000000	-6.000000	-6.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.000000	0.000000	0.000000
75%	5.283250e+04	669.000000	38.000000	30.250000	1.000000	1.000000	1.000000	0.000000	0.000000	0.0	0.0	0.0	2600.000000	2000.000000	0.000000
max	2.595650e+06	4518.000000	1245.000000	838.000000	9.000000	8.000000	7.000000	3.000000	1.000000	0.0	0.0	0.0	40200.000000	40000.000000	1.000000

DATASET AND DESCRIPTION

FEATURES DESCRIPTION :

- **Shift:** information about type of a shift (W - coal-getting shift, N -preparation shift).
- **genergy:** seismic energy recorded within previous shift by the most active geophone (GMax) out of geophones monitoring the longwall.
- **gpuls:** a number of pulses recorded within previous shift by GMax.
- **Gdenergy:** a deviation of energy recorded within previous shift by GMax from average energy recorded during eight previous shifts.
- **nbumps:** the number of seismic bumps recorded within previous shift.
- **Energy:** total energy of seismic bumps registered within previous shift.
- **Maxenergy:** the maximum energy of the seismic bumps registered within previous shift.
- **Class:** the decision attribute - '1' means that high energy seismic bump occurred in the next shift('hazardous state'), '0' means that no high energy seismic bumps occurred in the next shift.

DATA PREPROCESSING

In this step the data gets transformed, or Encoded, to bring it to such a state that now the machine can easily parse it. In other words, the features of the data can now be easily interpreted by the algorithm.

- Replacing Null values
- Categorical data to Numerical data
- Dimensionality Reduction using PCA



DATA PRE-PROCESSING

Why is Data Pre-processing important?

Features which have less null values are imputed using interpolation or mean, median values to create a consistent model. Features are dropped if high number of values are missing.

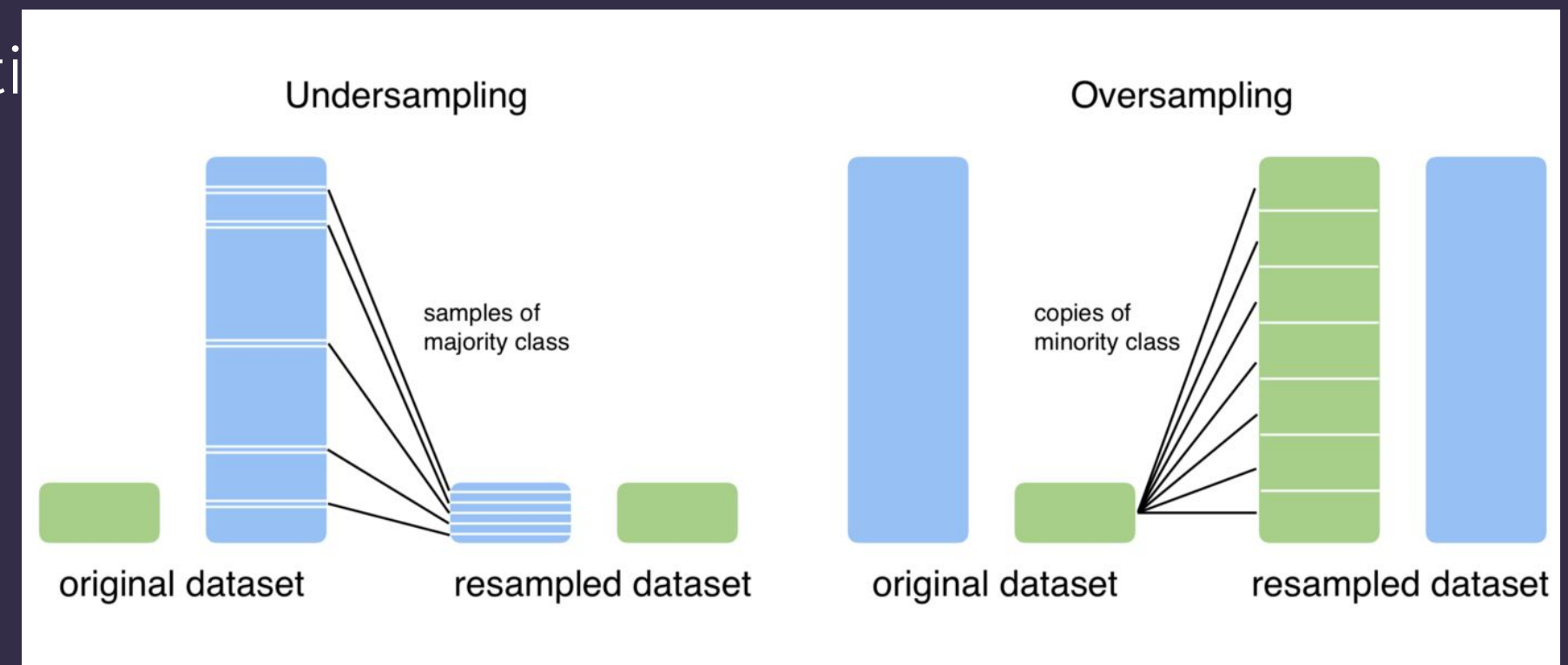
Categorical Data cannot be understood by computer so it is necessary to convert it to numerical form.

Data analysis tasks become significantly harder as the dimensionality of the data increases. As the dimensionality increases, the number planes occupied by the data increases thus the data becomes difficult to model and visualize.

UNDERSAMPLING AND OVERSAMPLING

An unbalanced dataset is one in which the target variable has more observations in one specific class than the others. Here our data is unbalanced since it contains 93% data as non-hazardous and 7% data as hazardous. This type of data pose problems like accuracy pradox. So we need to perform operations to balance the dataset.

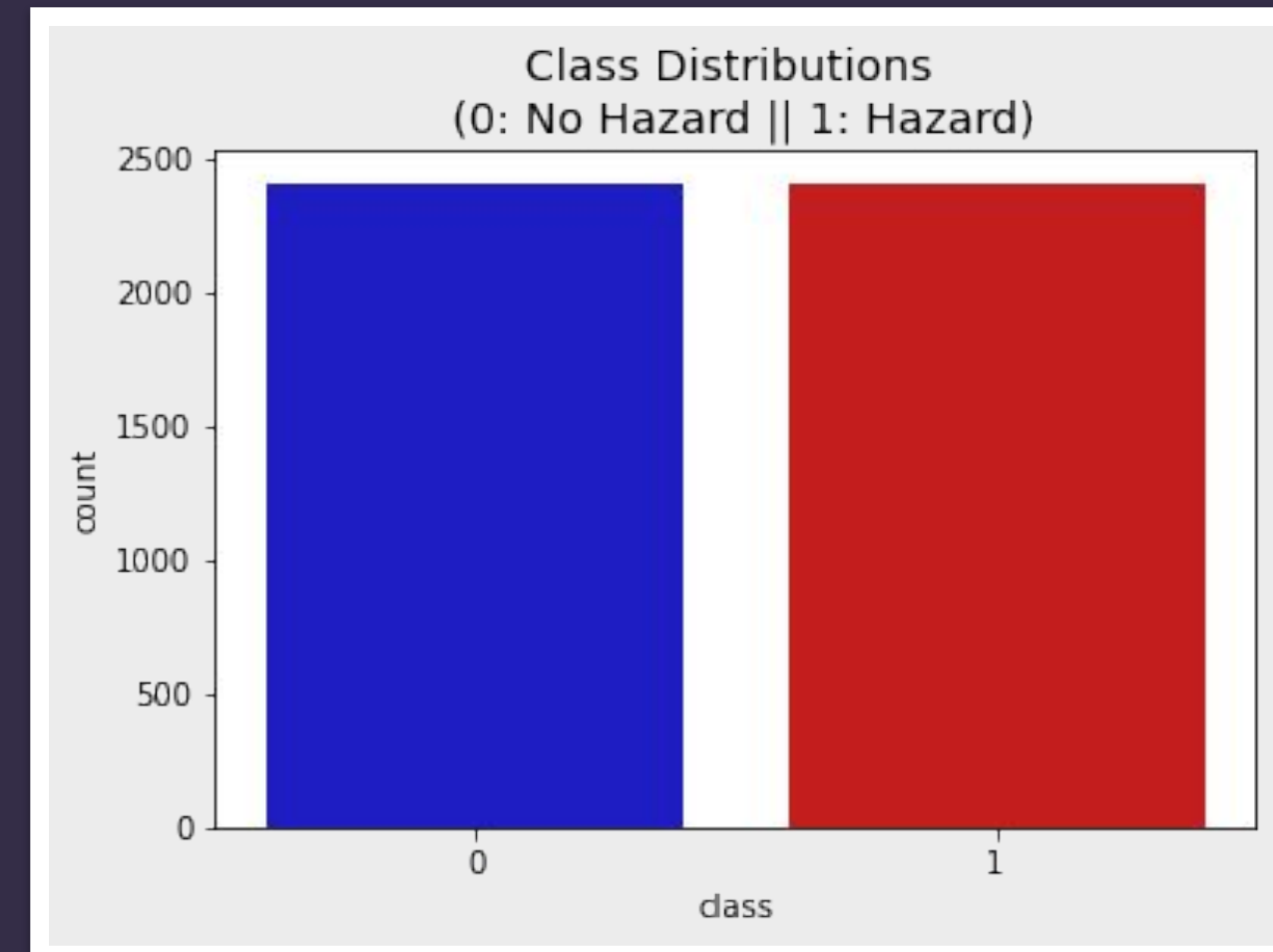
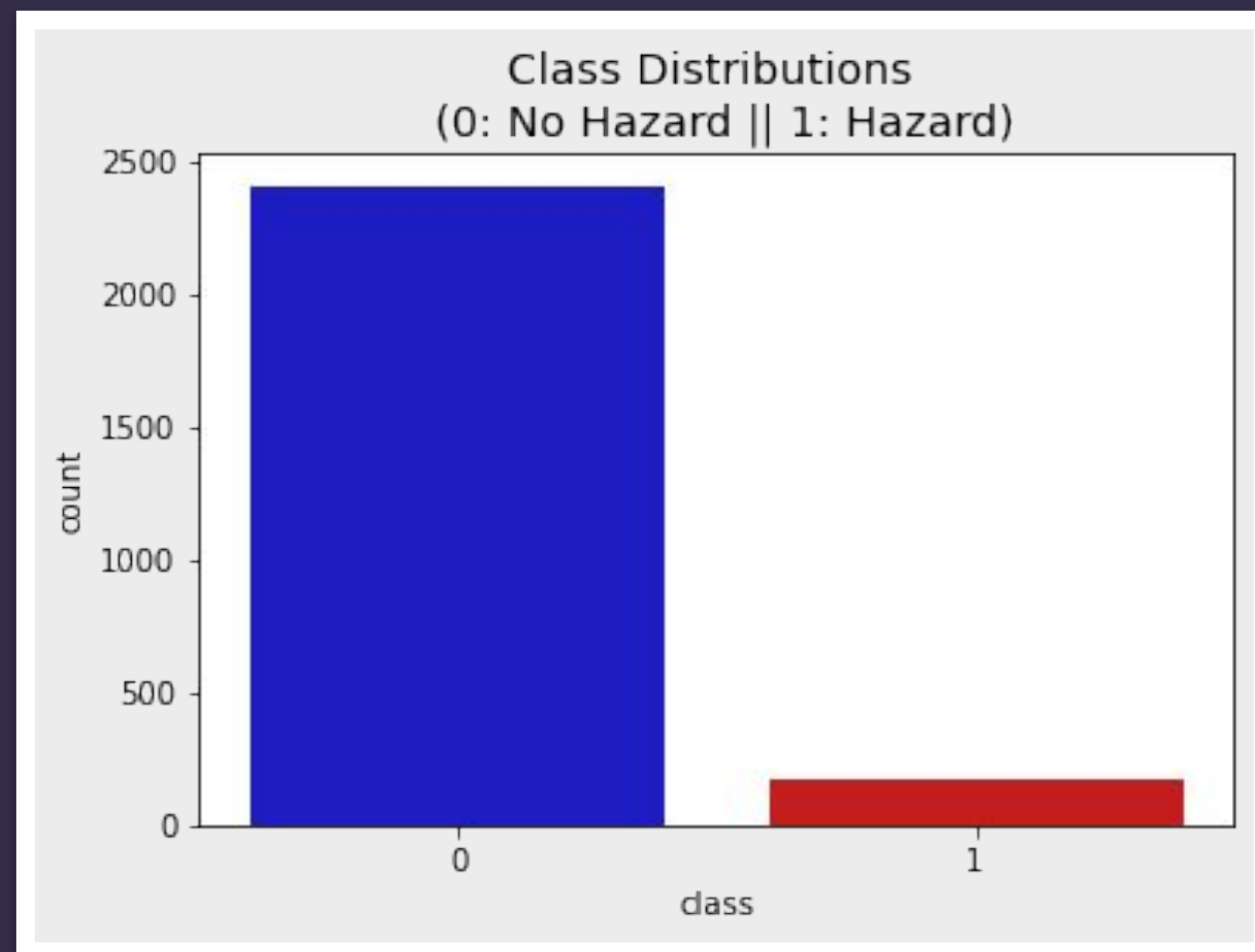
There are two basic methods to balance the data, undersampling and oversampling. This methods can be opti more real features .



UNDERSAMPLING AND OVERSAMPLING

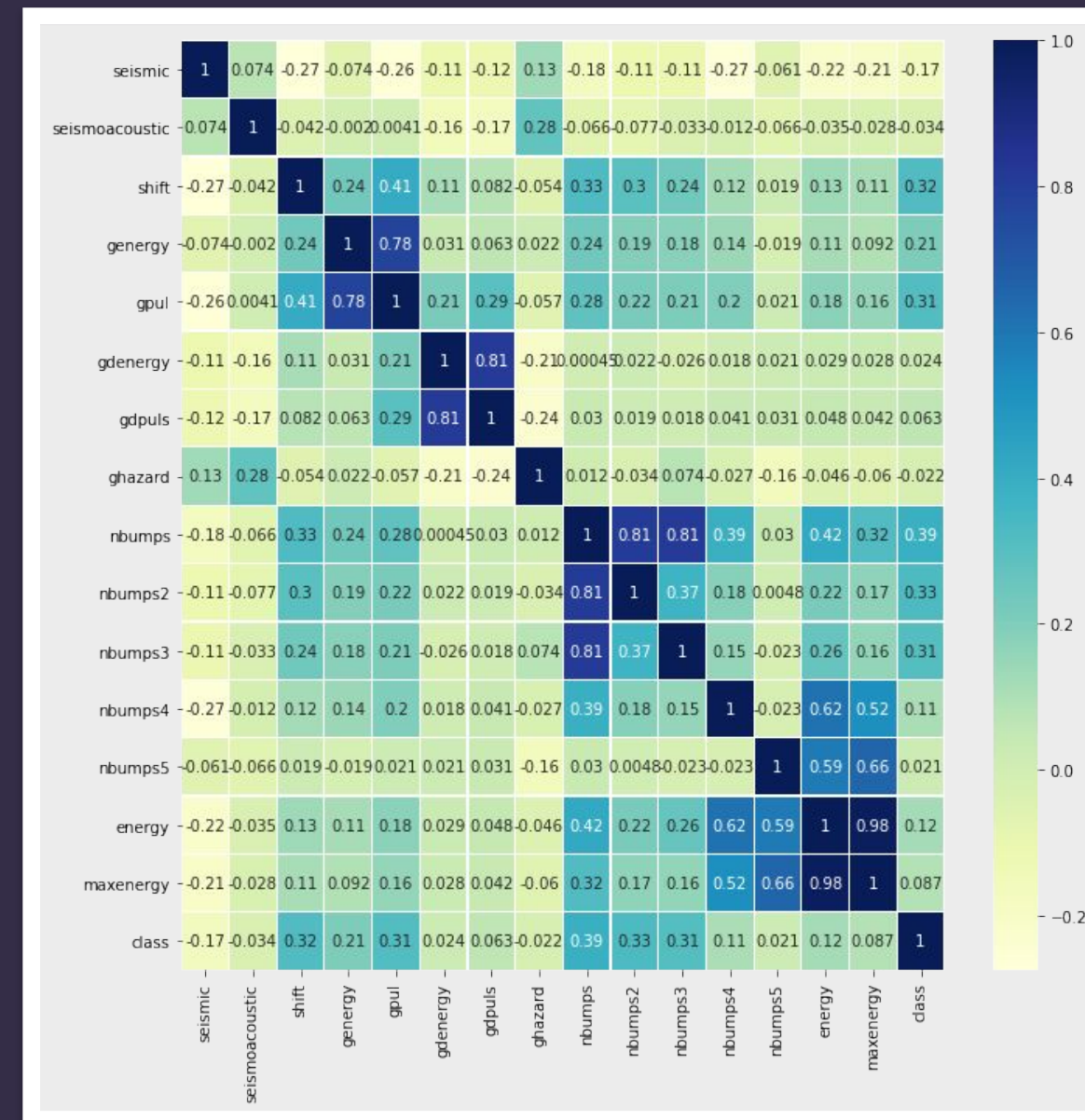
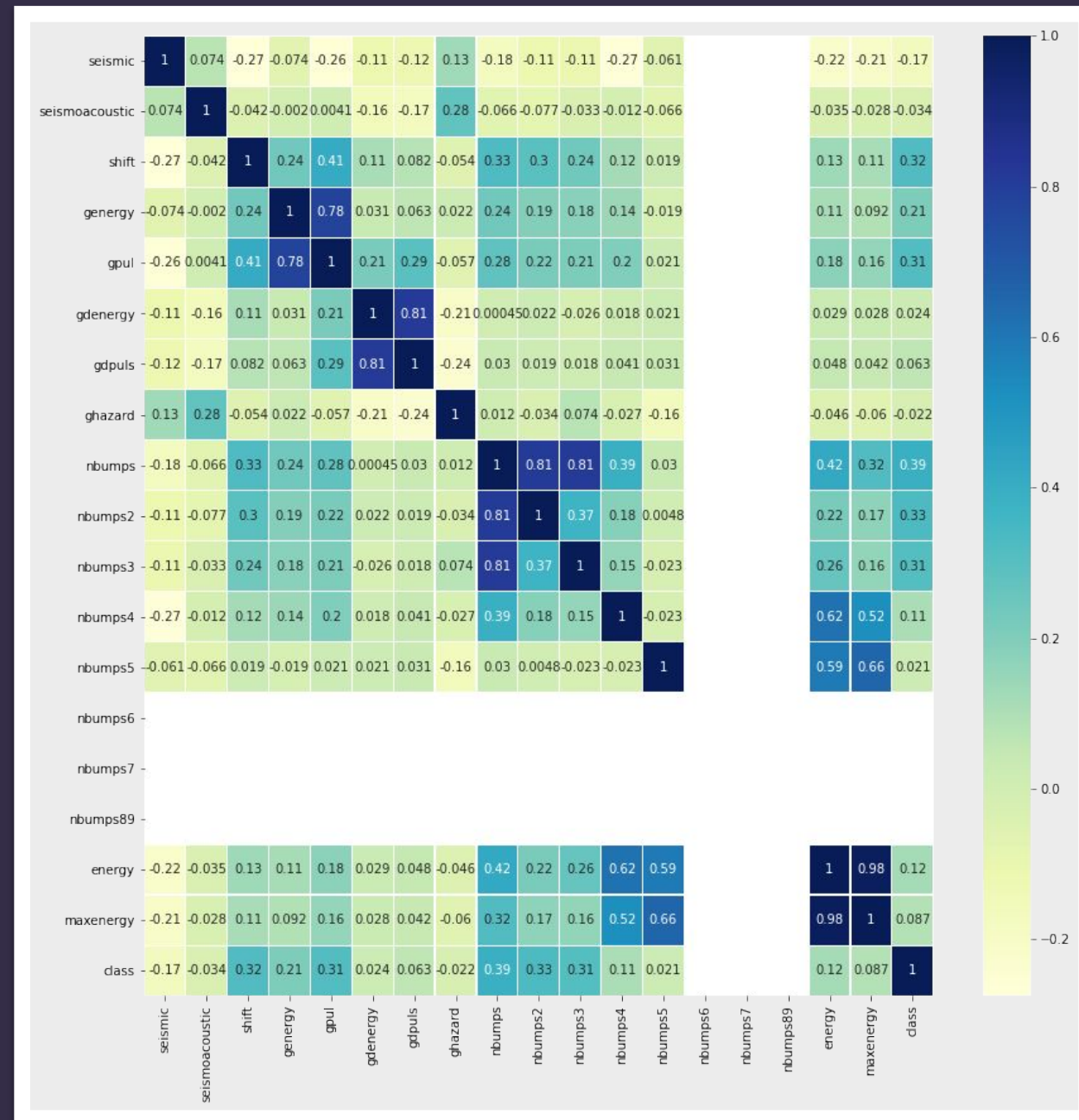
We oversample here to bring the ratio of class distribution to 50%.

We don't perform undersampling since we do not have enough minority class and there will be loss of high amount of data.



FEATURE SELECTION

In feature selection we select those features which contribute most to the target variable and remove those features which affects negatively to the model performance. Here we use correlation feature to find out the important features. Correlation function tells how two features are related with each other.



FEATURE SELECTION

Why is Data Feature selection important?

Feature selection hugely impacts the performance of the model. So the irrelevant features should be removed so that the performance of model is least affected.

- It helps in improving accuracy.
- It reduces overfitting because the model is not being trained on noisy data.
- It reduces model training time.

DATA DIVISION

The data we use here is split into training data and test data. The training set contains a known output and the model learns on this data in order to be generalized to other data later on. We have the test dataset (or subset) in order to test our model's prediction on this subset.

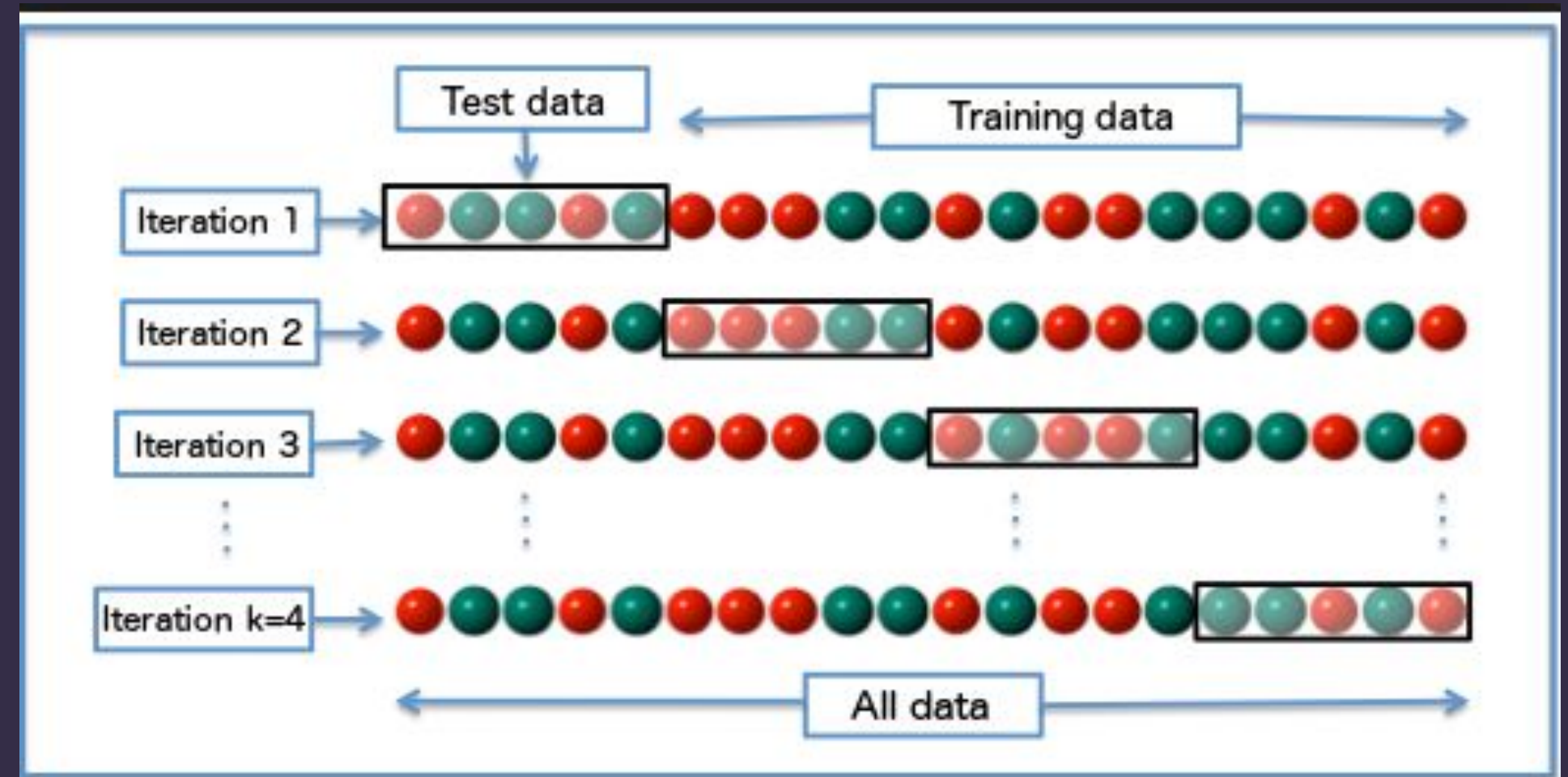
Here we split the data in 80/20 ratio. The training set contains 3862 instances and testing set contains 966 instances.

CROSS VALIDATION

We test the model on unseen data to see if the model overfits or underfits. Cross validation helps to train and test the model on entire data by splitting the data in k folds.

Why cross validation?

Cross-Validation helps us to test the effectiveness of our machine learning model. By finding the mean of all K folds we get the best accuracy for our model.



HYPERPARAMETER TUNING

Hyperparameter tuning is a method to choose the best parameters for any model for which the performance of the model is best.

The parameters like n_neighbours, max_depth, learning rate, epochs, hidden layers etc. govern the training process of any model and produce a significant impact on the accuracy of the model.

To tune the hyperparameters we used:

- Grid Search method

Grid Search method is a brute force method which trains, tests and validates the model for every combination of parameters which are passed and stores the parameters of best performing model.

Randomized search is similar to grid search method but instead of training for every combination it tests for random n combination and it helps to reduce tuning time.

LEARNING MODELS

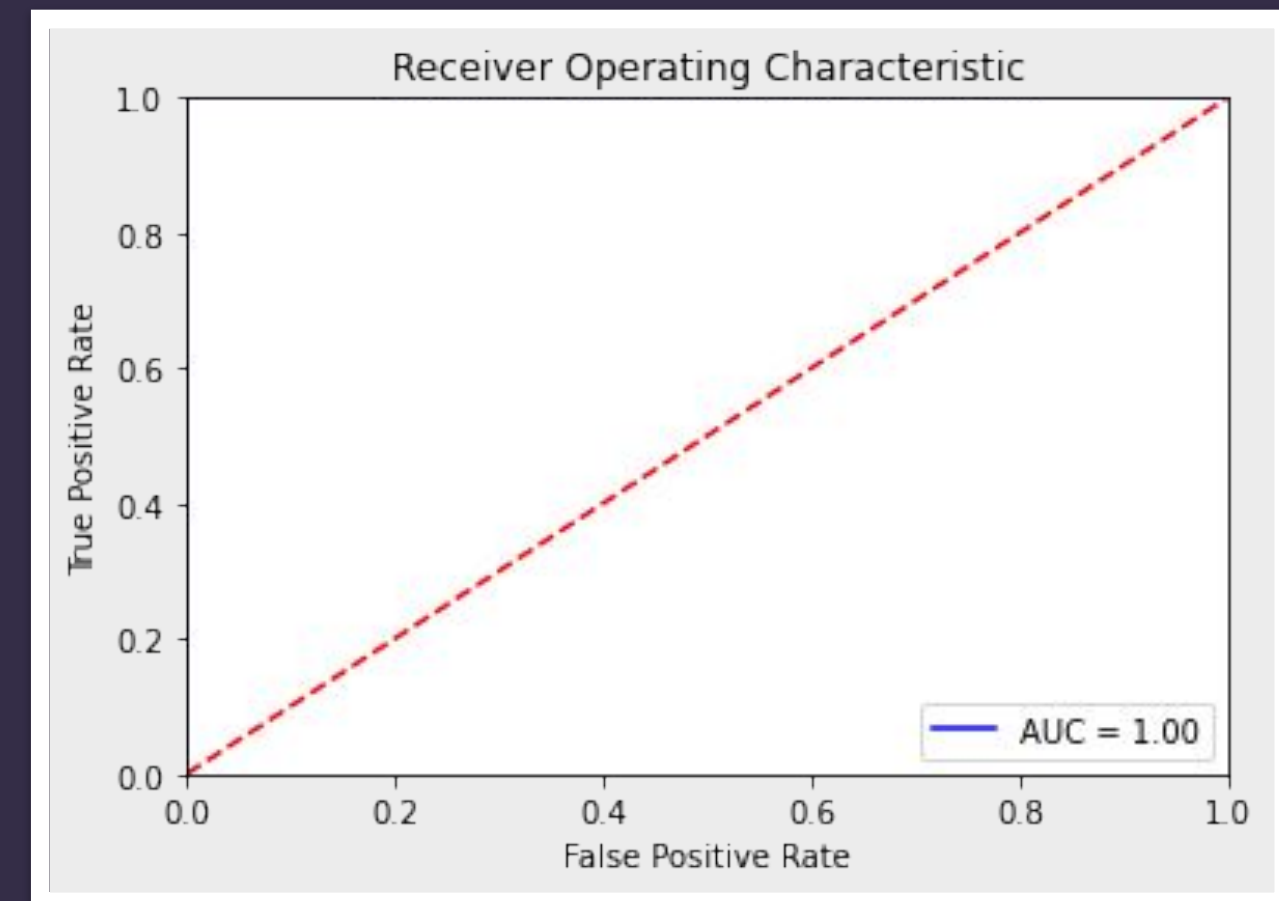
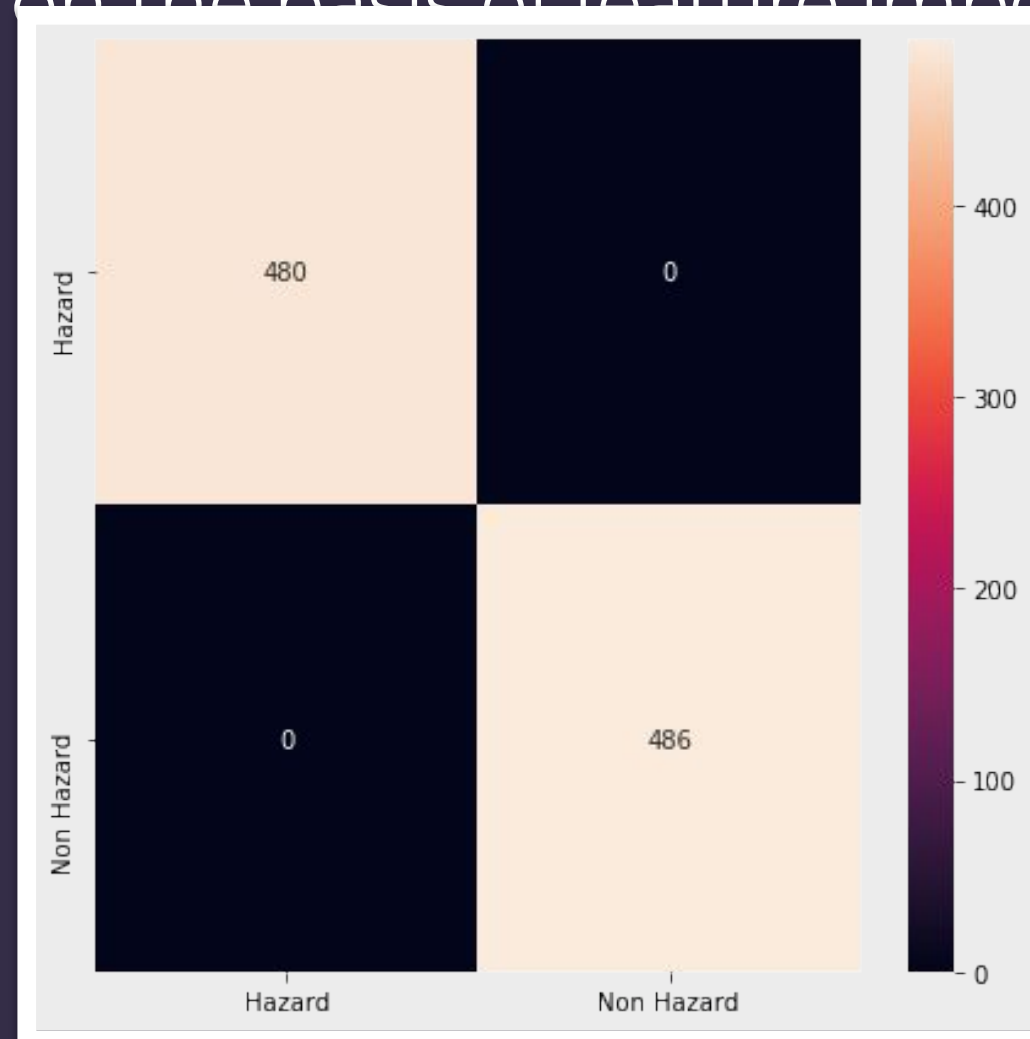
We trained and tested our dataset on different machine learning and deep learning models and the models on which our data showed good accuracy are:

- Decision Tree Classifier
- Random Forest Classifier
- K Nearest Neighbours
- Support Vector Classifier
- Artificial Neural Networks

DECISION TREE CLASSIFIER

Decision Tree is like a flowchart tree structure where every node has a decision function and the leaves are the outcomes. Decision Tree takes very less computation time and is faster than Neural Networks. For noisy and unbalanced data the algorithm does not work well so we need to remove noises to improve the model in such cases. The hierarchy of the tree is decided on the basis of feature importance.

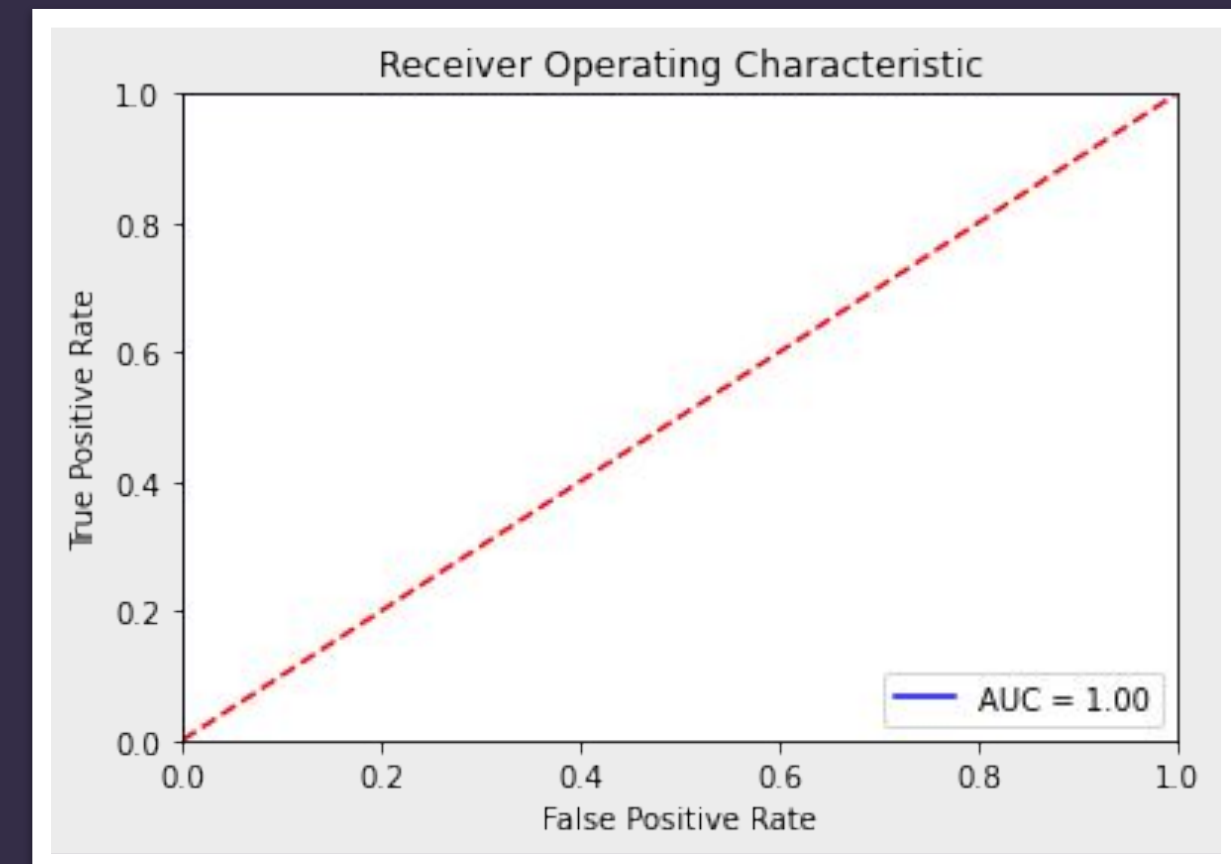
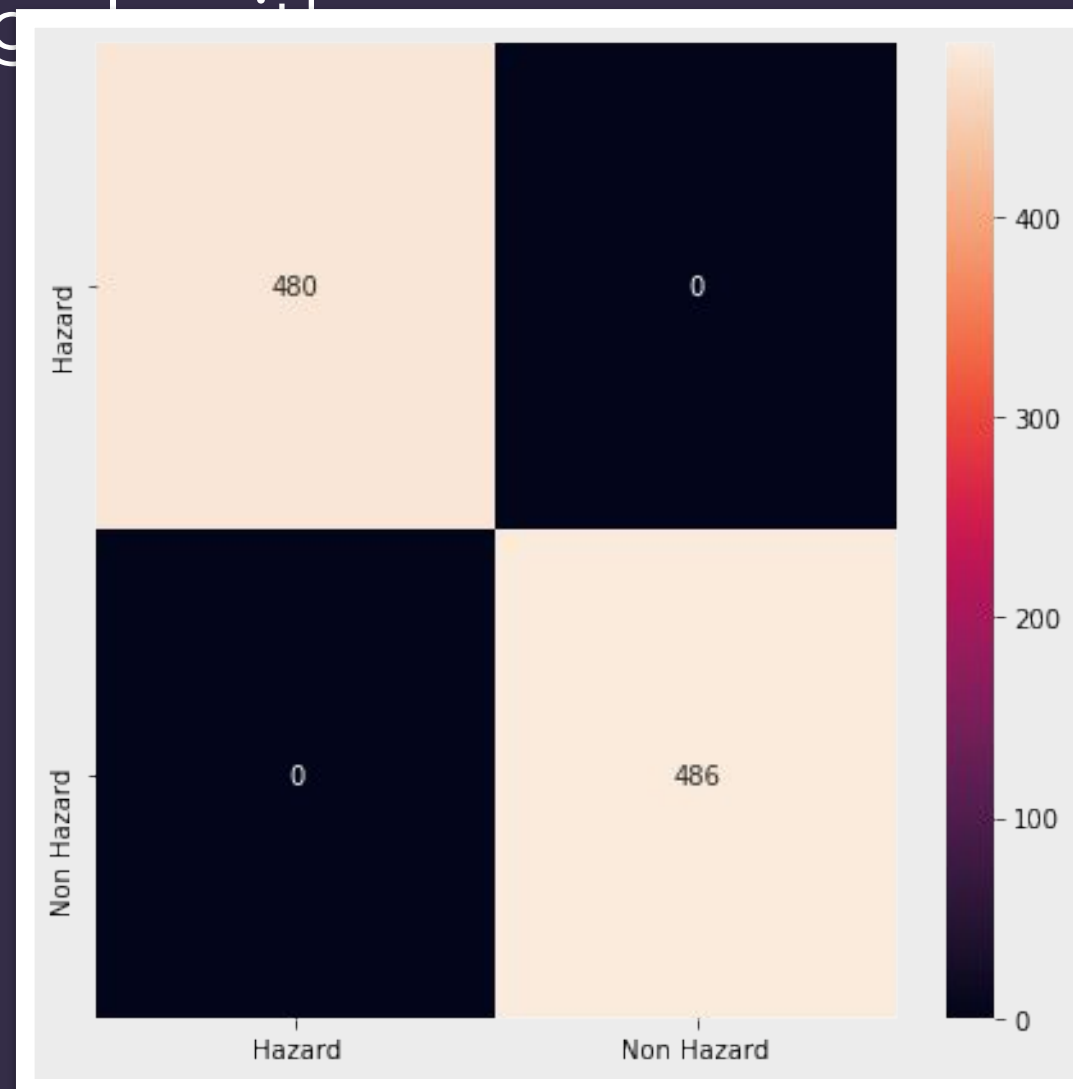
Accuracy: 1.00
Standard deviation: 0.0
ROC AUC SCORE: 1.00



RANDOM FOREST CLASSIFIER

It is an ensemble tree based algorithm. Random Forest classifier is a set of decision trees from randomly selected subset of training set. The trees are voted and the majority voting class decides the final class for the test set. Random forest provides the best accuracy but it takes longer computation time than decision tree, Artificial neural networks and gradient boosting.

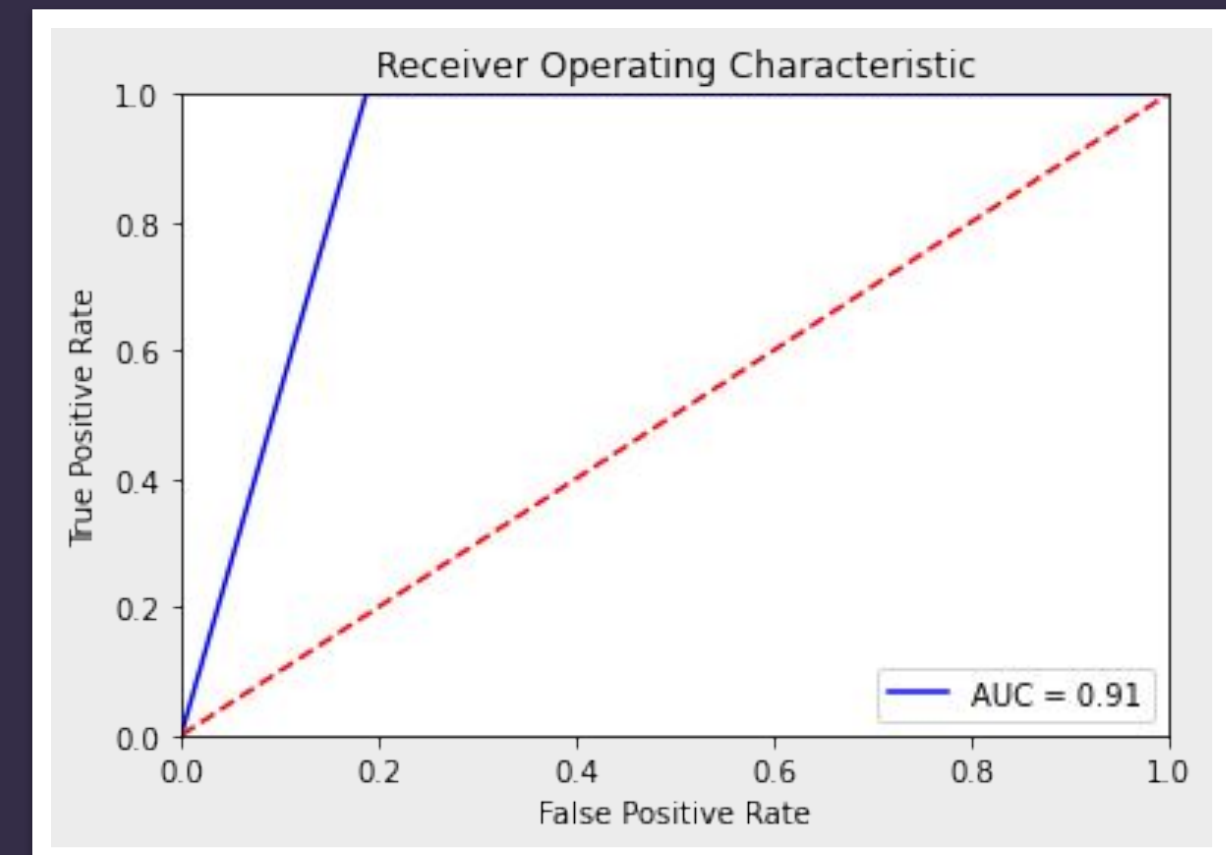
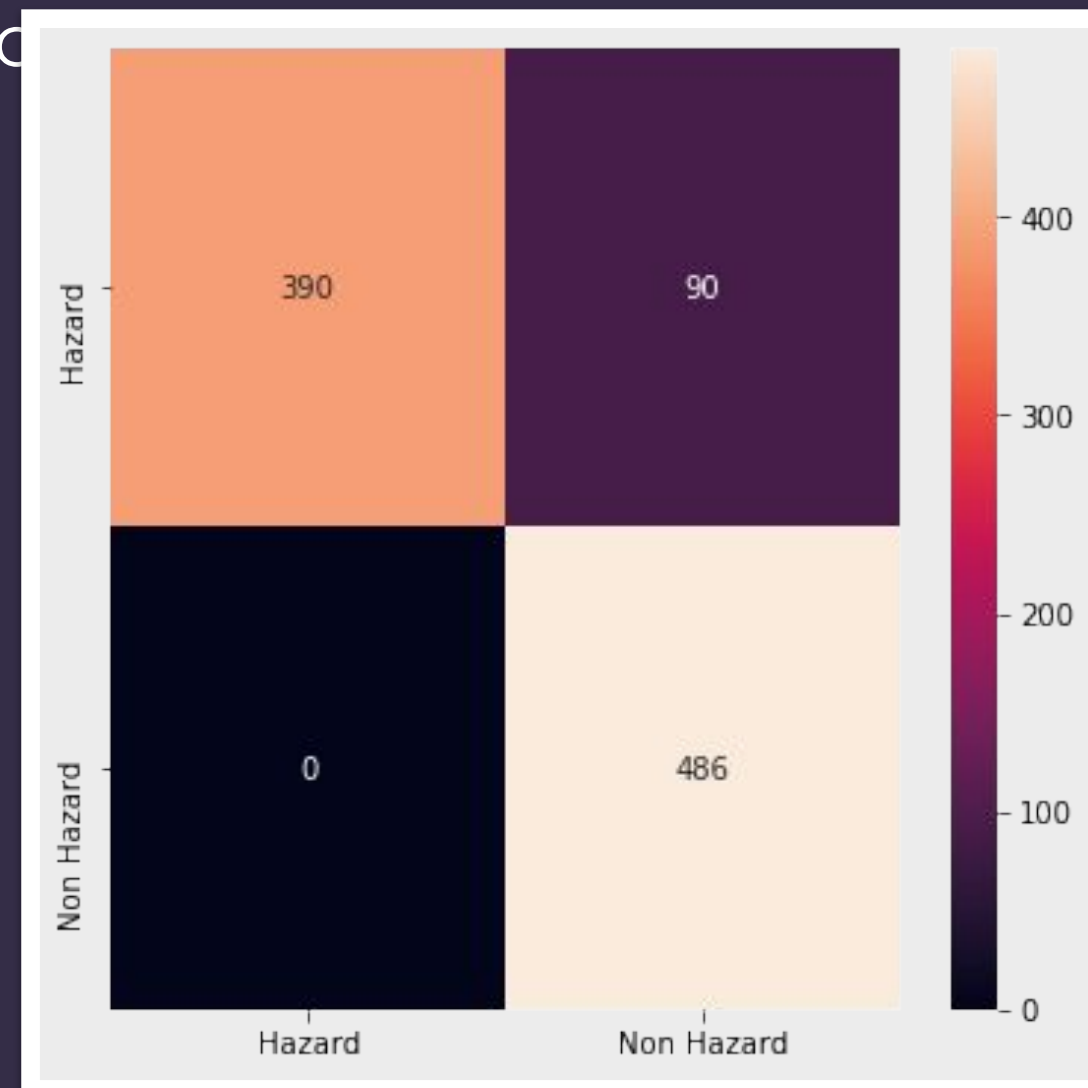
Accuracy: 1.00
Standard deviation: 0.0
ROC AUC SCORE: 1.00



K NEAREST NEIGHBOURS

KNN is a non-parametric and lazy learning algorithm i.e; model structure is determined by the dataset and KNN does not need to be trained for model generation so it makes testing phase slower and consumes high amount of memory. KNN model performs poor at high dimensional dataset as is the case. We normalize the dataset in this case and also tune this algorithm for different hyperparameters to achieve better results.

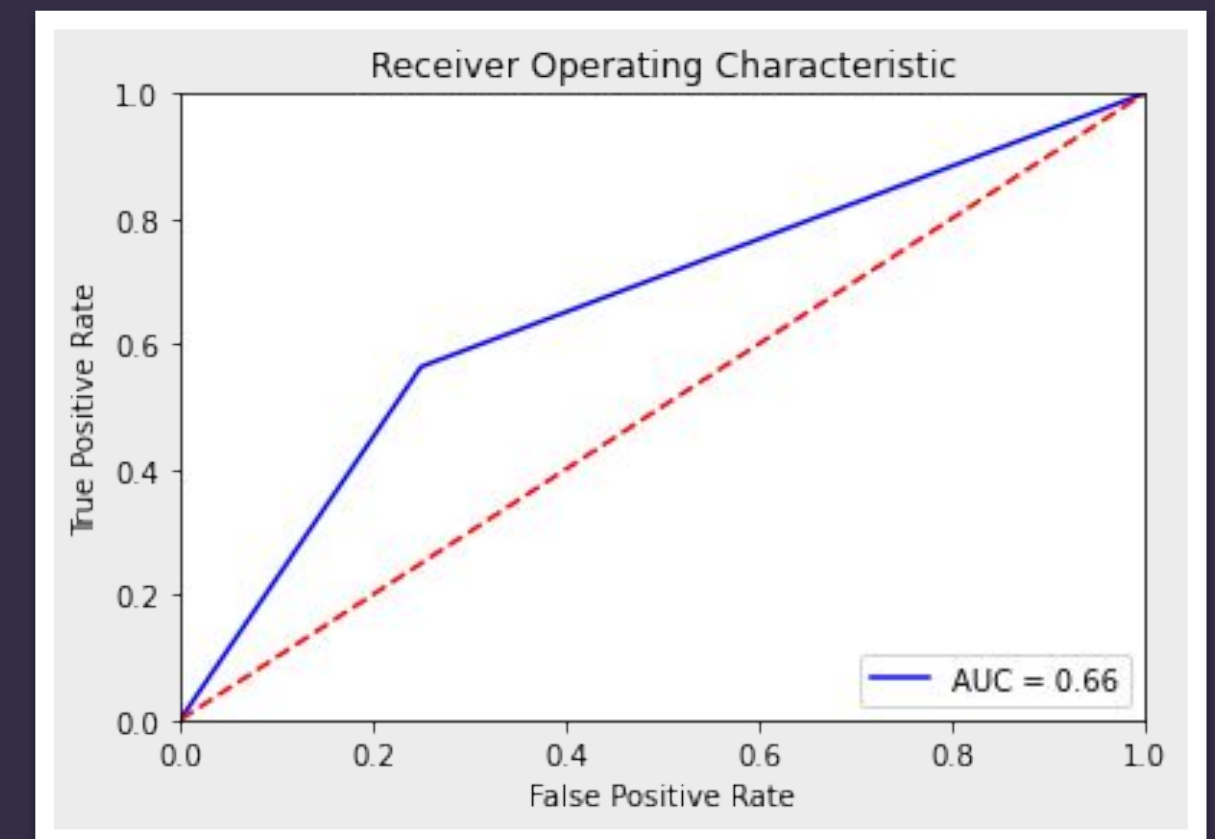
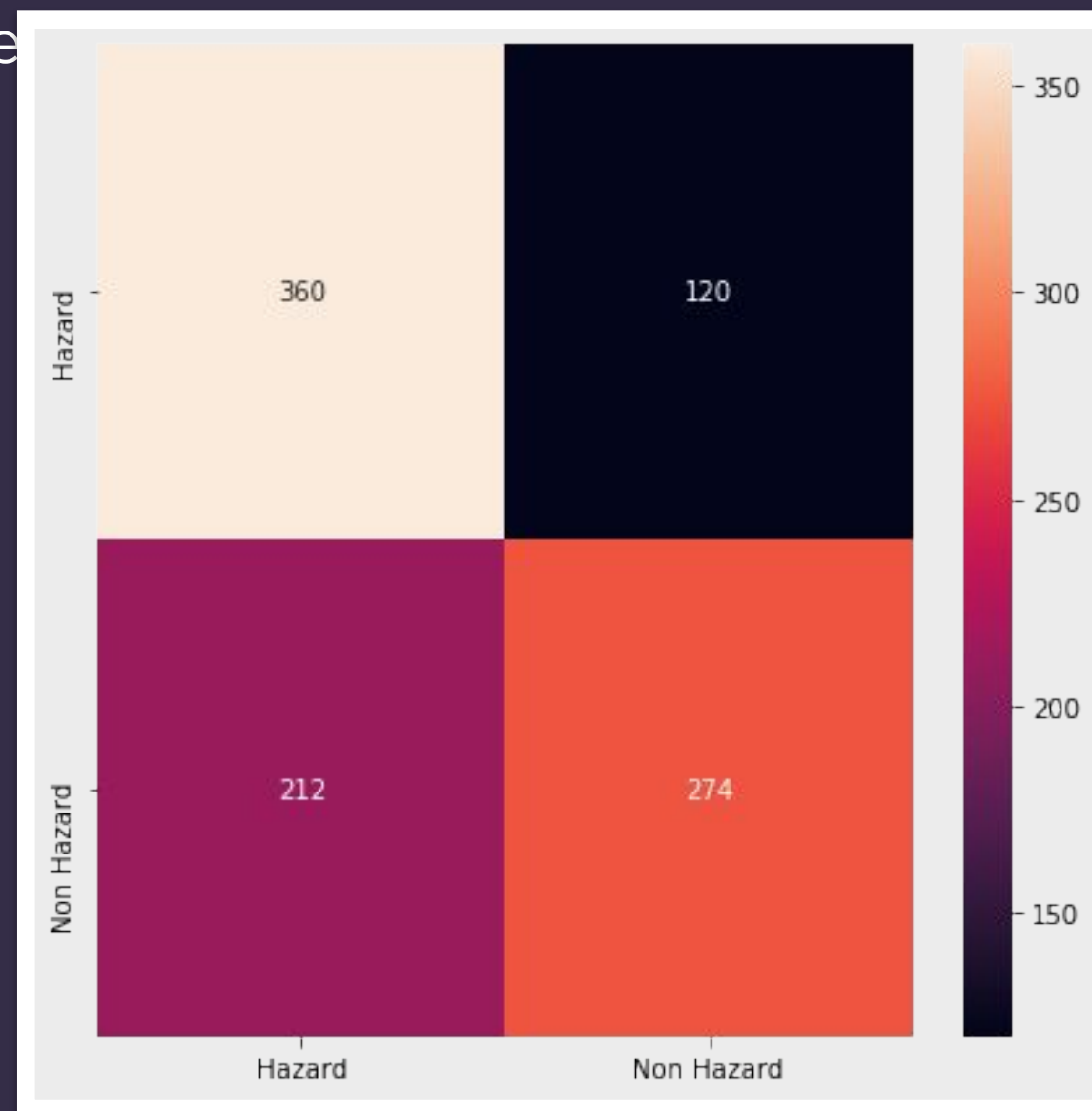
Accuracy: 0.897
Standard deviation : 0.01
ROC AUC SCORE: 0.90



SUPPORT VECTOR CLASSIFIER

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points. The hyperplane with maximum margin is chosen so that future data can be classified with more confidence. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dime

Accuracy: 0.664
Standard deviation : 0.02
ROC AUC SCORE: 0.65



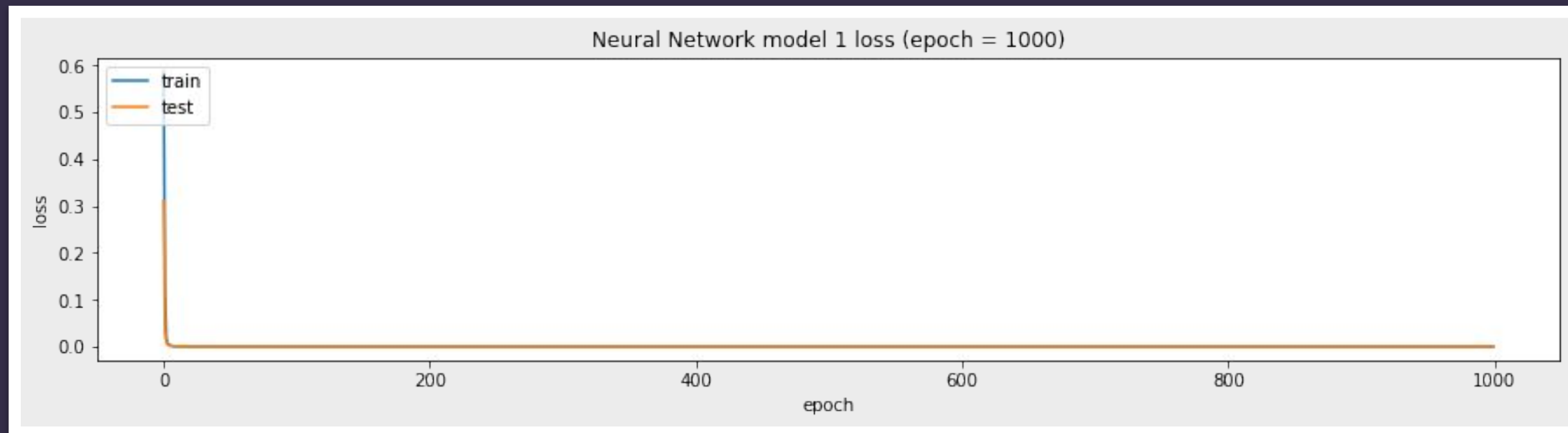
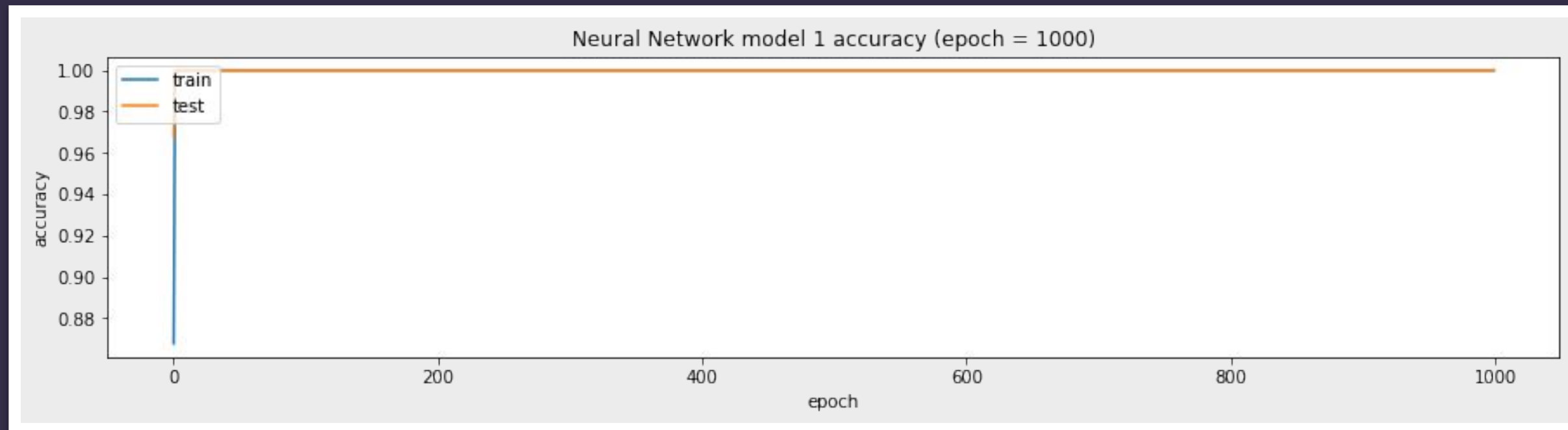
ARTIFICIAL NEURAL NETWORKS

Artificial Neural Network consists of an artificial network of functions, called parameters, which allows the computer to learn, and to fine tune itself by analyzing new data.

We created feed forward sequential model using keras libraray. The network consisted of three dense layers. It took a 2D input of dimension (1, 16) and passed it to next dense layer of 8 nodes through a Relu activation function. The second dense layer passes its input through a function and Relu acivation function to last layer of one node which used sigmoid function to determine the truth value of the input.

The train and test sets were scaled using standard scaler function so that input feature values implicitly weights all features equally in their representation.

ARTIFICIAL NEURAL NETWORKS



Training Accuracy: 1.00

Testing Accuracy: 1.00

Loss: 2.5329e-09

EVALUATION

Decision tree performs good in because it grows exponentially for each level and it is easily able to handle medium to high dimensional data.

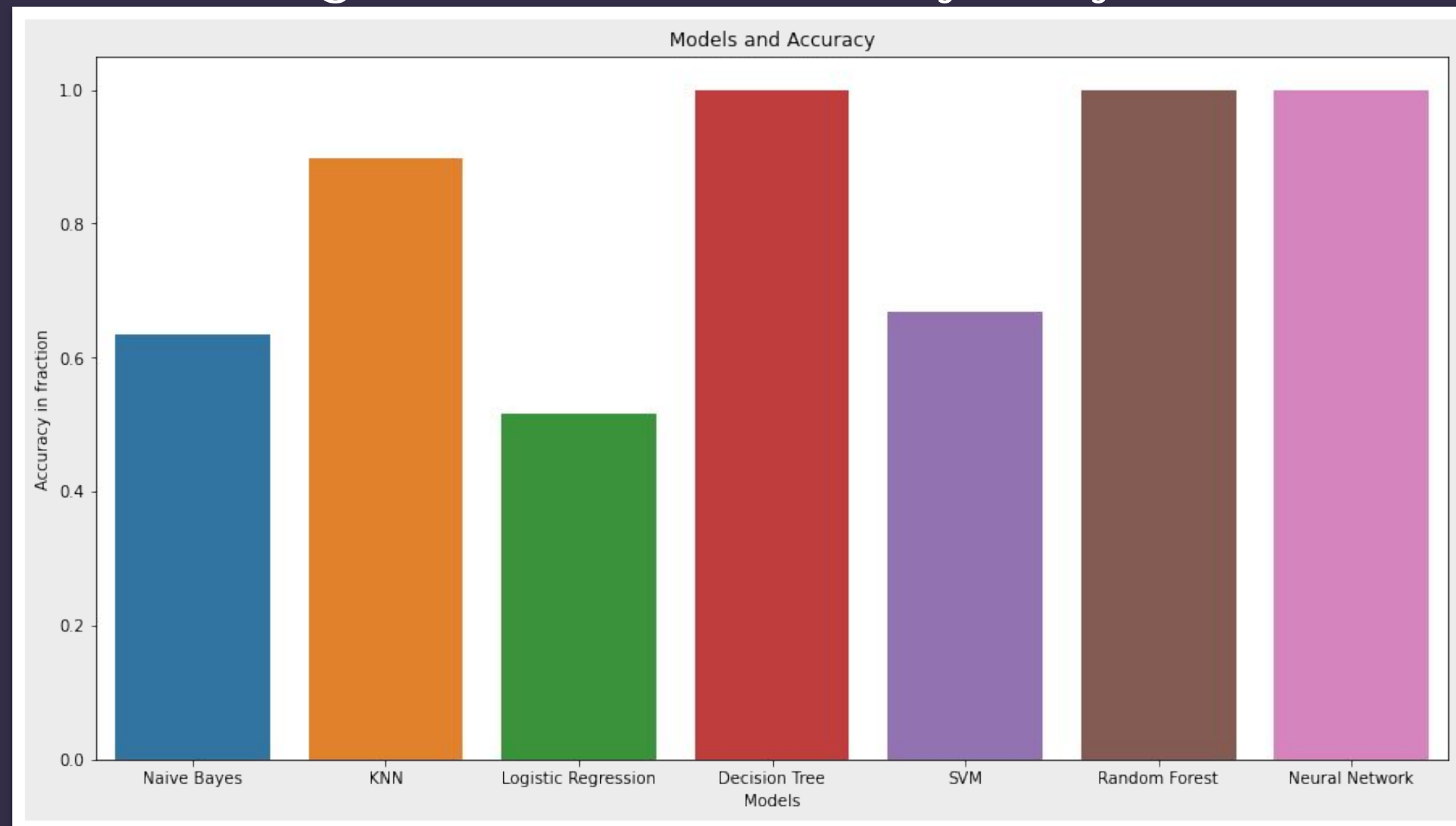
Similarly random forest uses decision tree as one of the component so they perform good for high dimensional data.

KNN model under performs as compared to other algorithms because it uses euclidean distances to cluster the data and in the case of high dimension data clustering becomes difficult since most of the vector data become equidistant to the search query vector.

Similarly, Support vector classifier which uses hyperplanes to classify the data are unable to handle high dimensional data.

EVALUATION


Neural Networks can have a large number of free parameters (the weights and biases between interconnected units) and this gives them the flexibility to fit highly complex data (when trained correctly) that other models are too simple to fit. So, it is able to handle high dimensional data very easily.



TECHNOLOGY AND FRAMEWORK

- I used python language it is powerful high level language and I am well versed with it.
- I used scikit learn library for the development of Machine Learning models as scikit learn provides many useful library and has a very good documentation of it's tools.
- I used keras library on tensorflow as keras is easy to learn and is a very powerful library used for deep learning purposes.
- For the data handling, computation and visualization purposes Pandas, numpy, seaborn and matplotlib was used.
- I used Google colaboratory as our platform so that we can use it's fast and free TPU accelerators which decreases model training and testing time to a great extent.

CONCLUSION



We can see that we get good results for the above dataset and this model helps us in the prediction of seismic hazards for the next shift. The accuracy of the model can be further improved if we collect more data, since that will help us better train the model. So, we can conclude that Machine Learning helps us predict results for serious hazardous problems like earthquake which is not possible to be solved through equations and statistical methods.