

Project 9: Structural Health Monitoring

This manuscript ([permalink](#)) was automatically generated from [dbudzik/project9 SHM@193c24f](#) on December 4, 2020.

Authors

- **David Budzik**
 -  [dbudzik](#)
- **Claudia Mederos**
 -  [cr25](#)
- **Bolaji Lawal**
 -  [bolajilawal](#)
- **Abdullah Assaf**
 -  [aboo12](#)

Abstract

Civil infrastructure all around is subjected to the challenges posed by aging, deterioration and extreme events. In recent years, structural health monitoring has grown in importance as failure and collapse of infrastructure result in harmful effects on a nation's economy and development. Recent advances as seen the use of sensors to obtain the dynamic characteristics of structures. This has led to the potential of collecting dynamic data at more frequent intervals thus leading to continuous monitoring. In this report, we use exploratory data analysis techniques to see if such dynamic data collected from structures can be used to infer the condition of a structure. We have also tried different machine learning algorithms to predict the condition of a structure based on given acceleration signals. For this study, the ASCE benchmark problem experimental phase II structure was used.

Machine Learning Methods

4. Methodology

4.1 Explanation of the Problem:

Our Kaggle Project was focused on Binary classification. Binary Classification is the task of classifying the elements of a set into two groups on the basis of a classification rule. Our problem revolved around classifying the structural damage detected in a structure. We had to choose different machine learning algorithms to help classify the data.

4.2 Steps For each algorithm:

First we had to create a model, softmax or sigmoid used in output layer, so that we could train the model and tune hyperparameters, such as accuracy, precision etc. Finally, we had to test the performance of model against the sample data. This allowed us to compare results from different models and see what model was the most accurate at classifying the data.

4.3 Data Preparation:

First we had to split the train dataset into 2, one set for training and the other for validation. Then we had to shuffle the train dataset before using it. We then used regularizations to apply penalties on layer parameters or layer activity during optimization. Dropouts were used to incorporate non-linearity. The data was normalized by subtracting the mean and dividing by standard deviation.

4.4 Method 1: Logistic Regression:

The first Method utilized was Logistic Regression. It was the simplest model for binary classification. The method outputs a probability between 0 and 1, exclusively. It then defines a decision threshold to map to binary category. The logistic regression yielded great results on the first run with an accuracy of ~96%, low precision ~24%, and recall plummeting to ~20%.

4.5 Method 2: Artificial Neural Networks:

The second method utilized was Artificial Neural Networks. Artificial Neural Networks are more complex than logistic regression. This method adds a bunch of hidden non-linear layers to the logistic regression model. Our group used this method to check if it offered an improvement on the previous model. The artificial neural networks yielded average results with great accuracy ~96%, Improved precision ~45%, Improved recall ~60%, although the metrics were not good overall.

4.6 Method 3: Random Forest Regression:

The third method utilized was Random Forest Regression. Random Forest Regression performs both regression and classification tasks with the use of multiple decision trees and bagging. It is easy to use and often returns good results even without hyperparameter tuning. Our group used this method to check if it offered an improvement on the previous model. The Random Forest Regression provided the worst results with extremely inaccurate accuracy rate, while also not working well for the type of data we had.

4.7 Issue:

We noticed that the problem was the precision and recall are very low even though accuracy is high. This was caused by the damaged data being too sparse, only 3.5% of our data represents the damaged condition. The solution was to add copies of the damaged data into the training set.

4.8 Take Two:

In take two we ran logistic regression and artificial neural networks again to see if it yielded better results with addition of new copies of damaged data. Logistic Regression returned good accuracy ~83%, much better precision ~60%, and Fantastic recall ~99%. Artificial Neural Network returned slightly better data with a fantastic accuracy ~98%, fantastic precision ~91%, and fantastic recall ~99%.

Results

Results go here

Discussion of Results

Discussion goes here

Conclusion

Conclusion goes here

References
