Towards Quasi-Real-Time Monitoring of Seismic Sensor Performance Using Machine Learning

S. Seo, T. Taira, S. Balaji (Berkeley Seismological Laboratory)

Overview

Power Spectral Density Analysis Dataset

- Power Spectral Density (PSD) analysis is a well-established technique for evaluating the performance of seismic sensors.
- The Northern California Earthquake Data Center (NCEDC) continuously computes and archives probability density functions (PDFs) of PSDs across 29 networks and over 2600 stations.
- The database currently exceeds 180 TB and encompasses evaluations over broadband, short-period, and strong-motion sensors, as well as other geophysical instruments.

Performance Monitoring with Machine Learning

- Leveraging the comprehensive NCEDC PDF dataset, we train a binary classifier machine learning (ML) model to quantitatively characterize PDF features for the rapid identification of sensor anomalies.
- This approach facilitates quasi-real-time performance monitoring, ensuring timely detection and response to sensor anomalies.

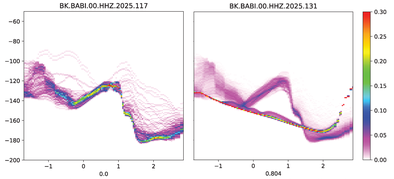


Fig. 1. Machine learning model computations for broadband sensor data: (a) the standard, expected broadband sensor PDF, resulting in a computed value of 0.0, and (b) the broadband sensor reading with an anomaly, yielding a computed value of 0.804. A value closer to 1 indicates that the model has detected an irregularity.

Machine Learning Architecture: Configuration and Parameter Metrics

Neural Network Parameter Configuration

Machine Learning Model Architecture

- The machine learning model contains an input layer corresponding to the number of probability data points from PDF entires, which pertain to 122 x 151 for broadband, 114 x 131 for strong motion, and 124 x 151 for short-period.
- The input is then transformed through two hidden layers of 512 nodes each. The hidden layers utilize a non-linearity function (ReLU) to abstract features of geophysical instrument readings.
- The neural network displays its output throughout the single-node output layer representing a classification of [0, 1] derived from the sigmoid function, with 1 representing positive diagnosis for poor performance of the sensor.

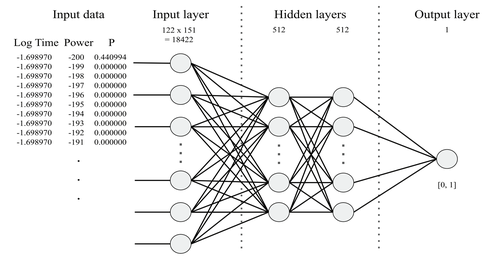


Fig. 2. Visual representation of the neural network architecture.

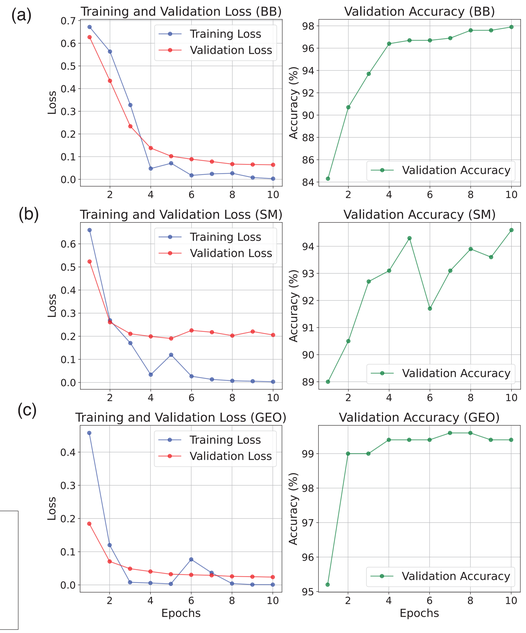
Model Parameter Configuration Equations

Rectified Linear Unit (ReLU): $f(x) = \max(0, x)$

Sigmoid: $\sigma(x) = \dfrac{1}{1 + e^{-x}}$

Binary Cross-Entropy Loss (BCELoss): $L(y, \hat{y}) = -[y \times \log(\hat{y}) + (1 - y) \times \log(1 - \hat{y})]$

Training Loss and Validation Loss



- Across 10 epochs, each machine learning model corresponding to broadband, strong motion, and short-period (also known as geophone) is trained on a pre-processed training dataset.
- A validation dataset, which are a set of entries that the machine never learns from but rather tests its parameters against after each epoch, is used to quantify the learning progress of the neural network.
- The loss function calculates the difference between the predicted probability (output from the model) and the actual truth (ground truth label). We selected the Binary Cross-Entropy loss function, which features high loss when the predicted probability is far from the true label on a binary scale of 0 to 1.
- Across 10 epochs, we find that each machine learning model obtains over 94% validation accuracy, indicating that it achieves such score on a foreign dataset it was not trained on.

Fig. 3. Training and Validation losses recorded for machine learning models tailored towards (a) broadband, (b) strong-motion, (c) short-period (geophone) respectively. Over 10 epochs, the data features a decrease in both training and validation loss overall, indicating the model's learning progress. Validation accuracy is also provided for a more intuitive metric in how correct the machine is at a certain point in time.

Neural Network Performance Metrics

- The validation set, a pre-processed dataset used to test the NN, can be used as metric to finalize the model's capability.
- The four metrics (True Positive, True Negative, False Positive, False Negative) can be used as parameters to determine the neural network's performance such as precision, recall, specificity, and accuracy.

Confusion Matrix for Short-Period (Geophone) NN

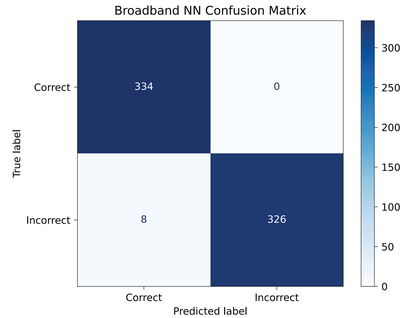


Fig. 4. Confusion Matrix calculating the TP, TN, FP, and FN over the validation dataset for short-period (geophone) neural network's decisions.

Performance Metrics Using Confusion Matrix

Accuracy: $\dfrac{TP + TN}{TP + TN + FP + FN}$

Precision: $\dfrac{TP}{TP + FP}$

Recall: $\dfrac{TP}{TP + FN}$

Specificity: $\dfrac{TN}{TN + FP}$

F1 Score: $\dfrac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

Across the Three ML Models

The following metrics were calculated across machine learning models trained on the broadband, strong motion, and short-period datasets:

- Accuracy: The overall proportion of correct predictions made by the model. It is a general measure of model performance.
- Precision: Proportion of correct positive predictions made by the model. It is useful when you want to minimize false positives.
- Recall: How well the model can identify positive cases. Useful when negatives are costly.
- Specificity: How well the model can identify negative cases. Useful when positives are costly.
- F1 Score: A balance between precision and recall. It is a general measure of model performance.

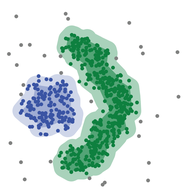
Broadband Statistics

- Precision: 0.9970059880239
  Recall: 0.9823008849557
  F1 Score: 0.9895988112927
  Specificity: 0.9820359281437
  Accuracy: 0.9895209580838

DBSCAN: Acceleration of Labeling Training Data

Automization of Data Processing

- The presence of over 180 TB data entries in the Northern California Earthquake Data Center necessitates a strategy to accelerate the process of manually labeling and reviewing training and validation PDFs
- Our solution leverages an algorithm called Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to separate large chunks of data into groups based on shared statistical features, allowing for manageable human review in obtaining training and validation data.



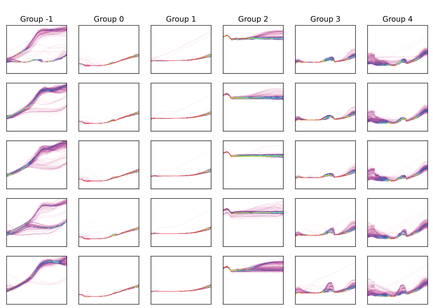
- Based on a heuristic function, DBSCAN algorithmically determines core points to group data on, with an emphasis on regional density.
- This allows robust detection of outliers, an important consideration when labeling data with binary values.

Fig. 5. Map view of a DBSCAN algorithm. It should be noted that these points on the two dimensional space is purely conceptual, and the actual implementation of DBSCAN utilizes a heuristic function to determine the distance between two points in space. The algorithm has been tailored so that each point corresponds to a point, and a personalized heuristic function determines the distance between two PDFs.

DBSCAN Heuristic

- For DBSCAN to accurately separate graphs with similar traits into the same groups, we must create a heuristic function to algorithmically determine the closeness of two PDF entries.
- We chose the Manhattan Distance (L1 norm), a method for calculating the distance between two matrices of the same size by summing the absolute differences of their corresponding elements.

$$D = \sum |Ak - Bk|$$



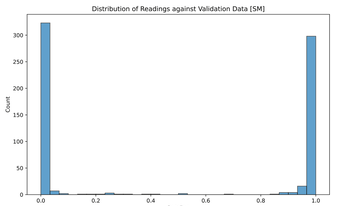
- The DBSCAN algorithm was leveraged to separate a large scope of data ranging from 2022 to 2025 (47,640 data entries for broadband sensors, 45,298 for strong-motion sensors, and 11,501 for short-period (geophone) sensors.
- The separated groups were then reviewed by previews (Fig. 6), allowing for a drastic acceleration in manual labeling of data entires.
- Entries for the broadband sensor required the manual labeling of 1,904 out of 47,640 entries, a 96% decrease in the number of data entries required for manual review.
- Strong motion and short-period (geophone) sensors required the labeling of 160 and 89 entries respectively, resulting in a 99.6% and 99.2% decrease in the number of data entries required for manual review.

Fig. 6. An output of the DBSCAN algorithm on data batch containing 1000 strong motion reading entries. Based on the heuristic and the epsilon value (a limiting number used to define the maximum boundaries of a single group), the DBSCAN algorithm dynamically decides the number of groups as well as the number of outliers that can then be reviewed manually. An epsilon value ranging from 23 to 26 has displayed optimal performance.

ROC Curve: Binary Classifier Performance Metric

Distribution of Classifier Readings

- A binary classifier displays an output ranging from the value of 0 to 1.
- It is often the case that a classifier, given an input data to analyze, produces a result that is close to the value of 0 or 1, but not exactly identical.
- As a result, a threshold is used to define the boundaries at which the neural network differentiates a positive result from the negative. The default threshold for Berkeley Seismology Laboratory's models is 0.5, where any ML output below 0.5 is considered negative and above positive.



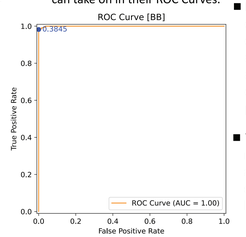
Trade-Off Between TP/TN and FP/FN

- Alternatively, the threshold could be set to any value between 0 and 1 in order to change the boundaries at which the machine learning model decides an input is positive or negative.
- Adjusting the threshold results in changes to the rate at which the model produces true positives and true negatives. For example, an extremely low threshold will naturally increase the rate of false negatives, as the tolerance for a positive is much lower.
- The model's performance can be measured across multiple thresholds in order to determine the confidence with which it makes its decisions, as well as the quality of validation data.

Fig. 7. The distribution of strong motion sensor ML model readings against validation data. Most of the analysis run by the neural network results in a value very close to 0.0 or 1.0, a good indicator of the machine's decisiveness in its evaluations. A small number of values fall into the range between 0.2 to 0.8, which can either be attributed to the imperfection of the current machine learning model or the actual presence of ambiguity within the validation dataset presented to the model.

ROC Curve: Measuring the Model's Performance

Fig. 8. Guidelines on the different shapes that classifiers can take on in their ROC Curves.



- An ROC (Receiver Operating Characteristic) Curve is a graphical representation used to evaluate the performance of a binary classification model.
- As the thresholds are adjusted, different True Positive Rate (TPR) and False Positive Rate (FPR) are recorded on the graph, eventually connecting to create a curve.
- Because a perfect classifier will have a 100% TPR and 0% FPR, models that can approach the top left corner the closest are considered to have optimal performance.
- By contrast, models that perform worse will display output values similar to a completely random classifier, which will randomly guess between 0 and 1 at any given input. This results in a diagonal line across the ROC Curve, as the threshold will directly correlate to the rate of true positives and false positives.
- The Area Under the Curve (AUC) is a metric used to measure how close the machine learning model is to a perfect classifier. A high AUC indicates that at a certain threshold, the classifier approaches the top left corner in close proximity.