Predictive Modeling for Tsunami Risk Assessment from Seismic Data

Name: Justin Xiao

Affiliation: Brown University

Github Repository: https://github.com/JustinXre2020/Earthquake-Tsunami/

Introduction

Motivation

Tsunamis are among the most catastrophic natural hazards and often follow sudden seafloor displacement during large earthquakes. This project tackles rapid tsunami prediction by using machine learning to classify earthquakes by their tsunamigenic potential. The goal is to help disaster agencies strengthen early warning systems and coastal safety. By identifying earthquakes likely to generate tsunamis, authorities can issue faster evacuation alerts, reducing loss of life and improving resilience.

Dataset Properties

This study uses about 1,000 historical significant earthquakes with features such as magnitude, focal depth, MMI, and CDI. Seismic energy is right-skewed and long-tailed, and the binary target is imbalanced (33% Tsunami, 67% No Tsunami) for predicting tsunami occurrence from immediate physical parameters.

Previous Work

Prior work has applied ensemble and gradient-boosting methods, including Random Forest, and XGBoost, often reaching ROC-AUC values around 0.7 – 0.85. Despite strong overall discrimination, achieving high precision remains difficult because many large earthquakes do not cause the seafloor displacement needed for tsunamis. This project focuses on recall and f1 score as core metrics given the disastrous nature of Tsunami.

EDA (Exploratory Data Analysis)

Exploratory analysis shows three interesting findings: Earthquake Magnitude vs. Depth Patterns by Tsunami Occurrence, Global Distribution of Earthquakes and Tsunamis Along the Pacific Ring of Fire and Depth Distribution Differences Between Tsunami and Non-Tsunami Events.

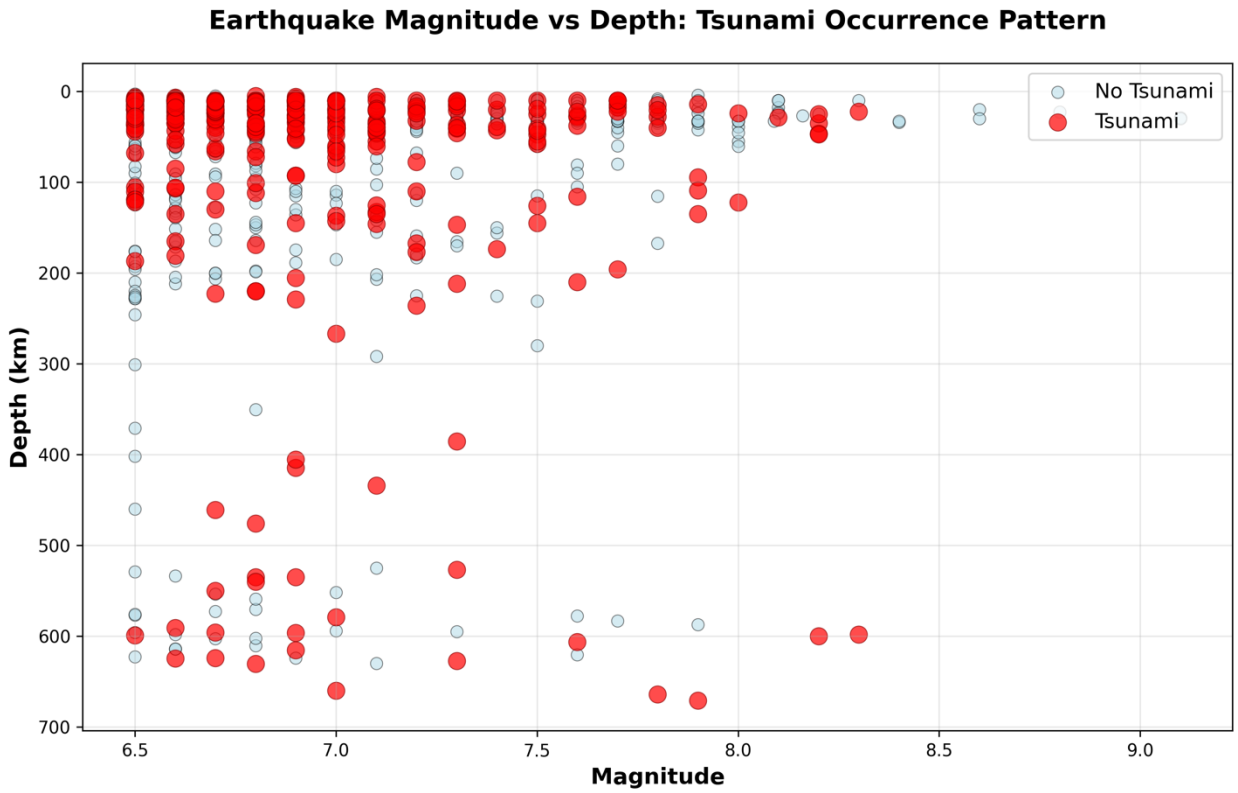

Figure 1: Magnitude and Depth: Tsunami-generating events tend to occur at higher magnitudes and shallower depths compared to non-tsunami events.

This scatter plot shows that Tsunami events (red) cluster mostly in shallow quakes (~0–100 km) across 6.5–8.5 magnitude, while deeper events are rarer and less consistently tsunamigenic. The strong overlap between classes indicates magnitude and depth alone can't clearly distinguish tsunami-producing earthquakes.

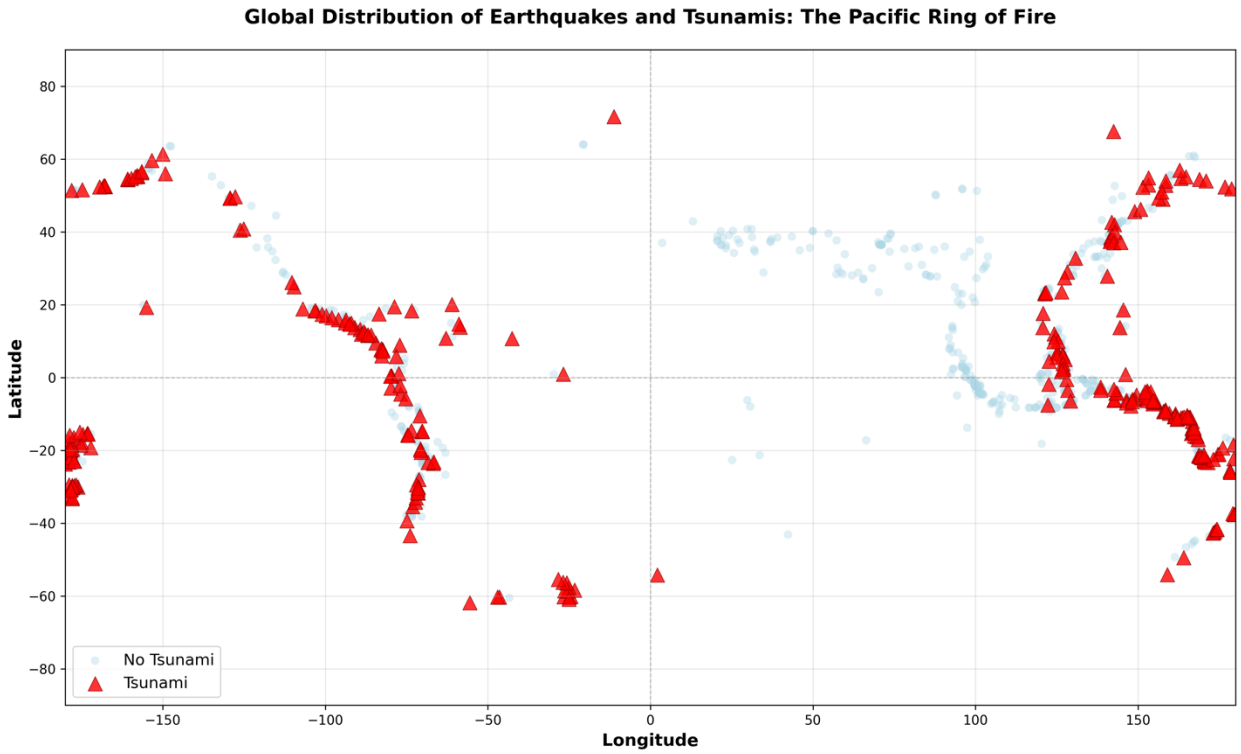

Figure 2: Global earthquake and tsunami events concentrate along the Pacific Ring of Fire, underscoring the role of plate-boundary subduction zones in tsunamigenic risk.

The map shows earthquakes and tsunami events clustering along tectonic plate boundaries, with the strongest concentration around the Pacific “Ring of Fire.” Tsunamis (red triangles) are most common near subduction zones along the Americas and the western Pacific, with far fewer events in continental interiors. Overall, tsunami occurrence is highly geographic and closely tied to coastal plate-boundary seismicity.

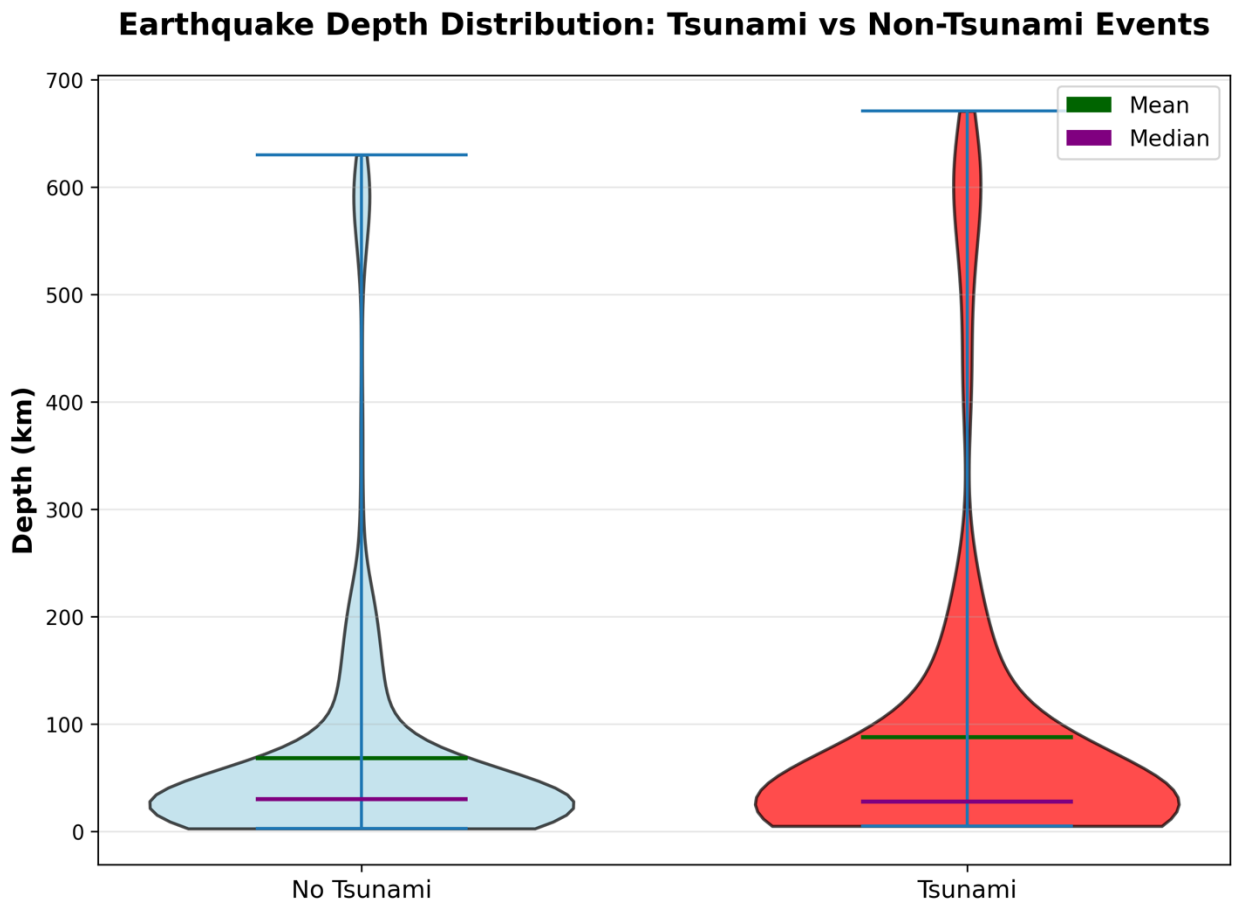

Figure 3: Tsunami-generating earthquakes generally occur at shallow depths but show a wider depth distribution than non-tsunami events.

This violin plot compares earthquake depths for tsunami and non-tsunami events. Both groups show most earthquakes occurring at shallow depths, but tsunamis (red) have a slightly higher mean and broader distribution, with a few extending to very deep levels. Non-tsunami events (blue) are more concentrated near the surface. Overall, tsunamigenic earthquakes tend to occur in relatively shallow but somewhat deeper ranges on average than non-tsunamigenic ones.

Methods

Data Assembly and Splitting

Two earthquake-event datasets were concatenated after identifying shared columns, and duplicate events were removed. The target variable was tsunami, with all remaining columns used as features. For each run, the data was split into training/validation (80%) and a held-out test set (20%) using stratification to preserve class proportions. Hyperparameter tuning was performed only within the 80% subset using 5-fold StratifiedKFold. The full pipeline was repeated across ten random seeds [0,1,7,13,21,42,66,88,123,2025] to assess robustness.

Feature Engineering

Two engineered ordinal features were created:

1. tsunami_potential_bin, computed from magnitude and clipped depth and binned into (none, low, medium, high)
2. gap_bin, derived by binning original gap feature into (poor, fair, good, excellent) to reflect measurement quality.

In addition, a set of non-modeling or redundant fields (e.g., metadata/text/location variables) was dropped after feature creation.

Data Preprocessing

A ColumnTransformer was used with remainder='drop':

- Ordinal Encoding: tsunami_potential_bin and gap_bin were imputed with a constant and encoded using OrdinalEncoder with fixed orderings (unknown categories encoded as −1).
- Min–max Scaling: longitude, latitude, mmi, cdi were median-imputed and scaled with MinMaxScaler.
- Standardization Scaling: depth and magnitude were median-imputed and scaled with StandardScaler.

Model Pipeline

The machine learning pipeline for the tsunami prediction model was developed through a systematic four-step process.

- **Splitting**: The processed dataset was partitioned into an 80% set for train/validation and a 20% set for test.
- **Preprocessing**: A ColumnTransformer was fitted using the training dataset to transform both the validation and test sets while preventing data leakage.
- **Training/Tuning with Grid Search and Cross-Validation**: Grid search was implemented to identify the best hyperparameters for each model under different random states. The use of cross-validation allowed for performance evaluation across multiple subsets to reduce the risk of overfitting.
- **Evaluation**: The best-performing model identified through grid search was evaluated on the independent test set to assess its generalization performance.

Machine Learning Pipeline and Hyperparameter Tuning

Hyperparameter selection was performed within the training/validation portion of each run using stratified cross-validation, with F1-score as the primary selection metric. The full procedure was repeated across ten random seeds [0,1,7,13,21,42,66,88,123,2025] of four models used.

Table 1: Tuned Hyperparameters for each ML model, ratio is 2 since occurrence of No Tsunami/Tsunami is 2

Algorithm	Hyperparameter	Grid Search Values
XGBoost	max_depth	[3, 4, 5]
	reg_lambda	[1, 10, 50]
	reg_alpha	[1, 10]
	gamma	[1, 5]
	min_child_weight	[5, 10, 20]
	scale_pos_weight	[ratio, ratio × 0.8, ratio × 1.2]
Random Forest	n_estimators	[100, 200, 300]
	max_depth	[3, 5, 10, 20, 50]
	max_features	[0.3, 0.5, 0.7, 'sqrt']
SVM	C	[0.01, 0.1, 1, 10, 100]
	gamma	[0.001, 0.01, 0.1, 1, 10]
	kernel	['rbf', 'linear']
Logistic Regression	C	[0.001, 0.01, 0.1, 1, 10, 100, 1000]
	solver	['liblinear', 'saga']

Evaluation Metrics

Final performance was reported on the held-out test set using Accuracy, Precision, Recall, and F1-score. Given class imbalance and the cost of missed tsunami events, F1-score and Recall were emphasized. Mean +/- standard deviation across the ten random seeds summarized robustness. Also given the high stakes of tsunami prediction (where missing a positive case can be catastrophic), Recall is particularly critical to ensure we capture as many true tsunami events as possible.

Uncertainty and Robustness Analysis

There are many sources of performance uncertainty in the model. First, data leakage from the original dataset (e.g., dmin and alert may contain future information) can inflate evaluation results. Second, using grid search across ten random seeds introduces additional variability due to different train/validation splits. Finally, both XGBoost and Random Forest involve randomness during training, which can lead to run-to-run performance differences. To improve robustness, leakage-prone features are removed and stratified K-fold cross-validation with randomized splitting is used to better estimate generalization performance.

Results

Model Performance and Baseline Comparison

All models significantly outperformed the baseline accuracy of 0.67.

Table 2: Performance Summary on Test Set with Standard deviation.

Model	Accuracy	Precision	Recall	F1-Score
XGBoost	0.7650 ± 0.0270	0.6041 ± 0.0369	0.8262 ± 0.0492	0.6961 ± 0.0235
Random Forest	0.7725 ± 0.0270	0.6590 ± 0.0467	0.6262 ± 0.0572	0.6409 ± 0.0444
SVC	0.6775 ± 0.0439	0.5042 ± 0.0442	0.7431 ± 0.0596	0.6001 ± 0.0470
Logistic Regression	0.6960 ± 0.0232	0.5530 ± 0.0629	0.3708 ± 0.0512	0.4410 ± 0.0445
Baseline	0.6700 ± 0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000

Table 3: Best Hyperparameter for all four models.

Model	Best Hyperparameters
XGBoost	n_estimators=5000, max_depth=4, learning_rate=0.1, gamma=1, min_child_weight=5, subsample=0.8, colsample_bytree=0.8, reg_alpha=1, reg_lambda=10, scale_pos_weight=2.49
Logistic Regression	C=1000, solver=liblinear
SVC (RBF)	C=100, gamma=0.1, kernel=rbf
Random Forest	n_estimators=300, max_depth=10, max_features=0.5

Performance Analysis

Against the baseline (accuracy 0.67, F1 = 0.0), SVC is only 0.17 std above baseline (0.6775 +/- 0.0439) and Logistic Regression is 1.12 std above baseline (0.6960 +/- 0.0232), while Random Forest (0.7725 +/- 0.0270) and XGBoost (0.7650 +/- 0.0270) perform best at about 3.8 std and 3.5 std above baseline and achieve strong F1 scores (0.6409 +/- 0.0444 and 0.6961 +/- 0.0235).

XGBoost offered the best overall tradeoff for tsunami detection, with the highest Recall (0.8262 ± 0.0492) and F1 (0.6961 ± 0.0235) while keeping moderate Precision (0.6041 ± 0.0369). Random Forest was more conservative, achieving the highest Accuracy (0.7725 ± 0.0270) and Precision (0.6590 ± 0.0467) but lower Recall (0.6262 ± 0.0572), meaning fewer false alarms but more missed tsunamis. SVC had relatively high Recall (0.7431 ± 0.0596) but low Precision (0.5042 ± 0.0442), resulting in a mid-range F1 (0.6001 ± 0.0470), while Logistic Regression performed worst for tsunami detection (Recall 0.3708 ± 0.0512, F1 0.4410 ± 0.0445) despite moderate Accuracy (0.6960 ± 0.0232).

Conclusion on Model Selection

Given the high-stakes nature of tsunami prediction, Recall is the most critical metric because false negatives represent missed tsunami events. Under this criterion, XGBoost is the preferred model, achieving the best Recall (0.8262 ± 0.0492) and the best overall F1-score (0.6961 ± 0.0235). If minimizing false alarms is prioritized instead, Random Forest offers higher Precision (0.6590 ± 0.0467) but with a substantial reduction in Recall.

Figures

Confusion Matrix

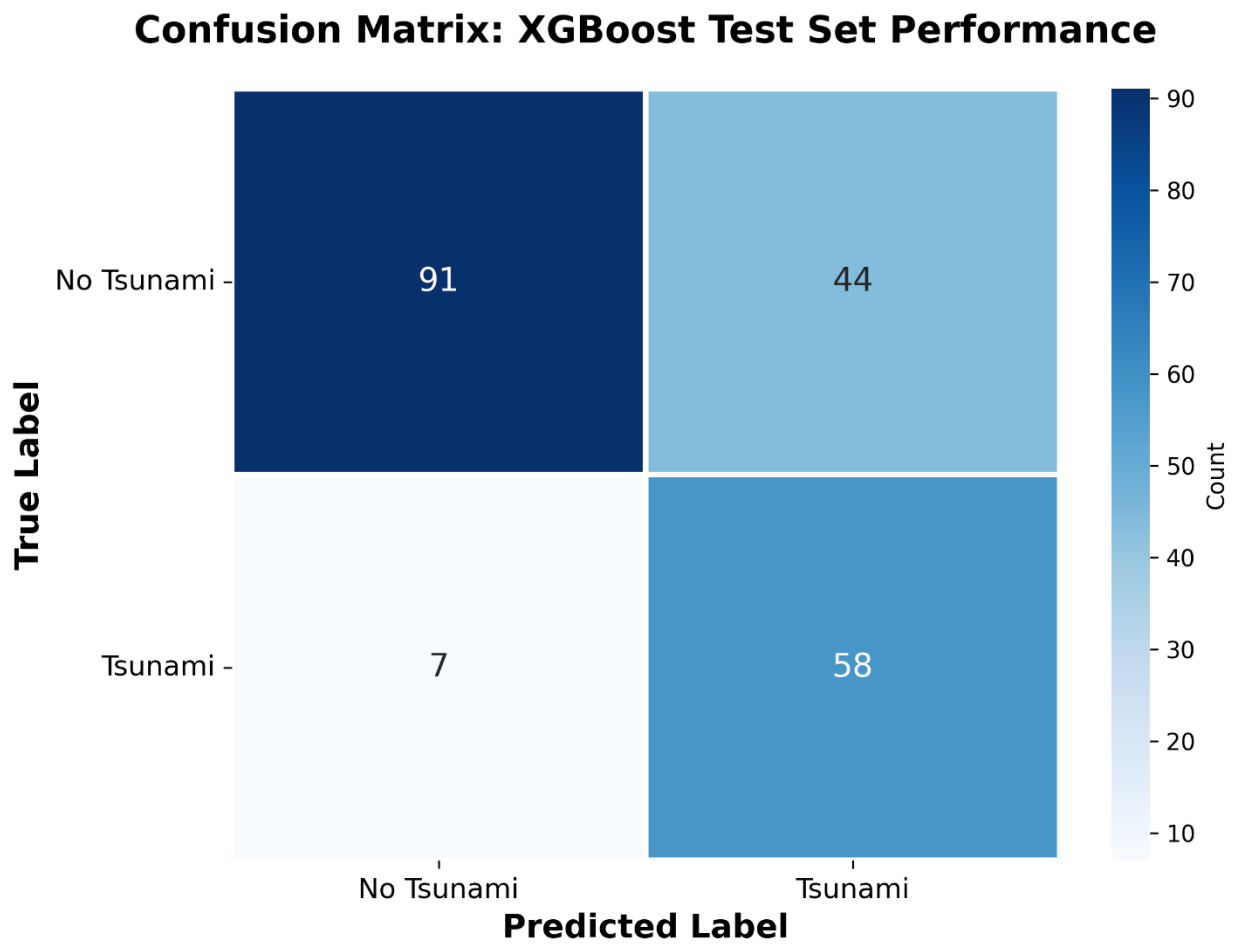

Figure 4: Confusion Matrix for XGBoost Model.

This confusion matrix summarizes XGBoost's test performance for tsunami prediction: it correctly identifies 58 tsunami events (true positives) while missing 7 (false negatives), indicating high recall (~0.89). However, it also flags 44 non-tsunami events as tsunamis (false positives), which lowers precision (~0.57). Overall, the model achieves 149/200 correct predictions (~0.75 accuracy), with most errors coming from false alarms.

ROC Curves

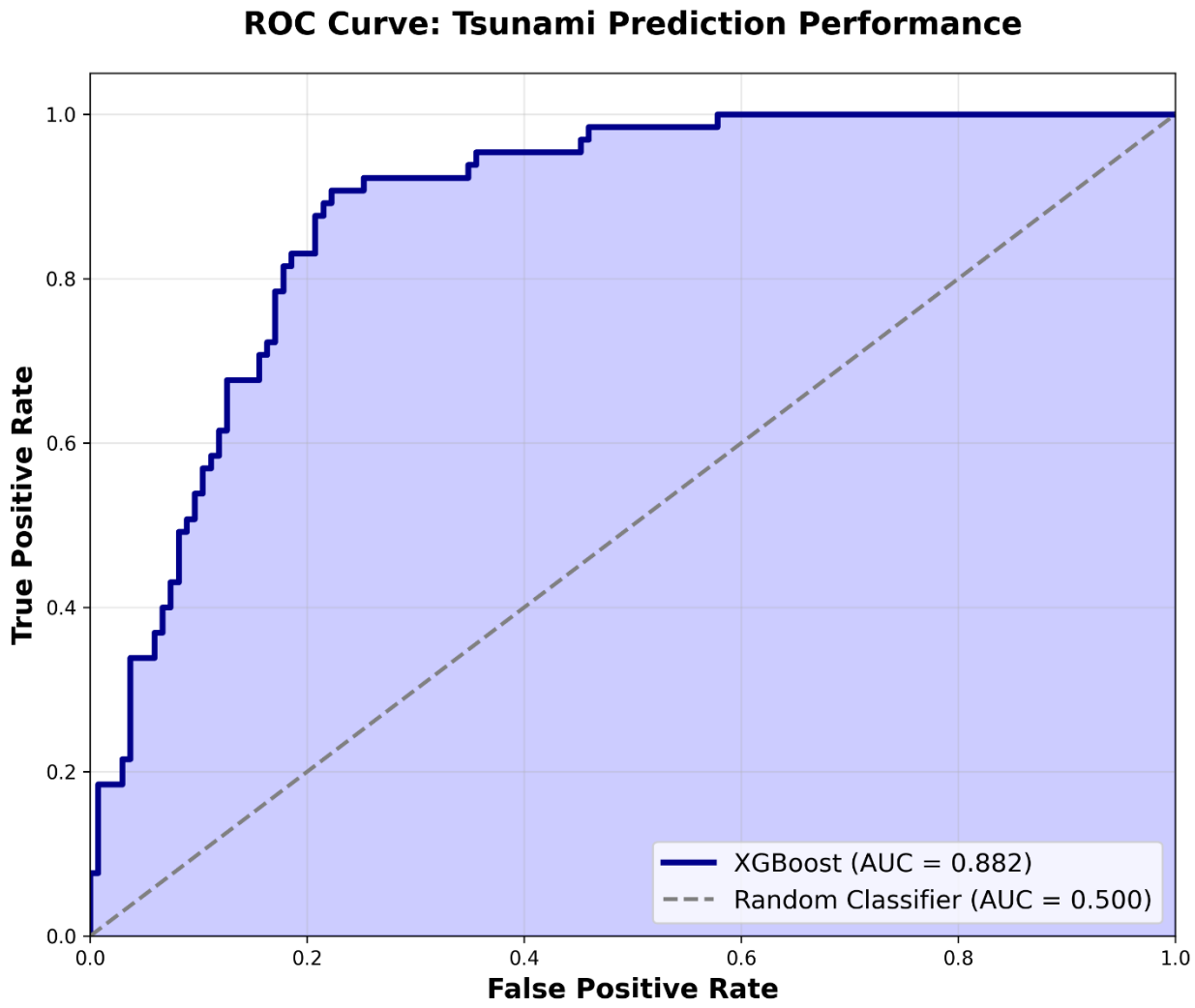

Figure 5: ROC Curve for the XGBoost showing an AUC of 0.882, demonstrating excellent class separation capabilities.

This ROC curve shows that the XGBoost model has strong discrimination ability for tsunami prediction, with an AUC of 0.882, meaning it can reliably rank tsunami events above non-tsunami events across many threshold choices. The curve stays well above the diagonal random-classifier baseline (AUC = 0.500), indicating substantially better-than-chance performance, especially at low false-positive rates where the true-positive rate rises quickly.

Global Feature Importance

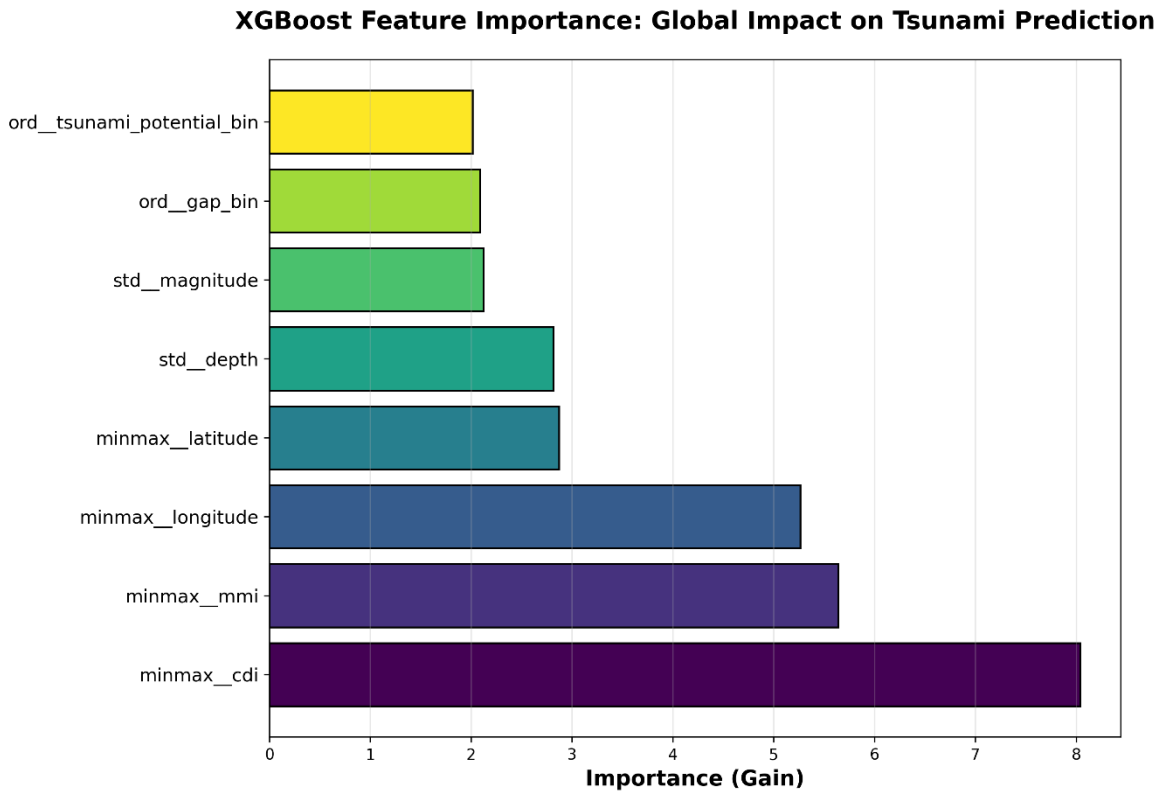

Figure 6: XGBoost Feature Importance (with Gain). This chart illustrates global feature importance using the Gain metric.

This gain-based XGBoost importance plot shows CDI and MMI as the dominant predictors, followed by longitude/latitude, while depth and magnitude contribute less and binned features (gap, tsunami potential) have the smallest impact.

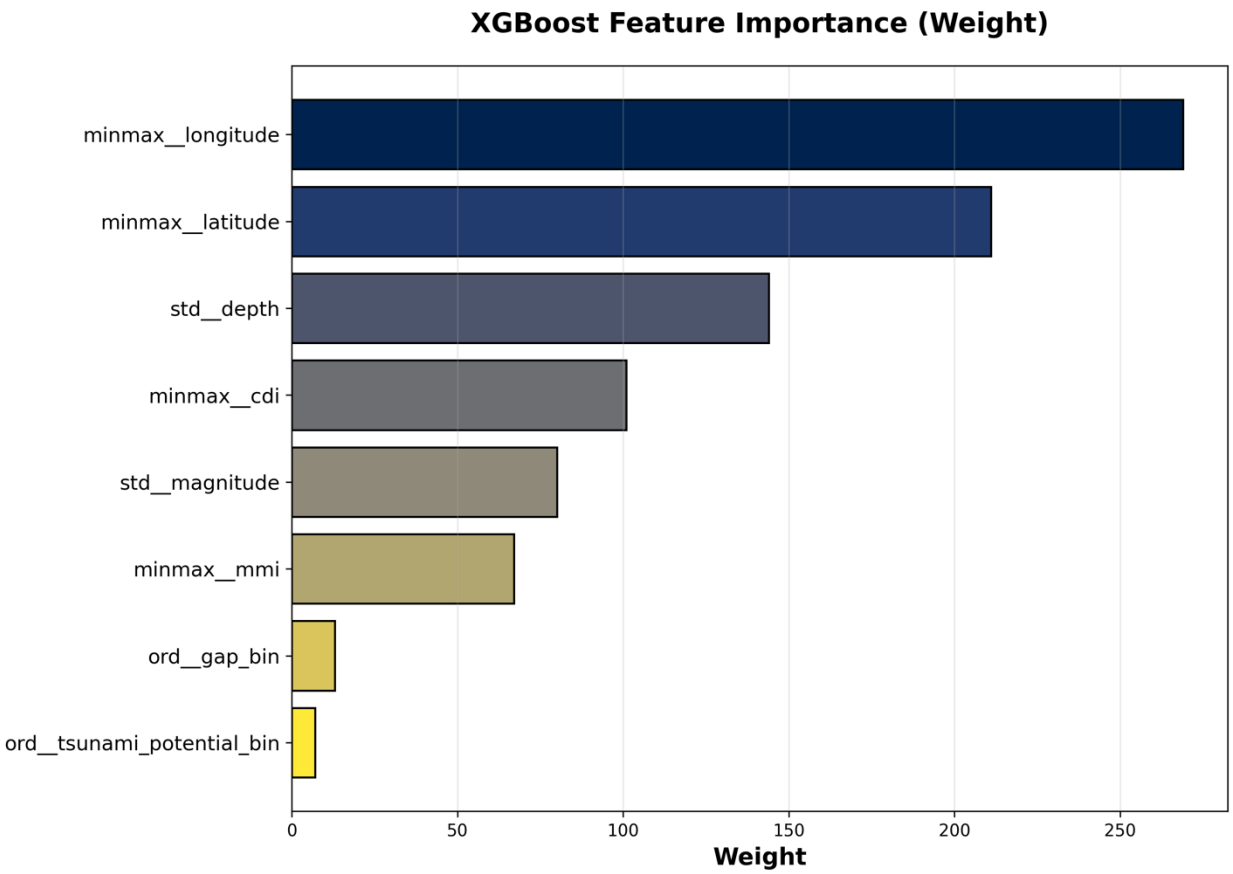

Figure 7: XGBoost Feature Importance (with Weight).

This plot shows XGBoost feature importance by split frequency: longitude and latitude dominate, indicating location drives many decisions, while depth and intensity/magnitude features (CDI, MMI, magnitude) matter moderately and the binned features contribute least.

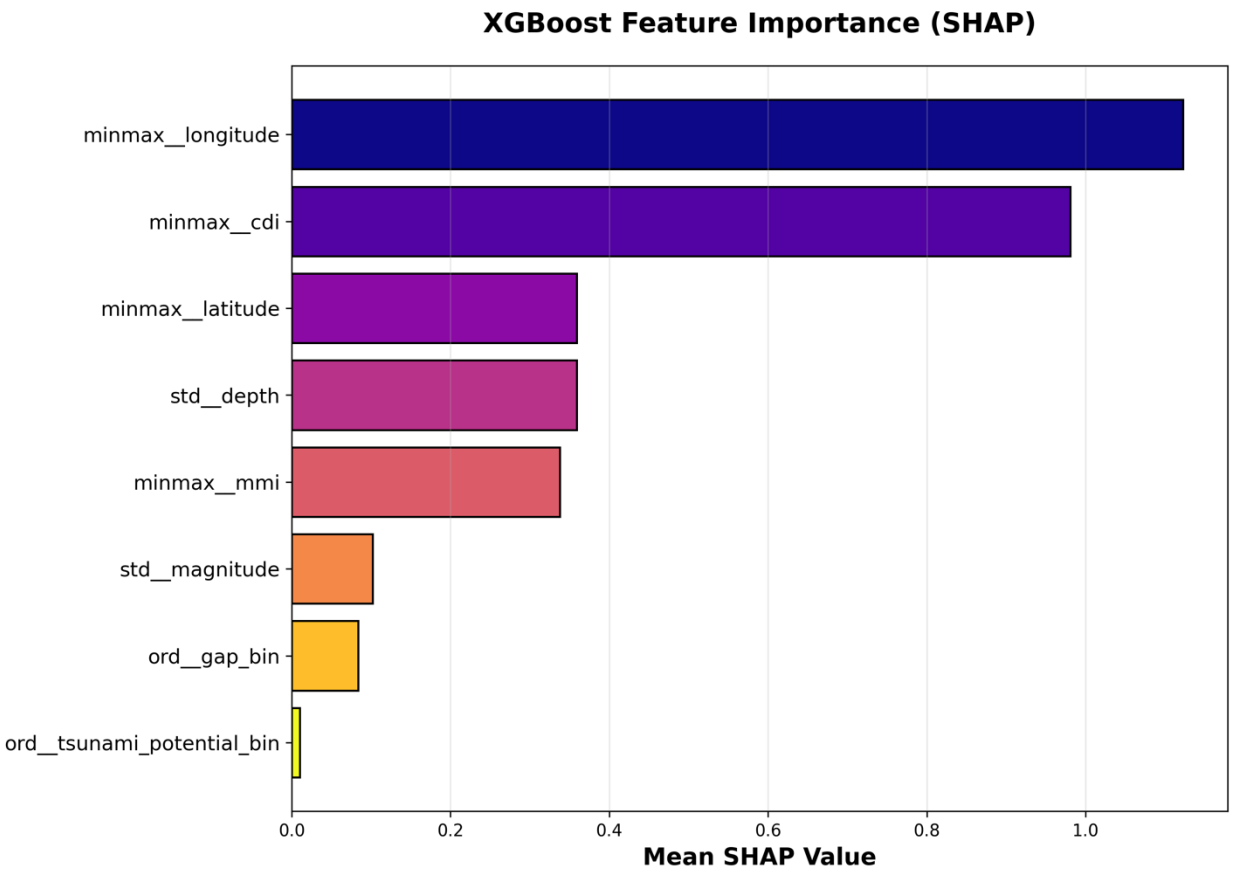

Figure 8: XGBoost Feature Importance (SHAP).

This SHAP importance plot shows longitude as the strongest driver of XGBoost predictions, followed by CDI, with latitude and depth also contributing meaningfully. MMI has moderate impact, while magnitude and the binned features (gap_bin, tsunami_potential_bin) contribute relatively little, suggesting predictions depend more on location and felt intensity than magnitude alone.

Local Feature Importance

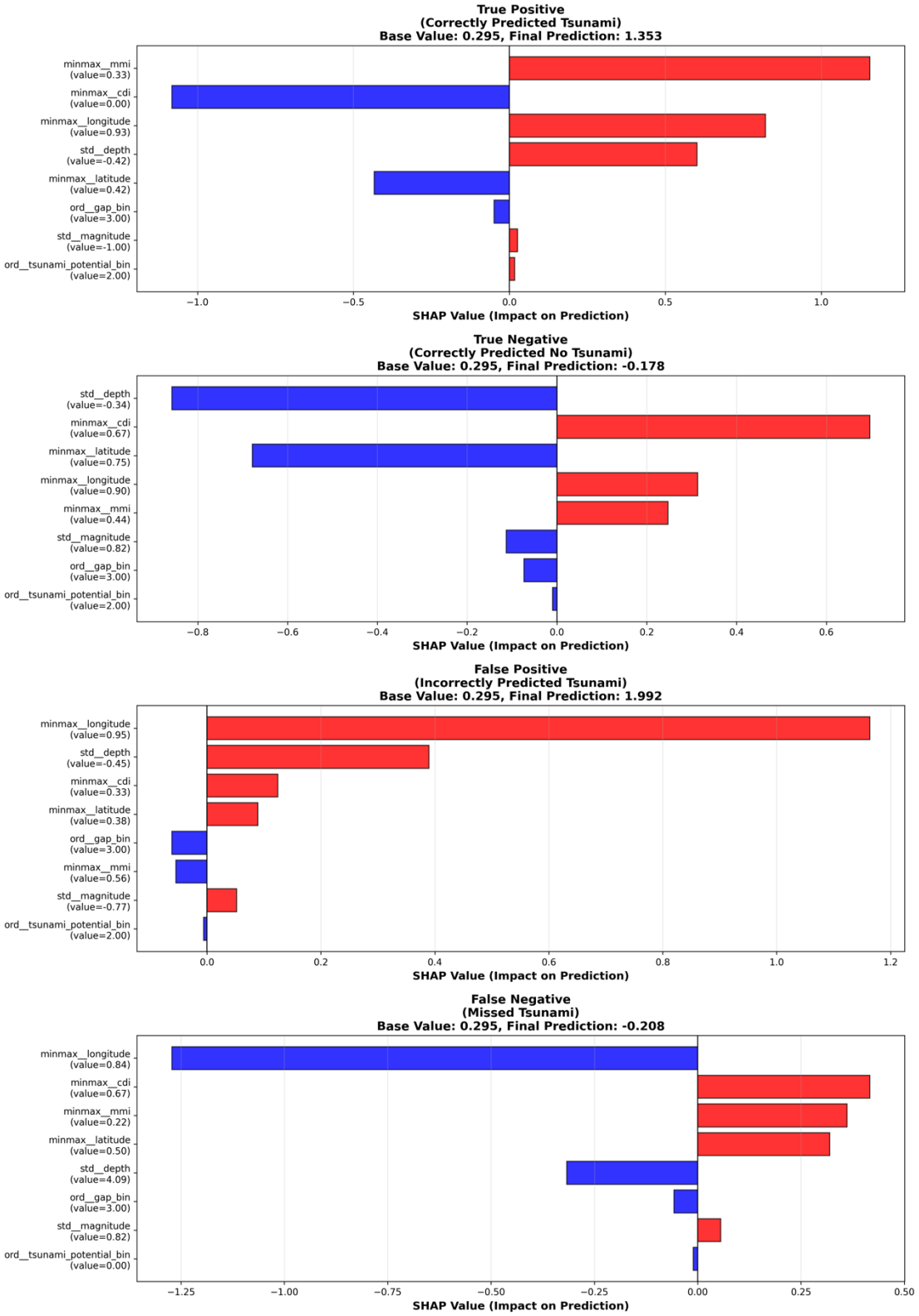

Figure 9: XGBoost Local Feature Importance.

This figure shows four SHAP local explanations (TP, TN, FP, FN), illustrating how features move each prediction from the base value to the final output. Red bars push the model toward “Tsunami,” blue bars push it away, and bar length indicates contribution strength for that event. Overall, longitude/latitude, CDI/MMI, and depth are the main drivers, and the FP/FN cases show how these signals can either over-trigger a tsunami prediction or be offset by opposing factors.

Key Interpretations:

- **Most Important:** cdi and mmi are the most influential features. These shaking intensity capture the felt impact and locations of earthquakes at the surface, which directly correlates with the vertical seafloor displacement necessary to generate tsunamis — earthquakes that are strongly felt at the surface are more likely to displace water.
- **Least Important:** Magnitude of the earthquake is one of the least important features. It’s interesting that the result corresponds with the figure 1 that many small magnitude but shallow earthquakes cause tsunami.

Outlook

While the XGBoost model demonstrates high performance, particularly with a Recall of 82.6% and an AUC of 0.882, several measures for future improvement exist to enhance its reliability as a disaster warning system.

- **Data Scarcity:** The study is limited by a small sample size (1,000 observations), which hinders generalizability and increases variance. Future research requires larger datasets (exceeding 100,000 earthquake-tsunami events) to ensure robust training and validation.
- **Precision Refinement:** Although the model effectively catches over 90% of tsunami events, the current Precision of around 60% indicates a significant number of false alarms. It should prioritize reducing these false positives through collecting more oceanic data and the fine-tuning of decision thresholds to move Precision toward a target of 75% or higher.

References

National Centers for Environmental Information. *Global Historical Tsunami Database, 2100 BC to Present*. National Oceanic and Atmospheric Administration, 2024, https://www.ncei.noaa.gov/access/metadata/landing-page/bin/iso?id=gov.noaa.ngdc.mgg.hazards:G01215. [3]

U.S. Geological Survey. *Earthquake Hazards Program: Technical Documentation*. U.S. Department of the Interior, 2024, https://www.usgs.gov/programs/earthquake-hazards/science/technical-documentation. [4]

Chen, Tianqi, and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System." *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794. *ACM Digital Library*, https://dl.acm.org/doi/10.1145/2939672.2939785. [5]

Pedregosa, Fabian, et al. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research*, vol. 12, 2011, pp. 2825-2830, https://scikit-learn.org/stable/about.html. [6]

Lundberg, Scott M., and Su-In Lee. "A Unified Approach to Interpreting Model Predictions." *Advances in Neural Information Processing Systems 30*, 2017, https://shap.readthedocs.io/en/latest/index.html. [7]