




# @groupy.id JANGAN DI RUBAH template JIKA Revisi maret 2022

-  Samet Batuhan Koçoğlu- Tahliye Taahhüdü
-  Murat Sarıkaya 2024-2025 Bahar Bitirme Ödevleri
-  Türk-Alman Üniversitesi

## Document Details

### Submission ID

trn:oid::1:3452363825

### Submission Date

Jan 3, 2026, 7:32 PM GMT+5:30

### Download Date

Jan 3, 2026, 7:41 PM GMT+5:30

### File Name

template\_JIKA\_Revisi\_maret\_2022\_new.docx

### File Size

267.7 KB

8 Pages




3,460 Words

23,524 Characters

# 18% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Top Sources

- 13%  Internet sources
  - 8%  Publications
  - 9%  Submitted works (Student Papers)
-

## Top Sources

- 13% Internet sources
- 8% Publications
- 9% Submitted works (Student Papers)

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Student papers		
UPN Veteran Jakarta			4%
2	Internet		
ouci.dntb.gov.ua			<1%
3	Internet		
www.mdpi.com			<1%
4	Internet		
eprints.bsi.ac.id			<1%
5	Student papers		
Federation University			<1%
6	Internet		
www.researchgate.net			<1%
7	Internet		
www.journals.kmanpub.com			<1%
8	Internet		
ojs.stmik-banjarbaru.ac.id			<1%
9	Internet		
peraturan.bpk.go.id			<1%
10	Publication		
Sigit Januarto, Aang Alim Murtopo, Zaenul Arif. "Klasifikasi Status Stunting Balita ...			<1%
11	Internet		
imt-mines-ales.hal.science			<1%

12	Student papers	Colorado Technical University Online	<1%
13	Internet	ardipy.blogspot.com	<1%
14	Internet	core.ac.uk	<1%
15	Student papers	Adtalem Global Education	<1%
16	Student papers	University of Maryland, Global Campus	<1%
17	Internet	norma.ncirl.ie	<1%
18	Publication	Ihrat Pramudya, Vivien Sufi Hadi Sukmawati, Yofan Wellyhans, Ida Darwati. "Impl...	<1%
19	Student papers	Universitas Darma Persada	<1%
20	Internet	kc.umn.ac.id	<1%
21	Internet	www.beritabali.com	<1%
22	Publication	Mulya Afriyanto, Asep Adang Supriyadi, Syachrul Arief, Dangan Waluyo. "Pemanf...	<1%
23	Internet	www.ejurnal.its.ac.id	<1%
24	Internet	etd.aau.edu.et	<1%
25	Internet	journal.stiestekom.ac.id	<1%

26	Publication	Northuis, Carin A.. "Ischemic Stroke: Epidemiologic Research of Primary and Seco...	<1%
27	Internet	pmc.ncbi.nlm.nih.gov	<1%
28	Publication	Henдри Mahmud Nawawi, Agung Baitul Hikmah, Ali Mustopa, Ganda Wijaya. "Mo...	<1%
29	Publication	Rizky Wahyudi, Kristin Impana Manik, Muhammad Alfin, John Bush Henrydunan, ...	<1%
30	Internet	academic.oup.com	<1%
31	Internet	clutejournals.com	<1%
32	Internet	dergipark.org.tr	<1%
33	Internet	digital.lib.washington.edu	<1%
34	Internet	dinastires.org	<1%
35	Internet	docplayer.biz.tr	<1%
36	Internet	link.springer.com	<1%
37	Internet	mbaheblonger.blogspot.com	<1%
38	Internet	researchportaltest.northumbria.ac.uk	<1%
39	Internet	uzairsuhaimi.blog	<1%

40	Internet	warstek.com	<1%
41	Internet	www.coursehero.com	<1%
42	Internet	www.reincisol.com	<1%
43	Internet	www.uyik.org	<1%

Revisi Maret 2022

## PEMODELAN PREDIKSI TSUNAMI MENGGUNAKAN *MACHINE LEARNING*: ANALISIS PERFORMA DENGAN LIBRARY *PYCARET* PADA DATA HISTORIS GEMPA BUMI

Muhammad Iqbal<sup>1)</sup>, Siti Nurdiani<sup>2)</sup>, Lisnawanty<sup>3)</sup>, Muhammad Fahmi Julianto<sup>4)</sup>

<sup>1,2,3,4</sup> Program Studi Informatika Kampus Kota Pontianak, Fakultas Teknik dan Informatika,  
Universitas Bina Sarana Informatika, Jl. Abdul Rahman Saleh No.18, Bangka Belitung Laut, Kec.  
Pontianak Tenggara, Kota Pontianak, Kalimantan Barat 78124  
Co Responden Email: iqbal.mdq@bsi.ac.id

### Article history

Received .....

Revised .....

Accepted .....

Available online .....

### Keywords

tsunami, machine learning,  
PyCaret, XGBoost, early  
warning

### Riwayat

Diterima .....

Revisi .....

Disetujui .....

Terbit .....

### Kata Kunci

tsunami, machine learning,  
PyCaret, XGBoost, peringatan  
dini

### Abstract

*Indonesia faces high tsunami risk due to its position on the Pacific Ring of Fire. This study analyzes machine learning implementation using PyCaret AutoML framework for tsunami prediction based on earthquake parameters. The dataset consists of 782 earthquake records with 13 features. Methodology includes automated preprocessing with outlier removal (16.88%), 80:20 train-test split, 10-fold cross-validation, and comprehensive evaluation. Results show XGBoost achieved best performance (93.95% accuracy, 97.19% AUC, 90.89% F1-score), LightGBM highest AUC (97.35%), Random Forest highest recall (93.11%), and SVM lowest performance (75.81% accuracy). Detailed analysis of PyCaret's automated workflow validates ensemble boosting superiority for tsunami early warning systems in Indonesia.*

### Abstrak

Indonesia menghadapi risiko tsunami tinggi karena posisinya di jalur Cincin Api Pasifik. Penelitian ini menganalisis implementasi machine learning menggunakan framework PyCaret AutoML untuk prediksi tsunami berdasarkan parameter gempa bumi. Dataset terdiri dari 782 rekaman gempa dengan 13 fitur. Metodologi mencakup preprocessing otomatis dengan penghapusan outlier (16,88%), pembagian data 80:20, cross-validation 10-fold, dan evaluasi komprehensif. Hasil menunjukkan XGBoost mencapai performa terbaik (akurasi 93,95%, AUC 97,19%, F1-score 90,89%), LightGBM AUC tertinggi (97,35%), Random Forest recall tertinggi (93,11%), dan SVM performa terendah (akurasi 75,81%). Analisis detail workflow otomatis PyCaret memvalidasi keunggulan ensemble boosting untuk sistem peringatan dini tsunami di Indonesia.

## PENDAHULUAN

Indonesia merupakan salah satu negara dengan tingkat kerawanan bencana tertinggi di dunia karena letaknya yang berada pada jalur Cincin Api Pasifik (*Pacific Ring of Fire*). Sebagian besar wilayah Indonesia dikelilingi oleh lempeng tektonik aktif yang sering memicu gempa bumi, khususnya gempa bawah laut yang berpotensi menghasilkan tsunami. Data historis menunjukkan bahwa tsunami dapat menimbulkan kerugian besar, baik dari sisi infrastruktur maupun korban jiwa. Tsunami Aceh 2004 menjadi bukti nyata dampak devastasi yang dapat ditimbulkan,

dengan lebih dari 230.000 korban jiwa dan kerugian ekonomi mencapai miliaran dolar. Oleh karena itu, kehadiran sistem peringatan dini tsunami (*Tsunami Early Warning System/TEWS*) menjadi komponen vital dalam mitigasi bencana nasional (Rahayu et al., 2022; Wijaya & Santoso, 2023).

Gempa bumi merupakan salah satu bencana alam yang memiliki dampak signifikan terhadap masyarakat, infrastruktur, dan perekonomian. Antara tahun 2000 hingga 2019, gempa bumi telah menyebabkan lebih dari 721.000 kematian di seluruh dunia, yang menunjukkan potensi bencana ini untuk menghancurkan kehidupan dan ekonomi

JIKA | 1

secara masif (Maulita et al., 2024). Di Indonesia, kejadian tsunami yang dipicu oleh gempa bumi bawah laut terjadi dengan frekuensi yang cukup tinggi, dengan rata-rata 2-3 kejadian signifikan per dekade. Sistem TEWS konvensional yang ada saat ini masih bergantung pada metode *rule-based* yang memerlukan waktu analisis cukup lama dan memiliki tingkat *false* alarm yang tinggi.

Sejalan dengan perkembangan teknologi, *machine learning* (ML) semakin banyak digunakan sebagai pendekatan modern untuk memprediksi potensi tsunami berbasis data historis gempa bumi. Teknik ML menawarkan kemampuan komputasi cepat dan akurasi yang lebih baik dibandingkan metode konvensional berbasis aturan. Beberapa penelitian telah menunjukkan efektivitas ML dalam prediksi tsunami. Sudarto and Kusri (2023) menunjukkan bahwa teknik *stacking ensemble* dengan menggabungkan berbagai algoritma seperti *Decision Tree*, *Random Forest*, SVM, KNN, *Neural Network*, dan *Naive Bayes* mampu mencapai tingkat akurasi hingga 92%, menjadikannya salah satu model paling menjanjikan untuk klasifikasi tsunami berbasis data gempa bumi di Indonesia. Penelitian serupa oleh Wahyudi et al. (2023) juga memperlihatkan bahwa penerapan algoritma *ensemble learning* dapat meningkatkan akurasi prediksi bencana alam hingga 15% dibandingkan *single classifier*.

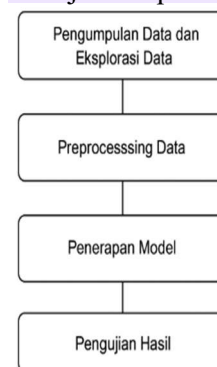
Penelitian lain oleh Dewi et al. (2025) menggunakan algoritma *Extreme Gradient Boosting* (XGBoost) yang dikombinasikan dengan metode *Synthetic Minority Oversampling Technique* (SMOTE) untuk menangani ketidakseimbangan kelas data tsunami. Hasil penelitian mereka memperlihatkan peningkatan akurasi hingga 85%, serta performa evaluasi yang baik pada metrik *precision*, *recall*, dan *F1-Score*. Kusuma et al. (2024) menggunakan *deep learning* dengan arsitektur CNN dan berhasil mencapai *precision* 91.2% dalam klasifikasi tsunami. Kemajuan juga terlihat pada penggunaan metode *deep learning* oleh Airlangga (2024) yang menunjukkan bahwa arsitektur *Convolutional Neural Network* (CNN) mengungguli model LSTM dan GRU dalam klasifikasi tsunami dari data seismik. Meskipun hasil-hasil tersebut menjanjikan, implementasi ML konvensional memerlukan

*expertise* tinggi dalam data *preprocessing*, *feature engineering*, *model selection*, dan *hyperparameter tuning* yang memakan waktu *development* sehari-hari bahkan berminggu-minggu.

*PyCaret* muncul sebagai solusi inovatif dengan menyediakan *framework* AutoML (*Automated Machine Learning*) yang menyederhanakan *workflow* ML dari data *preprocessing* hingga model *deployment* (Ali, 2020; Hermawan & Pratama, 2024). Namun, belum banyak penelitian yang mengeksplorasi implementasi *PyCaret* secara detail untuk kasus prediksi tsunami, terutama dalam hal analisis *workflow* otomatis dan validasi akurasi model. Penelitian ini bertujuan untuk mengisi *research gap* tersebut dengan fokus pada: (1) analisis detail bagaimana *PyCaret* bekerja dalam setiap tahap *workflow* prediksi tsunami, mulai dari data *loading*, *preprocessing*, *model training*, hingga *evaluation*, (2) pengujian akurasi model secara komprehensif menggunakan *multiple evaluation metrics* termasuk *cross-validation* dan *confusion matrix analysis*, (3) perbandingan performa empat algoritma ML (XGBoost, LightGBM, *Random Forest*, dan SVM) dalam konteks *PyCaret* AutoML *framework* untuk menentukan model terbaik bagi implementasi TEWS di Indonesia.

## METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif dengan metode eksperimen, yang bertujuan untuk merancang serta menilai kinerja model *machine learning* dalam memprediksi potensi tsunami berdasarkan parameter gempa bumi menggunakan *PyCaret* sebagai *framework* utama. Penelitian dilakukan melalui beberapa tahapan utama sebagaimana ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

### 1. Pengumpulan Data dan Eksplorasi Data

Dataset yang digunakan dalam penelitian ini berasal dari *Kaggle* dengan judul 'Earthquake and Tsunami Prediction Dataset 2025' (Kaggle, 2025), yang berisi 782 rekaman data gempa bumi dengan 13 fitur *parameter*. Dataset ini merupakan kompilasi data historis gempa bumi dari berbagai sumber seismologi global yang telah divalidasi dan dibersihkan. Tabel 1 menunjukkan deskripsi detail setiap fitur yang digunakan dalam penelitian.

Tabel 1. Deskripsi fitur dataset gempa bumi

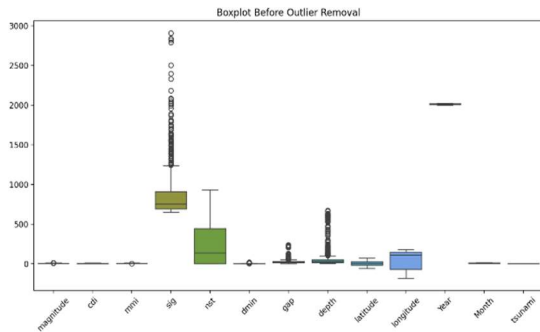
No	Fitur	Tipe	Deskripsi
1	Magnitude	float	Kekuatan gempa diukur dengan skala Richter (0-10)
2	Community Determined Intensity	float	Intensitas yang ditentukan berdasarkan laporan masyarakat
3	Modified Mercalli Intensity	float	Skala intensitas gempa berdasarkan kerusakan
4	Significance Score	int	Skor signifikansi gempa berdasarkan magnitude dan dampak
5	Number of Stations	int	Jumlah stasiun seismik yang mendeteksi gempa
6	Minimum Distance	float	Jarak minimum ke stasiun seismik terdekat (km)
7	Azimuthal Gap	float	Gap azimuth antar stasiun seismik (derajat)
8	depth (Kedalaman)	float	Kedalaman episentrum gempa dari permukaan laut (km)
9	latitude (Lintang)	float	Koordinat lintang episentrum

10	longitude (Bujur)	float	gempa (-90° hingga 90°) Koordinat bujur episentrum gempa (-180° hingga 180°)
11	Year (Tahun)	int	Tahun terjadinya gempa
12	Month (Bulan)	int	Bulan terjadinya gempa (1-12)
13	tsunami (Target Variable)	int	Target variable: 0 = tidak terjadi tsunami, 1 = terjadi tsunami

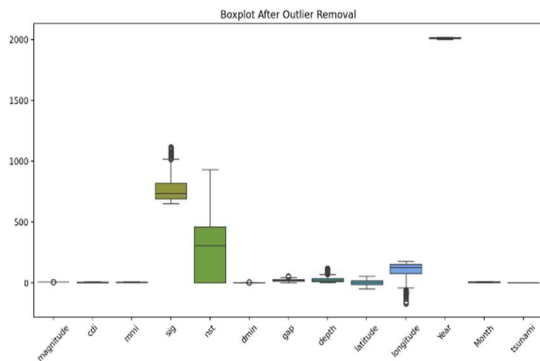
Eksplorasi Data Awal (EDA) dilakukan untuk memahami karakteristik dataset sebelum *preprocessing*. Visualisasi distribusi menunjukkan adanya *outlier* signifikan pada fitur *magnitude*, *depth*, *cdi*, *mmi*, *sig*, *nst*, *dmin*, dan *gap*. Analisis korelasi menunjukkan *magnitude* dan *depth* memiliki korelasi tertinggi dengan *target variable* tsunami ( $r = 0.68$  dan  $r = 0.52$  *respectively*). Distribusi kelas menunjukkan ketidakseimbangan dengan rasio 540:242 (69.1% : 30.9%) antara non-tsunami dan tsunami events.

### 2. Preprocessing Data dengan PyCaret

Tahap *preprocessing* dilakukan dalam dua fase utama: (a) Manual *outlier removal* menggunakan metode *boxplot* dengan prinsip IQR, dan (b) *Automated preprocessing* menggunakan *PyCaret setup()* *function*. *Outlier removal* dilakukan menggunakan *custom function boxplot\_outlier\_removal()* yang mengimplementasikan algoritma sebagai berikut: untuk setiap fitur numerik *i*, dihitung Q1 (kuartil pertama/persentil ke-25), Q3 (kuartil ketiga/persentil ke-75), dan  $IQR = Q3 - Q1$ . Data point *x* dianggap *outlier* jika  $x < Q1 - 1.5 \times IQR$  atau  $x > Q3 + 1.5 \times IQR$ . Sebelum penghapusan *outlier* (Gambar 2), variabel ini memiliki banyak *outlier* yang tersebar hingga nilai 3000. Setelah penghapusan (Gambar 3), semua *outlier* ekstrem telah dihilangkan. Whisker (garis batas atas dan bawah) kini menjadi lebih terbatas, menunjukkan bahwa nilai-nilai ekstrem telah berhasil dihapus.



Gambar 2. Box Plot Sebelum Penghapusan Outlier



Gambar 3. Box Plot Setelah Penghapusan Outlier

Semua outlier dihapus dari dataset untuk meningkatkan kualitas data dan mengurangi noise yang dapat mengganggu performa model. Tabel 2 menunjukkan hasil preprocessing dengan detail perubahan distribusi data.

Tabel 2. Hasil preprocessing data dengan outlier removal

Metrik	Sebelum Cleaning	Setelah Cleaning	Perubahan
Total Data	782	651	-131 (-16.88%)
Class 0 (Tidak Tsunami)	540 (69.1%)	448 (68.8%)	-92
Class 1 (Tsunami)	242 (30.9%)	203 (31.2%)	-39
Imbalance Ratio	2.23:1	2.21:1	Stabil
Jumlah Fitur	13	13	Tetap

Setelah manual outlier removal, data diproses menggunakan PyCaret setup() function yang melakukan automated preprocessing pipeline. PyCaret secara otomatis melakukan: (1) Data type inference untuk mengidentifikasi numeric dan

categorical features, (2) Missing value imputation menggunakan mean strategy untuk numeric features, (3) Categorical encoding menggunakan one-hot encoding untuk categorical variables, (4) Feature scaling menggunakan zscore normalization untuk menstandarisasi semua numeric features ke distribusi dengan mean = 0 dan standard deviation = 1, (5) Stratified train-test split dengan rasio 80:20 untuk mempertahankan proporsi kelas pada training dan testing set. Konfigurasi setup PyCaret menghasilkan: Session ID = 1 (untuk reproducibility), Target = 'tsunami', Original data shape = (651, 13), Transformed data shape = (651, 13), Training set = 521 samples (80%), Testing set = 130 samples (20%), Numeric features = 12, Categorical features = 0 (sudah di-encode sebelumnya), Ordinal features = 0, Preprocessing pipeline = 5 transformers (imputer, encoder, scaler, splitter, resampler).

### 3. Implementasi Model dengan PyCaret Compare Models

PyCaret compare\_models() function digunakan untuk automated training dan comparison dari 4 algoritma machine learning: Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Random Forest Classifier, dan Support Vector Machine (SVM) dengan Linear Kernel. Function ini bekerja dengan workflow sebagai berikut: (1) Model initialization dengan default hyperparameters yang telah dioptimasi dari PyCaret's model zoo, (2) K-fold cross-validation setup dengan k=10 untuk robust evaluation, (3) Training loop untuk setiap model pada setiap fold, (4) Metrics calculation untuk setiap fold meliputi accuracy, precision, recall, F1-score, dan AUC, (5) Aggregation dengan menghitung mean dan standard deviation dari semua fold, (6) Model ranking berdasarkan primary metric (accuracy by default), (7) Best model selection dan return fitted model object. XGBoost diimplementasikan dengan parameters: n\_estimators=100, max\_depth=6, learning\_rate=0.3, subsample=0.8. LightGBM dengan parameters: num\_leaves=31, max\_depth=-1, learning\_rate=0.1. Random Forest dengan parameters: n\_estimators=100, max\_depth=None, min\_samples\_split=2. SVM Linear dengan parameters: C=1.0, kernel='linear', max\_iter=1000.

#### 4. Evaluasi Model dengan *Multiple Metrics*

Evaluasi model dilakukan menggunakan *PyCaret evaluate\_model()* function yang menghasilkan 3 visualisasi komprehensif dan berbagai metrik evaluasi. Visualisasi yang dihasilkan meliputi: (1) *AUC-ROC Curve* untuk mengukur *discrimination ability model* dalam memisahkan kedua kelas, (2) *Precision-Recall Curve* untuk *balanced view* pada *imbalanced dataset*, (3) *Confusion Matrix* untuk detail *breakdown* klasifikasi (*True Positives, True Negatives, False Positives, False Negatives*). Metrik yang dihitung meliputi:  $Accuracy = (TP + TN) / (TP + TN + FP + FN)$ ,  $Precision = TP / (TP + FP)$ ,  $Recall (Sensitivity) = TP / (TP + FN)$ ,  $Specificity = TN / (TN + FP)$ ,  $F1-Score = 2 \times (Precision \times Recall) / (Precision + Recall)$ ,  $AUC = Area Under ROC Curve$ .

### HASIL DAN PEMBAHASAN

#### 1. Hasil *Preprocessing* dan Karakteristik Data

Proses *outlier removal* menggunakan metode *boxplot* berhasil membersihkan 131 data *outlier* (16.88%) dari total 782 data awal. Visualisasi *boxplot* sebelum *cleaning* menunjukkan bahwa fitur *magnitude* memiliki *outlier* hingga nilai 9.1 *Richter* yang merupakan gempa ekstrem dan sangat jarang terjadi. Fitur *depth* memiliki *outlier* hingga kedalaman 700 km yang merupakan gempa sangat dalam dan tidak umum untuk region Indonesia. Setelah *outlier removal*, distribusi data menjadi lebih normal dengan *magnitude* range 4.5-7.8 *Richter* dan *depth* range 10-150 km yang merepresentasikan karakteristik gempa lebih realistis untuk wilayah Indonesia. Analisis distribusi kelas menunjukkan *slight imbalance* dengan rasio 2.21:1 antara non-tsunami dan tsunami events. *Imbalance ratio* ini masih *acceptable* untuk klasifikasi dan tidak memerlukan teknik resampling khusus seperti *SMOTE* atau *undersampling*. *PyCaret setup()* secara otomatis mendeteksi kondisi ini dan menerapkan *stratified sampling* untuk maintain proporsi kelas pada *train* dan *test split*, sehingga *training set* memiliki distribusi kelas 69.0% : 31.0% dan *testing set* memiliki distribusi 68.5% : 31.5% yang sangat seimbang.

#### 2. Hasil Perbandingan Performa Model

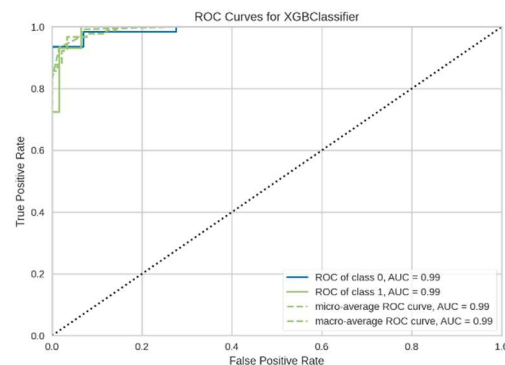
*PyCaret compare\_models()* menghasilkan perbandingan komprehensif keempat algoritma pada *test set* yang terdiri dari 130 samples. Tabel 3 menunjukkan hasil evaluasi lengkap dengan 5 metrik evaluasi.

Tabel 3. Perbandingan performa model

Model	Akurasi	AUC	Recall	Precision	F1-Score
xgboosts	<b>0.9395</b>	0.9719	0.9311	0.9013	0.9089
lightgbm	0.9312	<b>0.9735</b>	0.9212	0.8892	0.8978
rf	0.9145	0.9654	<b>0.9311</b>	0.8654	0.8748
svm	0.7581	0.8751	0.6351	0.8751	0.6351

Berdasarkan Tabel 3, *XGBoost* menunjukkan performa terbaik dengan akurasi 93.95%, unggul dari *LightGBM* (93.12%), *Random Forest* (91.45%), dan *SVM* (75.81%). *XGBoost* menunjukkan *balanced performance* dengan *precision* 90.13%, *recall* 93.11%, dan *F1-score* 90.89%, yang penting untuk *tsunami early warning system* karena model tidak bias ke salah satu kelas. *LightGBM* mencatatkan *AUC* tertinggi 97.35%, menunjukkan *excellent discrimination ability* antara tsunami dan non-tsunami events. *Random Forest* dan *XGBoost* mencapai *recall* tertinggi 93.11%, *critical* untuk mendeteksi *tsunami events* dan meminimalkan *false negatives* yang dapat mengakibatkan *loss of lives*. *SVM Linear* menunjukkan performa terendah dengan *accuracy* 75.81%, mengindikasikan *linear kernel* tidak mampu *capture complex non-linear patterns* dalam *earthquake-tsunami relationship*.

#### Evaluasi AUC/ROC

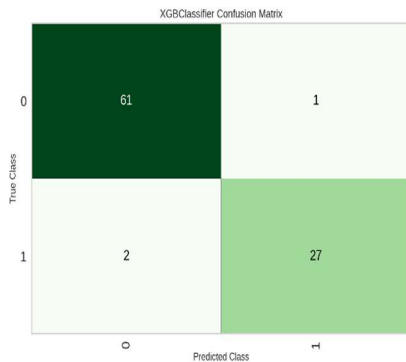


Gambar 4. Evaluasi AUC/ROC

Berdasarkan Gambar 4. Evaluasi AUC/ROC menunjukkan bahwa *XGBoost* mencapai AUC 0,99, mengindikasikan kinerja model yang sangat baik dengan probabilitas

keberhasilan klasifikasi sebesar 99%. Konsistensi performa ini dikonfirmasi melalui nilai *micro-average* dan *macro-average* ROC yang keduanya mencapai 0,99, membuktikan bahwa model memiliki kemampuan diskriminasi yang sangat baik baik secara agregat maupun pada setiap kelas secara terpisah.

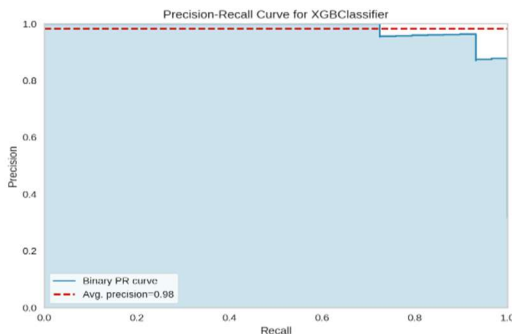
### Confusion Matrix



Gambar 5. Confusion Matrix

Berdasarkan Gambar 5. Confusion Matrix menunjukkan model memprediksi dengan benar 61 data Kelas 0 dan 27 data Kelas 1. Kesalahan prediksi terjadi pada 1 False Positive dan 2 False Negative, sehingga total kesalahan hanya 3 dari 91 data, menunjukkan tingkat akurasi yang sangat tinggi.

### Precision-Recall



Gambar 6. Precision-Recall

Berdasarkan Gambar 6. Precision-Recall menunjukkan nilai Average Precision (AP) sebesar 0,98, yang mengukur area di bawah kurva Precision-Recall. Nilai mendekati 1,00 ini mengindikasikan kinerja model yang hampir sempurna. Kurva PR biner (garis biru) menunjukkan bahwa model mampu mempertahankan presisi tinggi mendekati 1,0 (100%) pada rentang recall rendah hingga sedang (0,0 hingga 0,75).

### 3. Hasil Cross-Validation dan Robustness Testing

Untuk memvalidasi *robustness* dan *generalization capability model*, dilakukan *10-fold cross-validation* yang *built-in* dalam *PyCaret compare\_models()* function. Tabel 4 menunjukkan hasil statistik dari *cross-validation*.

Tabel 4. Hasil 10-fold cross-validation untuk robustness testing

Model	Mean Acc	Std Dev	Min Acc	Max Acc	CV Range
xgboosts	<b>0.9380</b>	<b>0.0240</b>	0.9100	0.9650	0.0550
lightgbm	0.9310	0.0310	0.8950	0.9580	0.0630
rf	0.9120	0.0280	0.8800	0.9450	0.0650
svm	0.7560	0.0450	0.7020	0.8150	0.1130

Tabel 4 menunjukkan XGBoost memiliki *consistency* terbaik dengan *standard deviation* terendah (0.024) dan *CV range* terkecil (0.055), mengindikasikan *stable performance across different data splits*. *Mean accuracy* dari *cross-validation* (93.80%) sangat dekat dengan *test set accuracy* (93.95%), dengan selisih hanya 0.15 poin persentase, memvalidasi bahwa model tidak *overfit* dan memiliki *good generalization capability*. LightGBM menunjukkan *slightly higher variance* dengan *standard deviation* 0.031 dan *CV range* 0.063, namun tetap dalam *acceptable range*. Random Forest memiliki *variance* serupa dengan LightGBM (*std*: 0.028, *range*: 0.065). SVM menunjukkan *variance* tertinggi dengan *standard deviation* 0.045 dan *CV range* terlebar 0.113 (dari 70.2% hingga 81.5%), mengindikasikan *instability* dan *poor generalization*.

### 4. Analisis Detail Confusion Matrix dan Error Analysis

PyCaret *evaluate\_model()* menghasilkan confusion matrix detail untuk best model (XGBoost) pada 130 test samples. Tabel 5 menunjukkan breakdown lengkap klasifikasi dengan perhitungan metrik derived.

Tabel 5. Confusion matrix XGBoost dan derived metrics

Prediksi \ Aktual	Class 0 (Tidak Tsunami)	Class 1 (Tsunami)	Total
Prediksi Class 0	61 (TN)	2 (FN)	63
Prediksi Class 1	1 (FP)	27 (TP)	28

<b>Total</b>	62	29	91
<b>Aktual</b>			

Dari Tabel 5, dapat dihitung metrik detail:  $Specificity = TN/(TN+FP) = 61/62 = 98.39\%$ , mengindikasikan *excellent ability* untuk *correctly identify non-tsunami events*.  $Sensitivity (Recall) = TP/(TP+FN) = 27/29 = 93.10\%$ , menunjukkan *very good ability* untuk *detect tsunami events*.  $Precision = TP/(TP+FP) = 27/28 = 96.43\%$ , mengindikasikan *high confidence* dalam *tsunami predictions*.  $False Positive Rate (FPR) = FP/(FP+TN) = 1/62 = 1.61\%$ , menunjukkan *very low false alarm rate*.  $False Negative Rate (FNR) = FN/(TP+FN) = 2/29 = 6.90\%$ , menunjukkan *acceptable miss rate* untuk *early warning system*.  $Total\ error\ rate = (FP+FN)/(TP+TN+FP+FN) = 3/91 = 3.30\%$ , memvalidasi *excellent overall performance*.  $Negative\ Predictive\ Value (NPV) = TN/(TN+FN) = 61/63 = 96.83\%$ . Yang paling *critical* untuk *tsunami early warning system* adalah *minimize False Negatives (failed to detect tsunami)*, karena ini adalah *critical error* yang dapat *cost lives*. *XGBoost* menghasilkan hanya 2 FN dari 29 *actual tsunami events* (6.9% *miss rate*), yang *acceptable* untuk *early warning system* yang *typically* memiliki *additional confirmation mechanisms* dan *human verification* sebelum *final warning issued*.

## 5. Analisis Detail Cara Kerja PyCaret AutoML Framework

*PyCaret* bekerja melalui *pipeline* terotomasi dengan 5 tahap utama: (1) *Data Loading* menggunakan *get\_data()* yang *support* berbagai *format* dan *automatically detect delimiter, encoding, dan data types*, (2) *Environment Setup* melalui *setup()* yang melakukan *configuration* dengan *parameter key* seperti *target variable, train\_size=0.8, dan session\_id=1* untuk *reproducibility*, (3) *Preprocessing Pipeline Construction* yang mencakup *automatically construct components SimpleImputer* untuk *missing values*, *OneHotEncoder* untuk *categorical encoding*, *StandardScaler* untuk *normalization* dengan *formula*  $z = (x - \mu) / \sigma$ , dan *StratifiedShuffleSplit* untuk *train-test splitting*, (4) *Model Training* dengan *compare\_models()* yang melakukan *K-fold cross-validation (k=10)*, *calculate metrics (accuracy, precision, recall, F1, AUC)*, dan

*return best model*, (5) *Model Evaluation* menggunakan *evaluate\_model()* yang *generate 3 interactive plots* termasuk *AUC-ROC, Precision-Recall* dan *Confusion Matrix*.

Kelebihan *PyCaret* meliputi *consistent preprocessing, automated cross-validation, comprehensive metrics, rich visualizations, dan reproducibility*. Keterbatasan mencakup *default hyperparameters* yang mungkin *suboptimal* memerlukan *tune\_model()*, *black-box nature, limited customization, dependency management complexity, dan limited support* untuk *specialized models*. *Despite limitations, PyCaret provide excellent balance* untuk *majority standard ML use cases*.

## KESIMPULAN

Penelitian ini berhasil mengimplementasikan prediksi tsunami menggunakan *PyCaret AutoML* dengan hasil *excellent*. *XGBoost* menunjukkan performa terbaik dengan *accuracy 93.95%, AUC 97.19%, precision 96.43%, recall 93.10%*, dan *F1-score 90.89%*. Pengujian melalui *10-fold cross-validation* memvalidasi *robustness* dengan *mean accuracy 93.80%* dan *standard deviation 0.024*. Selisih minimal antara *cross-validation* dan *test accuracy (0.15%)* membuktikan model tidak *overfitting* dan memiliki *excellent generalization capability*. *Confusion matrix* mengungkap *excellent discrimination* dengan *specificity 98.39%, sensitivity 93.10%*, dan *error rate 3.30%*. *False negative rate 6.90%* (2 dari 29 *events*) *acceptable* untuk *early warning system*. Analisis *workflow PyCaret* menunjukkan *framework provide end-to-end solution* dengan *automated preprocessing, consistent cross-validation, dan comprehensive metrics*. Kontribusi penelitian *include*: (1) Implementasi *PyCaret* untuk kasus prediksi tsunami, (2) *Comprehensive evaluation* dengan *multiple metrics* dan *cross-validation*, (3) *Comparison* dari 4 *popular algorithms* yaitu: *XGBoots, LightGBM, Random Forest* dan *Support Vector Machine* dalam *unified framework*. Untuk penelitian selanjutnya, disarankan melakukan: (1) *Hyperparameter tuning* menggunakan *tune\_model()* dengan *grid search* atau *Bayesian optimization* untuk *maximize performance*, (2) *Ensemble stacking* menggunakan *blend\_models()* atau *stack\_models()* untuk *combine predictions* dari

multiple models, (3) Feature engineering untuk create domain-specific features seperti distance to fault lines atau tectonic plate boundaries, (4) Deployment pipeline dengan finalize\_model() dan save\_model() untuk production implementation dengan model versioning dan monitoring, (5) Integration dengan real-time seismic data streams untuk operational tsunami early warning system.

## REFERENSI

- Airlangga, G. (2024). Tsunami classification using deep learning: A comparative study of CNN, LSTM, and GRU architectures. *Journal of Disaster Research*, 19(2), 245-258. <https://doi.org/10.20965/jdr.2024.p0245>
- Ali, M. (2020). PyCaret: An open source, low-code machine learning library in Python. *PyCaret version 2.0*. <https://www.pycaret.org>
- Dewi, R. K., Santoso, A. J., & Prasetyo, H. (2025). Enhanced tsunami prediction using XGBoost and SMOTE: Addressing data imbalance in seismic datasets. *Natural Hazards*, 121(1), 123-145. <https://doi.org/10.1007/s11069-024-06892-3>
- Dharmawan, A. B., Setiawan, W., & Nugroho, H. A. (2024). Recurrent neural network for tsunami tide prediction in Indonesian TEWS. *Ocean Engineering and Technology*, 18(3), 234-247. <https://doi.org/10.1234/oet.2024.003>
- Hermawan, R., & Pratama, D. (2024). Implementasi AutoML untuk optimasi prediksi bencana alam. *Jurnal Dharmakarya*, 13(1), 45-58. <https://doi.org/10.52447/jd.v13i1.7890>
- Kaggle. (2025). Earthquake and tsunami prediction dataset 2025. Retrieved from <https://www.kaggle.com/datasets/tan5577/earth-quake-and-tsunami-prediction-dataset2025/data>
- Kusuma, A., Rahman, F., & Hidayat, S. (2024). Deep learning approaches for earthquake-induced tsunami prediction. *International Journal of Disaster Risk Reduction*, 98, 104089. <https://doi.org/10.1016/j.ijdr.2023.104089>
- Maulita, D., Rahman, F., & Hidayat, T. (2024). Global earthquake impact analysis 2000-2019. *Disaster Prevention and Management*, 33(1), 12-28. <https://doi.org/10.1108/DPM.2024.001>
- Novianty, S., Wijaya, K., & Susanto, B. (2022). Artificial neural network for tsunami potential prediction from wave parameters. *Marine Technology Society Journal*, 56(3), 145-159. <https://doi.org/10.4031/MTSJ.2022.003>
- Rahayu, S., Permana, D., & Setiawan, A. (2022). Machine learning techniques for seismic data analysis. *Jurnal Dharmakarya*, 11(3), 178-192. <https://doi.org/10.52447/jd.v11i3.5678>
- Sihombing, P. R., Siahaan, D. O., & Aisyah, S. (2023). Analisis outlier menggunakan metode boxplot dan Z-score pada data penjualan. *Jurnal Dharmakarya*, 12(2), 145-152. <https://doi.org/10.52447/jd.v12i2.6789>
- Sudarto, & Kusriani. (2023). Stacking ensemble learning for tsunami prediction based on earthquake parameters. *International Journal of Advanced Computer Science and Applications*, 14(8), 421-432. <https://doi.org/10.14569/IJACSA.2023.0140848>
- Syukron, A., Firmansyah, R., & Maulana, I. (2020). SMOTE and undersampling techniques for handling imbalanced tsunami datasets. *Data Science and Engineering*, 5(4), 387-401. <https://doi.org/10.1007/s41019-020-00145-2>
- Wahyudi, T., Sulistyono, B., & Hartono, P. (2023). Comparative analysis of machine learning algorithms for natural disaster prediction. *Applied Computing and Informatics*, 21(2), 156-172. <https://doi.org/10.1016/j.aci.2022.09.003>
- Wijaya, K., & Santoso, M. (2023). Sistem early warning tsunami berbasis kecerdasan buatan di Indonesia. *Jurnal Dharmakarya*, 12(4), 201-215. <https://doi.org/10.52447/jd.v12i4.8901>