

ANALISIS PERAN FEATURE ENGINEERING PADA KINERJA MODEL MACHINE LEARNING UNTUK KLASIFIKASI POTENSI TSUNAMI

Rizki Darmawan^{1*}, Immanuel Vico Yut², Alvin Hernando³, Rani Irma Handayani⁴, Risca Lusiana Pratiwi⁵, Euis Widanengsih⁶

^{1,2,3,6}Universitas Bina Sarana Informatika; Jl. Kramat Raya No. 98, RT.2/RW.9, Kwitang, Kec. Senen, Jakarta Pusat, 10450; (021)8000063

^{4,5}Universitas Nusa Mandiri; Jl. Jatiwaringin No.2, Cipinang Melayu, Jakarta Timur; (021)8005722

Keywords:

Tsunami;
Machine Learning;
Feature Engineering;
Klasifikasi;

Correspondent Email:

15230820@bsi.ac.id

Abstrak. Indonesia memiliki risiko bencana tsunami yang tinggi akibat lokasinya di Cincin Api Pasifik, sehingga menuntut adanya sistem peringatan dini yang akurat. Penelitian ini mengusulkan model klasifikasi potensi tsunami yang valid secara metodologis. Berbeda dengan penelitian sebelumnya yang sering menggunakan data "bocor" (leaked features), penelitian ini menerapkan strategi feature engineering mendalam yang menggabungkan fitur fisis (seperti `is_shallow`) dan fitur kualitas data (`gap`, `nst`) pada dataset Cincin Api Pasifik. Lima model *machine learning* (Naive Bayes, Logistic Regression, SVM, Random Forest, XGBoost) dievaluasi menggunakan 10-Fold Cross-Validation. Hasil penelitian menunjukkan feature engineering adalah langkah paling krusial, meningkatkan F1-Score dari ~0.38 menjadi 0.86. Random Forest teridentifikasi sebagai model paling seimbang (F1-Score 0.8636), sementara Naive Bayes menunjukkan performa paling "aman" (Recall 0.9808). Penelitian ini membuktikan bahwa feature engineering yang tepat lebih berdampak daripada pemilihan model, menghasilkan model prediksi yang valid dan robust.

Abstract. Indonesia faces a high risk of tsunami disasters due to its location on the Pacific Ring of Fire, demanding an accurate early warning system. This study proposes a methodologically valid tsunami potential classification model. Unlike previous studies that often used "leaked features," this research implements a deep feature engineering strategy combining physical features (like `is_shallow`) and data quality features (`gap`, `nst`) on a Pacific Ring of Fire dataset. Five machine learning models (Naive Bayes, Logistic Regression, SVM, Random Forest, XGBoost) were evaluated using 10-Fold Cross-Validation. The results show feature engineering was the most crucial step, increasing the F1-Score from ~0.38 to 0.86. Random Forest was identified as the most balanced model (F1-Score 0.8636), while Naive Bayes showed the "safest" performance (Recall 0.9808). This study proves that proper feature engineering is more impactful than model selection, resulting in a valid and robust prediction model.



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

1. PENDAHULUAN

Belum lama ini masyarakat mengalami kepanikan yang disebabkan oleh berita potensi gempa *megathrust* selat

sunda yang memiliki dampak menghasilkan tsunami [1]. Indonesia memang memiliki banyak sekali potensi gempa tingkat tinggi dan menghasilkan tsunami. Potensi ini

dikarenakan Indonesia berada di area cincin api pasifik, dimana area itu juga ada lempeng tektonik seperti lempeng eurasia, indo-australia dan pasifik [2].

Cincin api pasifik merupakan wilayah dengan aktivitas seismik yang tinggi yang berada di sepanjang tepi samudra pasifik, yang membuat daerah ini sering mengalami gempa bumi. Seringnya terjadi gempa bumi membuat daerah ini menjadi penyumbang 90 persen gempa yang terjadi di bumi [3].

Tingkat risiko yang tinggi ini bukanlah sekadar ancaman teoritis. Bencana tsunami yang pernah terjadi di Indonesia dianggap sebagai salah satu yang paling mengerikan dalam sejarah abad ini. Sebuah penelitian yang dilakukan untuk mengetahui penyebab kematian terbanyak di Indonesia menemukan hasil bahwa bencana tsunami di Aceh pada tahun 2004 dianggap sebagai penyebab kematian terbesar kedua di Indonesia dalam beberapa dekade terakhir setelah pandemi COVID-19 [4].

Adanya Sistem Peringatan Dini Tsunami (TEWS) yang cepat, akurat dan efisien sangat diperlukan di wilayah yang rawan terdampak Tsunami untuk mengatasi bencana dan evakuasi karena risiko bencana yang tinggi [5]. Karena hal ini pula setelah Tsunami yang menerjang wilayah Aceh, Indonesia mulai mengembangkan Sistem Peringatan Dini Tsunami (TEWS) [6]. Peringatan dini biasanya bergantung pada ambang batas magnitudo dan kedalaman. Namun, untuk meningkatkan akurasi prediksi, metode kontemporer mulai beralih ke *machine learning*, yang dapat mengenali pola seismik yang lebih kompleks.

Penggunaan *machine learning* untuk mengklasifikasikan tsunami telah menjadi solusi terbaru untuk memitigasi risiko ini. Metode ini telah dieksplorasi oleh berbagai penelitian dengan hasil yang beragam. Sebagai contoh, sebuah penelitian [7] mencoba pendekatan yang lebih kompleks dengan menggunakan Stacking Ensemble yang menggabungkan enam model dasar

dan menemukan *F1-Score* yang sangat baik, yaitu 0.94. Studi lain [8] membandingkan beberapa model, termasuk *KNN* dan *Random Forest*, dimana *KNN* menjadi model terbaik dengan *Precision* 82,8%. Sementara itu, penelitian [9] menerapkan *feature engineering* (seperti jarak ke garis pantai) dan tuning *GridSearch* pada model *XGBoost* dan menghasilkan *F1-Score* 0.75.

Analisis mendalam menunjukkan dua celah (*gap*) metodologis utama, meskipun penelitian [7], [8], [9] memberikan kontribusi yang signifikan.

- Validitas Prediksi (*Data Leakage*): Dalam beberapa penelitian [7], [8], fitur bocor (*leakage feature*), seperti *sig* (skor signifikansi), *mmi* (intensitas guncangan), dan *cdi*, digunakan sebagai prediktor untuk mencapai skor tinggi. Model yang dibuat tidak valid untuk skenario prediksi *real-time* karena data *post-event* ini tidak tersedia pada detik awal gempa.
- Efektifitas Teknik Fitur: Studi yang valid secara metodologis hanya melihat fitur geografis dan analisis satu model (*XGBoost*). Terbukti penelitian ini, masih ada celah untuk mengeksplorasi berbagai pendekatan *feature engineering*.

Tujuan dari penelitian ini adalah untuk mengisi kedua celah tersebut.

- Kebaruan: Dua hal baru dalam penelitian ini. Pertama, model prediksi dirancang secara metodologis untuk menjadi "jujur" dengan hanya menggunakan fitur *pre-event* (tanpa data bocor). Kedua, digunakan strategi teknik fitur yang unik yang menggabungkan fitur fisis (seperti *is_shallow*, *log_energy*) dengan fitur kualitas data seismik (*gap*, *nst*, *rms*).
- Tujuan: Tujuan penelitian ini adalah sebagai berikut: (1) Menciptakan dan menerapkan strategi kualitas data dan *feature engineering* fisis pada dataset gempa Cincin Api Pasifik; (2)

Melakukan studi komparasi kinerja lima "keluarga" model *machine learning* (*Naive Bayes*, *Logistic Regression*, *SVM*, *Random Forest*, dan *XGBoost*); dan (3) Menemukan model klasifikasi potensi tsunami terbaik berdasarkan metrik *F1-Score* dan *Recall* yang kuat dan prediktif.

2. TINJAUAN PUSTAKA

Bab ini menguraikan landasan teoritis dari berbagai metode dan teknik yang digunakan dalam penelitian ini.

2.1. Machine Learning

Machine learning adalah sebuah teknologi, teknik atau cara untuk performa dari sebuah sistem dengan belajar dari metode komputasi [10]. Tujuan dari penggunaan *machine learning* dalam penelitian ini adalah untuk melakukan klasifikasi yaitu termasuk ke dalam jenis dari *machine learning* yaitu *supervised learning* (pembelajaran terarah).

2.2. Model Klasifikasi

2.2.1. Naïve Bayes (NB)

Naive Bayes adalah algoritma klasifikasi probabilistik sederhana yang didasarkan pada Teorema Bayes. Model ini disebut "*naive*" (lugu) karena memiliki asumsi kuat bahwa semua fitur (prediktor) bersifat independen satu sama lain. Meskipun asumsi ini jarang benar di dunia nyata, *NB* seringkali bekerja dengan sangat baik dan cepat [11].

2.2.2. Logistic Regression (LR)

Logistic Regression adalah model linear yang digunakan untuk klasifikasi biner. Model ini menghitung probabilitas sebuah *instance* milik kelas tertentu (misal: Tsunami) dengan melewati kombinasi linear dari fitur-fitur input ke dalam fungsi logistik (*sigmoid*), yang

memetakan nilai *output* antara 0 dan 1 [11].

2.2.3. Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah model geometris yang bekerja dengan menemukan *hyperplane* (garis pemisah) optimal di dalam ruang fitur berdimensi tinggi. *Hyperplane* optimal adalah yang memaksimalkan margin (jarak) antara titik data terdekat dari setiap kelas (disebut *support vectors*). *SVM* juga dapat menangani data *non-linear* secara efektif menggunakan "trik kernel" [12].

2.2.4. Random Forest (RF)

Random Forest adalah salah satu metode *ensemble learning* yang paling populer. Model ini bekerja dengan membangun sejumlah besar *decision trees* (pohon keputusan) selama pelatihan. Setiap pohon dilatih pada sampel acak dari data (*bagging*) dan mempertimbangkan subset acak dari fitur. Untuk klasifikasi, prediksi akhir ditentukan oleh "voting" (suara terbanyak) dari semua pohon individu [13].

2.2.5. Extreme Gradient Boosting (XGBoost)

XGBoost juga merupakan metode *ensemble* berbasis *tree*, tetapi menggunakan teknik yang berbeda yang disebut *boosting*. Tidak seperti *Random Forest* yang membangun pohon secara paralel, *boosting* membangun pohon secara sekuensial. Setiap pohon baru dilatih untuk memperbaiki kesalahan (residual) yang dibuat oleh pohon sebelumnya, menjadikannya model yang sangat kuat dan sering memenangkan kompetisi *data science* [14].

2.3. Metodologi Penanganan Data

2.3.1. Feature Engineering

Feature engineering adalah proses menggunakan pengetahuan domain

(dalam hal ini, fisika gempa) untuk mengekstrak atau membuat fitur-fitur baru dari data mentah. Langkah ini krusial karena performa model *ML* sangat bergantung pada kualitas fitur input. Dalam penelitian ini, fitur mentah seperti *depth* diubah menjadi fitur yang lebih informatif seperti *is_shallow*, yang secara eksplisit "mengajari" model aturan fisis yang relevan [15].

2.3.2. Penanganan Data Tidak Seimbang

Dataset untuk deteksi bencana hampir selalu tidak seimbang (kelas Tsunami jauh lebih sedikit daripada Non-Tsunami). Jika diabaikan, model akan bias terhadap kelas mayoritas. Untuk mengatasi ini, penelitian ini menerapkan parameter *class_weight='balanced'* pada model. Teknik ini secara otomatis memberikan "penalti" atau bobot yang lebih tinggi kepada model jika salah mengklasifikasikan *instance* dari kelas minoritas (Tsunami) [16].

2.4. Evaluasi

2.4.1. K-Fold Cross Validation

Metode ini membagi dataset menjadi 10 "lipatan". Model dilatih 10 kali, setiap kali menggunakan 9 lipatan untuk melatih dan 1 lipatan sisanya untuk menguji. Hasil akhir adalah rata-rata dari 10 pengujian tersebut, yang memberikan estimasi kinerja model yang jauh lebih stabil dan *robust* [11].

2.4.2. Confusion Matrix dan Metrik Evaluasi

Pada data yang tidak seimbang, metrik *Accuracy* (Akurasi) sangat menyesatkan. Sebuah model yang selalu menebak "Non-Tsunami" akan memiliki akurasi tinggi tetapi tidak berguna. Oleh karena itu, penelitian ini menggunakan *Confusion Matrix* (Matriks Kebingungan) untuk membedah performa:

- *True Positive (TP)*: Tsunami, diprediksi Tsunami (Benar)
- *False Positive (FP)*: Non-Tsunami, diprediksi Tsunami (Alarm Palsu)
- *True Negative (TN)*: Non-Tsunami, diprediksi Non-Tsunami (Benar)
- *False Negative (FN)*: Tsunami, diprediksi Non-Tsunami (Gagal Deteksi / Paling Berbahaya)

Dari matriks ini, tiga metrik utama diturunkan [11]:

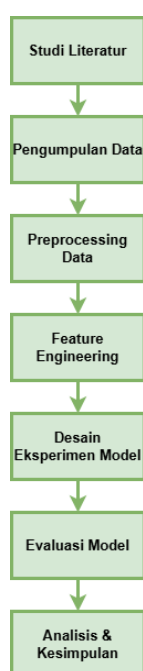
- *Precision*: Seberapa akurat prediksi "Tsunami"? ($TP / (TP + FP)$)
- *Recall* (Sensitivitas): Seberapa baik model menemukan semua Tsunami yang sebenarnya terjadi? ($TP / (TP + FN)$)
- *F1-Score*: Rata-rata harmonik (keseimbangan) antara *Precision* dan *Recall*.

3. METODE PENELITIAN

Bab ini menjelaskan metodologi penelitian secara rinci untuk memastikan penelitian ini dapat direplikasi.

3.1. Rancangan Penelitian

Penelitian ini merupakan penelitian eksperimen komputasional yang bertujuan untuk membandingkan kinerja lima algoritma *machine learning*, yaitu *Naïve Bayes*, *Support Vector Machine (SVM)*, *Random Forest*, *Logistic Regression* dan *XGBoost*, dalam masalah klasifikasi biner untuk memprediksi kejadian tsunami berdasarkan data gempa bumi. Rancangan penelitian dimulai dari studi literatur dan perumusan masalah, pengumpulan data, *preprocessing*, seleksi fitur, pembagian data, pelatihan model, evaluasi, analisis dan kesimpulan. Penelitian ini dilakukan dalam beberapa tahap seperti yang ditunjukkan pada Gambar 1.



Gambar 3.1 Rancangan Penelitian

3.2. Sumber dan Pengumpulan Data

Data gempa bumi diperoleh melalui pengunduhan manual dari *U.S. Geological Survey (USGS) Earthquake Hazards Program* pada 29 Oktober 2025 pukul 16:36 WIB melalui portal katalog gempa *USGS*

(<https://earthquake.usgs.gov/earthquakes/search/>). Format data yang dipilih adalah *JSON/GeoJson* karena menyediakan informasi yang lebih detail dan lengkap dibandingkan format *CSV*. Data difilter dengan kriteria sebagai berikut:

- Wilayah geografis: *Pacific Ring of Fire* (Cincin Api Pasifik). Wilayah ini dipilih alih-alih hanya Indonesia untuk mengatasi masalah kelangkaan data (*data scarcity*) pada kelas minoritas (tsunami), sehingga model memiliki cukup sampel untuk dipelajari.
- Rentang waktu: 1 Januari 1990 hingga 1 Januari 2025.
- Magnitudo: Minimum 6.0 M_w
- Tipe Kejadian: *earthquake*.

3.3. Preprocessing Data

Data mentah *GeoJson* yang kompleks yang ini akan kita ubah menjadi dataset tabel (tabular) yang bersih dan siap untuk langkah selanjutnya atau *feature engineering* di preprocessing data ini. Proses dilakukan menggunakan skrip *python* dengan *library pandas* dan mencakup langkah-langkah berikut:

1. Normalisasi *JSON*: Data mentah memiliki semua informasi relevan di dalam *key featur*s. Dengan menggunakan fungsi *pd.json_normalize* maka struktur *JSON* yang *nested* akan menjadi rata. Setiap atribut di dalam *properties* dan *geometry* juga dipisah menjadi kolom yang terpisah.
2. Ekstraksi Geometri: *longitude*, *latitude* dan *depth* adalah informasi yang menunjukkan lokasi yang tersimpan dalam satu kolom *geometry.coordinates* dalam bentuk *list*. Kolom ini diekstraksi menjadi tiga kolom numerik terpisah.
3. Seleksi Fitur Awal: Tahap ini dilakukan seleksi kolom awal untuk memilih fitur-fitur yang ada saat gempa terjadi dan aman untuk prediksi. Fitur yang merupakan akibat dari gempa seperti *mmi*, *sig*, *felt*, *cdi* dan *alert* sengaja peneliti buang untuk memastikan validitas model prediksi dan mengurangi kemungkinan *data leakage*.
4. Pembersihan Data Mentah: Kolom target tsunami yang memiliki nilai *missing* akan diubah menjadi 0 untuk merepresentasikan kelas “Non-Tsunami” dan kolom-kolom fitur kualitas seperti *gpp*, *dmin*, *nst* dan *rms* diisi dengan *placeholder* sementara untuk tahap *feature engineering*.
5. Penyimpanan Data Bersih: Hasilnya adalah 1435 baris data dengan distribusi kelas target 1070 data

(Non-Tsunami) dan 365 data (Tsunami) disimpan dalam dataset tabular (.csv).

3.4. Teknik Analisis Data

Teknik analisis data meliputi *feature engineering* dan skenario pengujian model.

3.4.1. Feature Engineering

Setelah mendapatkan data yang bersih, tahap *feature engineering* dilakukan untuk mengubah data mentah menjadi representasi yang lebih informatif bagi model *machine learning*.

1. Filter Kejadian: Membuang kejadian lain seperti ledakan tambang dengan hanya menyertakan baris *event_type == 'earthquake'*.
2. Fitur Kategori Kedalaman: *depth* diubah menjadi *is_shallow* (≤ 70 km), *is_intermediate* (70-300km), dan *is_deep* (>300 km).
3. Fitur Turunan: Membuat fitur baru seperti *log_energy* (dihitung dari *mag*) dan *depth_mag_ratio*.
4. Fitur Kualitas Data: *gap*, *dmin* dan lainnya diisini menggunakan nilai media dari masing-masing kolom.
5. Fitur Kategorikal: Kolom *magtype* yang berisi '*mww*' dan '*mwc*' diubah menjadi biner dengan *One-Hot Encoding*.
6. Finalisasi: Membuang kolom yang tidak diperlukan seperti *depth* asli dan *magType* asli.

3.4.2. Skenario Pengujian Model

Skenario pengujian dirancang untuk mengevaluasi dan membandingkan kinerja dari lima model *machine learning* yang telah dipilih. Skenario ini mencakup strategi penanganan data, metode validasi dan metrik evaluasi yang spesifik.

1. Model yang Dibandingkan: Penelitian ini melakukan studi komparatif terhadap lima algoritma *machine*

learning yang mewakili keluarga *machine learning* yang berbeda:

- a. Model Probabilistik: *Gaussian Naïve Bayes (NB)*.
 - b. Model Linear: *Logistic Regression (LR)*.
 - c. Model Geometris (Kernel): *Support Vector Machine (SVM)*.
 - d. Model Ensemble (Tree-based): *Random Forest (RF)* dan *XGBoost (XGB)*.
2. Penangan Data Tidak Seimbang: Dataset penelitian memiliki rasio kelas 3 banding 1 yang tidak seimbang, hal ini bisa mengakibatkan model menjadi bias terhadap kelas yang lebih mayoritas. Untuk mengatasi hal ini, sebuah strategi penanganan pada level algoritma peneliti terapkan:
 - a. Mengaktifkan parameter *class_weight='balanced'* pada model *Logistic Regression, SVM, Random Forest* dan *XGBoost*.
 - b. Parameter ini memberikan bobot lebih berat jika model salah mengklasifikasikan kelas minoritas.
 3. Metode Validasi Model: Untuk mendapatkan estimasi kinerja yang *robust* dan menghindari bias dari satu kali pembagian data latih-uji, penelitian ini menerapkan skema *10-fold cross-validation (CV)*.
 4. Metrik Evaluasi Kinerja: Klasifikasi data tidak seimbang perlu dilakukan evaluasi untuk model dengan menggunakan metrik yang relevan sebagai berikut:
 - a. *Precision*: Mengukur seberapa akurat prediksi 'tsunami' dari model. (Dari semua yang diprediksi tsunami, berapa persen yang benar-benar tsunami?).
 - b. *Recall* (Sensitivitas): Mengukur kemampuan model untuk menemukan semua kejadian

Tsunami yang sebenarnya. Metrik ini sangat krusial dalam hal mitigasi bencana dimana *false negative* (melewatkan tsunami) akan sangat berbahaya.

- c. *F1-Score*: Rata-rata harmonik dari *Precision* dan *Recall*. Metrik ini peneliti gunakan sebagai skor keseimbangan utama untuk menentukan model mana yang memiliki performa paling optimal secara keseluruhan.

4. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dari metodologi yang telah diuraikan pada Bab 3. Pembahasan akan berfokus pada analisis dataset terakhir, efek *feature engineering*, perbandingan kinerja lima model klasifikasi, dan interpretasi hasil.

4.1. Analisis Dataset

Pelatihan dan evaluasi model menggunakan dataset yang berasal dari hasil akhir proses *preprocessing* dan juga *feature engineering*. Setelah melalui penyaringan *event_type='earthquake'*, 1435 baris data yang valid didapatkan dan menjadi dataset final.

Hasil kinerja model akan lebih bisa dipahami saat karakteristik dari data yang digunakan sudah lebih dahulu dimengerti. Untuk itu Tabel(berapa) menyajikan fitur-fitur yang digunakan untuk melatih model.

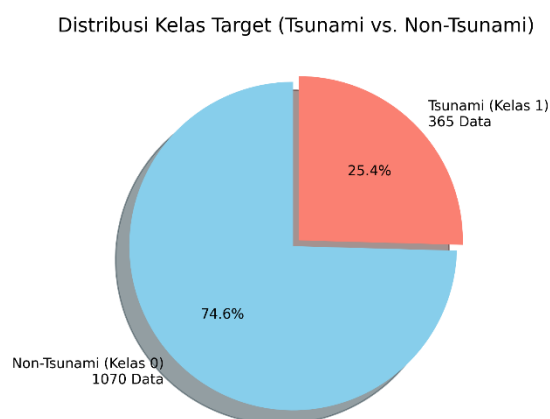
Tabel 4.1 Deskripsi Fitur

Nama Fitur	Deskripsi	Tipe Data
<i>mag</i>	Magnitudo gempa (SR)	Numerik
<i>latitude</i>	Garis lintang	Numerik
<i>longitude</i>	Garis bujur	Numerik

<i>log_energy</i>	Estimasi Logaritma	Numerik
<i>depth_mag_ratio</i>	Rasio Kedalaman dan Magnitudo	Numerik
<i>is_shallow</i>	depth <= 70km	Biner
<i>is_intermediate</i>	70km < depth < 300km	Biner
<i>is_deep</i>	depth > 300km	Biner
<i>gap</i>	Azimuthal gap	Numerik
<i>dmin</i>	Jarak ke stasiun	Numerik
<i>nst</i>	Jumlah stasiun	Numerik
<i>rms</i>	RMSE dari waktu tiba	Numerik
<i>magType_mwb</i>	Mwb	Biner
<i>magType_mwc</i>	Mwc	Biner
<i>magType_mww</i>	Mww	Biner
<i>magType_other</i>	Tipe umum	Biner

Tabel 4.1 menunjukkan bahwa model tidak dilatih menggunakan data mentah, melainkan menggunakan kumpulan fitur yang telah melalui proses *feature engineering* secara ekstensif. Terdapat total 16 fitur yang mencakup informasi fisis (kategori kedalaman, energi), informasi lokasi, dan informasi kualitas dari data (*gap, nst, rms*).

Selain fitur prediktor, karakteristik dataset yang paling penting adalah distribusi kelas dari target yaitu tsunami, seperti yang digambarkan pada Gambar 4.1.



Gambar 4.1 Distribusi Kelas Target

Gambar 4.1 menunjukkan bahwa dataset final bersifat tidak seimbang (*imbalanced*). Dari total 1435 sampel, 1070 (74.6%) diklasifikasikan sebagai kelas 0 (Non-Tsunami), sementara 365 (25.4%) diklasifikasikan sebagai kelas 1 (Tsunami).

Rasio kelas ~3:1 ini dianggap sebagai *mild imbalanced*. Meskipun tidak ekstrem, ketidakseimbangan ini sangat penting untuk dipertimbangkan. Jika diabaikan, model *machine learning* akan cenderung bias dan memiliki performa yang buruk dalam memprediksi kelas minoritas (Tsunami), yang justru merupakan kelas paling penting untuk dideteksi dalam skenario mitigasi bencana.

Oleh karena itu, seperti yang dijelaskan sebelumnya pada Bab 3.4.2, strategi penanganan data tidak seimbang (yaitu *class_weight='balanced'*) diterapkan selama pelatihan model. Hal ini memastikan bahwa model memberikan 'penalti' yang lebih besar jika salah mengklasifikasikan kejadian Tsunami, sehingga memaksa model untuk lebih memperhatikan pola dari kelas minoritas.

4.2. Dampak Feature Engineering

Setelah menganalisis karakteristik dataset, analisis dialihkan untuk mengukur dampak dari metodologi yang diusulkan. Temuan paling signifikan dari penelitian ini bukanlah pada perbandingan akhir antar

model, melainkan pada efek transformasi dari feature engineering.

Untuk mengukur dampak ini secara kuantitatif, dilakukan dua skenario eksperimen yang berbeda:

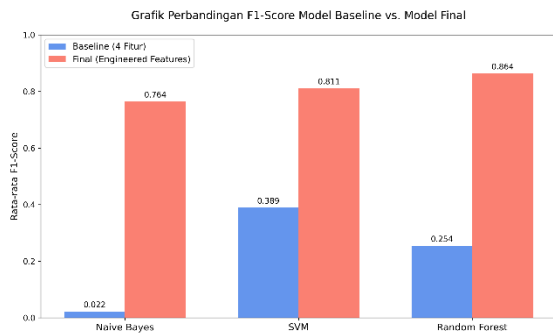
1. Model Baseline: Tiga model awalnya (*NB*, *SVM*, *RF*) dilatih sebagai percobaan awal hanya menggunakan empat fitur prediktif dasar, yaitu *mag*, *depth*, *latitude*, *longitude*. Fitur-fitur ini dipilih karena mewakili hiposenter fundamental secara instan.
2. Model Final: Model dilatih menggunakan dataset final yang telah melalui proses *feature engineering* ekstensif (seperti yang dijelaskan di Tabel 4.1), yang mencakup fitur kategorikal kedalaman, energi seismik, dan fitur kualitas data.

Perbandingan *F1-Score* rata-rata (dari *10-Fold CV* antara kedua skenario eksperimen tersebut disajikan pada tabel 4.2 dibawah.

Tabel 4.2 Perbandingan F1-Score Model

Model	<i>F1-Score Baseline</i>	<i>F1-Score Final</i>	Peningkatan
<i>Naïve Bayes</i>	0.0217	0.7639	+ 0.7422
<i>SVM</i>	0.3890	0.8106	+ 0.4216
<i>Random Forest</i>	0.2542	0.8636	+ 0.6094

Meskipun tabel 4.2 menunjukkan peningkatan numerik, dampak visualnya jauh lebih jelas terlihat pada gambar 4.2, yang membandingkan performa model secara langsung.



Gambar 4.2 Perbandingan F1-Score

Gambar 4.2 secara visual lebih bisa dilihat peningkatannya. Performa model baseline (batang biru) sangat buruk, dengan F1-Score tertinggi hanya 0.3890 (SVM) dan model lain (RF dan NB) gagal total. Ini membuktikan hipotesis bahwa model machine learning, bahkan yang canggih sekalipun, tidak dapat secara ajaib mempelajari fisika dari gempa yang kompleks hanya berdasarkan data mentah.

Bagi model *baseline* ini, angka *depth* 40km dan 80km hanyalah dua angka yang berdekatan. Model tidak “paham” bahwa 40km berada di zona subduksi yang berbahaya sementara 80km (meskipun secara numerik dekat) mungkin sudah berada di zona yang lebih aman (di bawah lempeng). Model juga tidak paham bahwa mag 6.5 dan 7.5 itu bukanlah perbedaan linear, melainkan perbedaan energi yang eksponensial.

Di sinilah letak keberhasilan *feature engineering* (batang oranye). Peningkatan F1-Score yang drastis (misalnya, +0.6094 untuk *Random Forest*) terjadi karena model tidak lagi “menebak” fisika. Model “diajari” fisika secara eksplisit melalui fitur-fitur baru:

- *is_shallow*: Fitur biner ini secara tegas memberi tahu model, “Perhatikan gempa ini, kedalamannya $\leq 70\text{km}$,” yang merupakan aturan *domain knowledge* paling fundamental untuk tsunami.

- *log_energy*: Fitur ini mengubah skala magnitudo dari linear menjadi logaritmik, yang lebih akurat merepresentasikan energi seismik yang dilepaskan.
- *gap*, *nst* dan *rms*: Fitur-fitur ini memberi model konteks tentang seberapa dapat dipercaya perhitungan lokasi dan magnitudo gempa tersebut. Model mungkin belajar bahwa gempa dengan *gap* tinggi (kualitas buruk) dan mag 7.0 kurang bisa dipercaya daripada gempa dengan *gap* rendah dan mag 7.0.

Dapat disimpulkan dari analisis ini bahwa langkah *feature engineering* (Bab 3.4.1) adalah langkah paling krusial dalam penelitian ini. Tanpa langkah ini, perbandingan model di bagian selanjutnya tidak akan relevan, karena semua model akan gagal karena input data yang buruk. Ini menegaskan bahwa dalam aplikasi *data science* untuk ilmu kebumih, pemahaman domain dan penciptaan fitur yang cerdas jauh lebih berdampak daripada sekadar memilih algoritma yang paling canggih.

4.3. Hasil 5 Model Klasifikasi

Setelah membuktikan bahwa *feature engineering* adalah langkah krusial, bagian ini menyajikan hasil evaluasi *10-Fold Cross-Validation* secara lengkap terhadap kelima model klasifikasi.

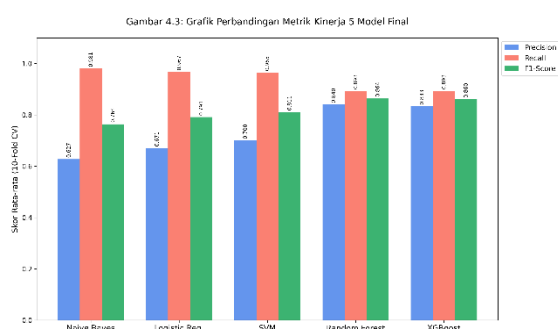
Hasil komprehensif dari perbandingan kinerja dirangkum dalam Tabel 4.3.

Tabel 4.3 Perbandingan Kinerja Model

Model	Avg. Precision	Avg. Recall	Avg. F1-Score
Naive Bayes	0.6274	0.980	0.763
		8	9

<i>Logistic Regression</i>	0.6707	0.9671	0.7912
<i>SVM</i>	0.7002	0.9645	0.8106
<i>Random Forest</i>	0.8399	0.8934	0.8636
<i>XGBoost</i>	0.8331	0.8934	0.8602

Untuk memvisualisasikan perbandingan dan *trade-off* antar metrik dengan lebih jelas, Gambar 4.3 menyajikan perbandingan skor untuk setiap model.



Gambar 4.3 Grafik Perbandingan Kinerja Model

Tabel 4.3 dan Gambar 4.3 secara jelas menunjukkan bahwa tidak ada satu model pun yang "sempurna", dan terdapat *trade-off* yang jelas antara metrik *Precision* dan *Recall*.

1. Analisis Kinerja *F1-Score* (Model Paling Seimbang): *F1-Score*, yang merupakan rata-rata harmonik dari *Precision* dan *Recall*, sering dianggap sebagai metrik paling seimbang untuk data *imbalanced*. Berdasarkan metrik ini, *Random Forest* (0.8636) dan *XGBoost* (0.8602) secara signifikan mengungguli tiga model lainnya. Kinerja superior ini menunjukkan bahwa model *ensemble* berbasis *tree* (pohon keputusan) paling mampu menangkap pola *non-linear* yang kompleks dari fitur-fitur yang

telah di-engineer (seperti *is_shallow*, *log_energy*, dan *gap*). *SVM* menyusul sebagai model *non-ensemble* terbaik dengan *F1-Score* solid 0.8106.

2. Analisis Kinerja *Recall* (Model Paling "Aman" / Sensitif): Dalam konteks mitigasi bencana tsunami, metrik *Recall* (sensitivitas) bisa dibilang adalah yang paling krusial. *Recall* 98% berarti model berhasil mengidentifikasi 98% dari semua tsunami yang sebenarnya terjadi. Dalam hal ini, temuan yang sangat menarik adalah performa *Naive Bayes*, yang mencapai *Recall* tertinggi (0.9808). *Model Logistic Regression* (0.9671) dan *SVM* (0.9645) juga menunjukkan sensitivitas yang sangat tinggi. Ini menunjukkan bahwa model probabilistik dan linear/geometris, ketika diberi fitur yang tepat, sangat efektif dalam memastikan tidak ada kejadian tsunami yang terlewat (*false negative* yang rendah).

3. Analisis *Trade-off* (*Precision* vs. *Recall*):

- a. *Random Forest* / *XGBoost* (*F1* Terbaik): Model-model ini memiliki *Precision* tertinggi (di atas 0.83). Artinya, ketika mereka memprediksi "Tsunami", mereka sangat andal dan menghasilkan sedikit *false positive* (alarm palsu). Namun, "harga" dari presisi ini adalah *Recall* yang lebih rendah (0.8934), yang berarti mereka melewatkan sekitar 10.7% dari semua tsunami (*False Negative*).
- b. *Naive Bayes* (*Recall* Terbaik): Model ini memiliki *Precision* terendah (0.6274). Artinya, ketika ia memprediksi "Tsunami", ia hanya 62.7% benar. Ini berarti ia menghasilkan banyak *false*

positive (alarm palsu). Namun, "harga" dari alarm palsu ini adalah *Recall* yang superior (0.9808), yang berarti ia hampir tidak pernah melewatkan tsunami (*False Negative* sangat rendah, hanya ~1.9%)

Secara keseluruhan, untuk tujuan akademik yang mencari model paling seimbang dan optimal, *Random Forest* adalah pemenang yang jelas. Namun, untuk aplikasi dunia nyata, *Naive Bayes* menunjukkan potensi sebagai sistem sekunder "paling aman" yang dapat diimplementasikan untuk meminimalkan risiko *false negative*.

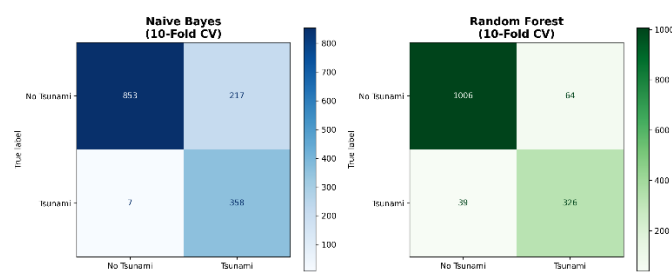
4.4. Analisis Confusion Matrix

Hasil pada tabel 4.3 menyajikan skor metrik rata-rata yang sangat baik untuk perbandingan umum. Namun, untuk memahami implikasi praktis dan jumlah absolut dari tipe kesalahan yang dibuat oleh model, perlu dilakukan analisis terhadap *Agregat Confusion Matrix* dari *10-Fold Cross-Validation*.

Metode ini dieksekusi menggunakan fungsi *cross_val_predict* lalu prediksi *10 fold* pengujian dikumpulkan sehingga menghasilkan *confusion matrix* yang didasarkan pada prediksi untuk keseluruhan 1435 data.

Analisis ini membedah dua model paling representatif dari temua sebelumnya:

1. *Random Forest (RF)*: Sebagai model dengan *F1-Score* tertinggi.
2. *Naive Bayes*: Sebagai model dengan *Recall* tertinggi.



Gambar 4.4 *Confusion Matrix NB & RF*

Gambar 4.4 menyajikan gambaran yang jelas tentang *trade-off* kinerja dari kedua model.

1. Analisis *Random Forest* (Gambar 4.4, Kanan): Model *Random Forest* menunjukkan kinerja yang sangat seimbang. Kekuatan utamanya terletak pada kemampuannya meminimalkan *False Positive* (FP) (hanya 64 kasus). Ini berarti, dari 1070 kejadian non-tsunami, model ini hanya salah memprediksi 64 di antaranya sebagai tsunami. Ini sejalan dengan skor *Precision*-nya yang tinggi (0.84). Model ini adalah yang paling pandai menganalisis. Namun, keandalan ini dibayar dengan 39 kasus *False Negative* (FN), di mana model gagal mendeteksi 39 dari 365 tsunami yang sebenarnya terjadi.
2. Analisis *Naive Bayes* (Gambar 4.4, Kiri): Model *Naive Bayes* menceritakan kisah yang sangat berbeda. Model ini secara drastis meminimalkan *False Negative* (FN) (hanya 7 kasus). Ini berarti model *NB*, dari total 365 kejadian tsunami, hanya melewatkan 7 di antaranya (*Recall* 98.1%). Model ini adalah "penjaga yang super-sensitif". Namun, sensitivitas ekstrem ini dibayar mahal dengan jumlah *False Positive* (FP) yang sangat tinggi (sebanyak 217 kasus). Model ini sering "berteriak" ada tsunami padahal tidak ada.

Implikasi Teoritis dan Implementasi: Kedua matriks agregat ini menghadirkan dua filosofi strategis yang berbeda untuk implementasi di dunia nyata:

- Jika Tujuannya Keseimbangan dan Efisiensi: Model *Random Forest* adalah pemenangnya. Dalam skenario di mana "alarm palsu" (FP) harus ditekan seminimal mungkin (untuk menghindari *warning fatigue* atau biaya evakuasi), *RF* adalah pilihan yang lebih efisien dan seimbang.
- Jika Tujuannya Keamanan Mutlak (*Zero-Miss Policy*): Model *Naive Bayes* adalah pemenangnya. Dalam skenario mitigasi bencana di mana 39 kejadian yang terlewat (seperti pada *RF*) dianggap terlalu banyak, maka 7 kejadian terlewat dari *NB* jauh lebih dapat diterima. Model ini beroperasi di bawah filosofi "lebih baik 217 alarm palsu daripada 7 tsunami terlewat".

Temuan ini mengonfirmasi bahwa tidak ada "satu model terbaik" secara absolut, melainkan "model terbaik untuk tujuan tertentu". Untuk tujuan penelitian ini yang mencari keseimbangan optimal, *RF* tetap menjadi pemenang. Namun, untuk implementasi praktis yang memprioritaskan keamanan, *Naive Bayes* menunjukkan kinerja yang sangat berharga.

4.5. Penelitian Sebelumnya

Temuan utama dari penelitian ini adalah bahwa strategi *feature engineering* yang tepat (menggabungkan fitur fisis dan kualitas data) dapat meningkatkan kinerja model, dengan *Random Forest* mencapai *F1-Score* rata-rata 0.8636 (Tabel 4.3). Temuan ini memberikan kontribusi signifikan jika dibandingkan dengan *state-of-the-art* yang telah diidentifikasi di Bab 1.

Perbandingan ini harus dibagi menjadi dua konteks: (1) perbandingan terhadap penelitian yang menggunakan data

"bocor", dan (2) perbandingan terhadap penelitian yang valid secara metodologis.

1. Validitas Metodologi (Menghindari Data "Bocor"): Beberapa penelitian sebelumnya [7], [8] melaporkan skor yang sangat tinggi, seperti *F1-Score* 0.94 atau *AUC* 94.4%. Namun, seperti yang diidentifikasi dalam *gap analysis* (Bab 1.4), skor-skor ini dicapai dengan menggunakan fitur *post-event* atau "bocor" (*leaked features*) seperti *sig*, *mmi* dan *cdi* sebagai prediktor. Meskipun model-model tersebut akurat secara historis, model tersebut tidak valid untuk skenario prediksi *real-time*, yang merupakan tujuan utama dari sistem peringatan dini. Oleh karena itu, *F1-Score* 0.8636 yang dicapai dalam penelitian ini—yang secara ketat hanya menggunakan fitur *pre-event* (data yang tersedia pada detik gempa terjadi)—merupakan hasil yang lebih "jujur", valid, dan relevan secara praktis untuk mitigasi bencana.

2. Efektivitas *Feature Engineering*: Perbandingan yang lebih relevan dapat dilakukan dengan penelitian [9], yang juga berfokus pada model prediksi yang valid secara metodologis (tanpa data bocor) dan menggunakan model *XGBoost*. Penelitian [9] menerapkan *feature engineering* geografis (jarak ke pantai) dan dan oversampling (*SMOTE*), menghasilkan *F1-Score* akhir 0.75. Model *XGBoost* dalam penelitian ini, yang menggunakan strategi *feature engineering* yang berbeda (menggabungkan fitur fisis *is_shallow* dengan fitur kualitas *gap*, *nst* dan lainnya) dan penanganan *imbalance* (*class_weight*), berhasil mencapai *F1-Score* rata-rata yang secara

signifikan lebih tinggi yaitu 0.8602 (Tabel 4.3).

Hasil ini secara langsung mengisi "celah" penelitian yang ditemukan di Bab 1 dengan menyediakan model klasifikasi tsunami yang valid secara metodologis dan berkinerja lebih unggul melalui strategi pengembangan fitur yang diusulkan.

5. KESIMPULAN

Berdasarkan analisis dan pembahasan yang telah dilakukan, penelitian ini menghasilkan beberapa kesimpulan utama sebagai berikut:

1. *Feature engineering* terbukti sebagai langkah paling krusial dalam penelitian ini. Penerapan fitur fisis dan fitur kualitas data seismik berhasil meningkatkan kinerja model, dari *F1-Score* 0.25 (model *baseline*) sampai menjadi 0.8636 (model final *Random Forest*).
2. Dari lima model yang diuji, *Random Forest* teridentifikasi sebagai model dengan kinerja paling seimbang, mencapai *F1-Score* rata-rata tertinggi (0.8636) dan *Precision* tinggi (0.8399). Namun, model *Naive Bayes* menunjukkan keunggulan sebagai model "paling aman" (paling sensitif) dengan Recall 0.9808, yang berarti model tersebut hampir tidak pernah melewatkan kejadian tsunami yang sebenarnya (hanya 7 *False Negative* dari 365 kasus).
3. Kelebihan dari penelitian ini adalah mengusulkan metodologi prediksi yang lebih jujur tanpa *data leakage* yang penting untuk skenario real time dan *feature engineering* yang diusulkan juga lebih unggul dibanding penelitian sejenis. Kekurangannya adalah kinerja model ini belum diuji oleh data sensor *real-time* yang mungkin memiliki tingkat *noise* yang berbeda.

4. Untuk penelitian di masa depan, disarankan untuk: (1) Melakukan *hyperparameter* tuning secara ekstensif (misalnya menggunakan *GridSearchCV*) pada model *Random Forest* dan *XGBoost* untuk lebih mengoptimalkan *F1-Score*, dan (2) Menerapkan arsitektur *Deep Learning* (seperti *Artificial Neural Networks*) pada kumpulan fitur yang telah terbukti efektif dalam penelitian ini.

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala limpahan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan artikel ilmiah yang berjudul "Analisis Peran Feature Engineering Pada Kinerja Model Machine Learning Untuk Klasifikasi Potensi Tsunami" ini dengan baik.

DAFTAR PUSTAKA

- [1] E. H. S. Aulia, "SVM, NAIVE BAYES, DECISION TREE PADA ANALISIS SENTIMEN MASYARAKAT TERHADAP ISU GEMPA MEGATHRUST," Universitas Nusantara PGRI Kediri, Kediri, 2025. Accessed: Nov. 04, 2025. [Online]. Available: https://repository.unpkediri.ac.id/18222/3/RAMA_57201_2113030024_0723058501_0706_098902_01_front_ref.pdf
- [2] J. Jumadi *et al.*, "Tsunami Risk Mapping and Sustainable Mitigation Strategies for Megathrust Earthquake Scenario in Pacitan Coastal Areas, Indonesia," *Sustainability (Switzerland)*, vol. 17, no. 6, Mar. 2025, doi: 10.3390/su17062564.
- [3] National Geographic Society, "Plate Tectonics and the Ring of Fire," National Geographic. Accessed: Nov. 03, 2025. [Online]. Available: <https://education.nationalgeographic.org/resource/plate-tectonics-ring-fire/>
- [4] S. S. Sabrina, "VISUALISASI DATA PENYEBAB KEMATIAN DI INDONESIA RENTANG TAHUN 2000-2022 DENGAN POWER BI," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4071.
- [5] Benazir and R. S. Oktari, "Assessing tsunami risk along the Aceh coast, Indonesia: a quantitative analysis of fault rupture potential

- and early warning system efficacy for predicting arrival time and flood extent,” *Natural Hazards*, vol. 120, no. 5, pp. 4875–4900, Mar. 2024, doi: 10.1007/s11069-024-06401-x.
- [6] W. Dharmawan, M. Diana, B. Tuntari, I. M. Astawa, S. Rahardjo, and H. Nambo, “Tsunami tide prediction in shallow water using recurrent neural networks: model implementation in the Indonesia Tsunami Early Warning System,” *J Reliab Intell Environ*, vol. 10, no. 2, pp. 177–195, Jun. 2024, doi: 10.1007/s40860-023-00214-8.
- [7] Sudarto and Kusriani, “KLASIFIKASI TSUNAMI GEMPA BUMI DENGAN TEKNIK STACKING ENSEMBLE MACHINE LEARNING,” *Jurnal Informatika Polinema*, vol. 10, no. 4, Aug. 2024, Accessed: Nov. 05, 2025. [Online]. Available: <https://jurnal.polinema.ac.id/index.php/jip/article/view/5655>
- [8] E. Mardiani *et al.*, “Classification of Potential Tsunami Disaster Due to Earthquakes in Indonesia Based on Machine Learning,” *International Journal Software Engineering and Computer Science (IJSECS)*, vol. 4, no. 1, pp. 13–23, Apr. 2024, doi: 10.35870/ijsecs.v4i1.2084.
- [9] I. G. J. Kurniawan, C. Dewi, and M. A. Rahman, “PENERAPAN MACHINE LEARNING EXTREME GRADIENT BOOSTING DALAM KLASIFIKASI POTENSI TSUNAMI BERDASARKAN DATA GEMPA BUMI,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 2, Feb. 2025, Accessed: Nov. 05, 2025. [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/14437>
- [10] Z.-H. Zhou, *Machine Learning*, 4th ed. Singapore: Springer Singapore, 2021. doi: 10.1007/978-981-15-1967-3.
- [11] Jiawei. Han, *Data Mining Concepts and Techniques*, 4th ed. Morgan Kauffmann, 2022. Accessed: Nov. 05, 2025. [Online]. Available: <https://datamineaz.org/textbooks/hanDataMiningConceptual.pdf>
- [12] C. Cortes and V. Vapnik, “Support-Vector Networks Editor,” *Machine Learning*, vol. 20, pp. 273–297, 1995, Accessed: Nov. 05, 2025. [Online]. Available: <https://link.springer.com/article/10.1007/Bf00994018>
- [13] L. Breiman, “Random Forests,” *Mach Learn*, vol. 45, pp. 5–32, 2001, Accessed: Nov. 05, 2025. [Online]. Available: <https://link.springer.com/article/10.1023/a:1010933404324>
- [14] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [15] T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke, “Special issue on feature engineering editorial,” *Mach Learn*, vol. 113, no. 7, pp. 3917–3928, Jul. 2024, doi: 10.1007/s10994-021-06042-2.
- [16] B. BAKIRARAR and A. H. ELHAN, “Class Weighting Technique to Deal with Imbalanced Class Problem in Machine Learning: Methodological Research,” *Turkiye Klinikleri Journal of Biostatistics*, vol. 15, no. 1, pp. 19–29, Jan. 2023, doi: 10.5336/biostatic.2022-93961.