BAB 2
LANDASAN TEORI

Bab ini menguraikan tinjauan pustaka dan landasan teoretis yang menjadi pilar utama dalam pelaksanaan penelitian ini. Konsep-konsep yang dibahas mencakup pemahaman mendalam mengenai penelitian terdahulu yang relevan, dinamika curah hujan dan dampaknya terhadap bencana banjir, prinsip dasar klasifikasi data meteorologi, karakteristik data deret waktu (*time series*) beserta rekayasa fitur temporal, hingga paradigma pembelajaran mesin (*machine learning*). Lebih jauh, bab ini juga membedah secara spesifik arsitektur matematis dan mekanisme kerja dari algoritma *Random Forest* dan *Extreme Gradient Boosting* (XGBoost), metrik evaluasi performa model termasuk *Receiver Operating Characteristic*–Area Under the Curve (ROC-AUC), serta signifikansi interpretabilitas model melalui pendekatan *SHapley Additive exPlanations* (SHAP) dalam pengambilan keputusan mitigasi bencana.

2.1 Penelitian Terdahulu

Kajian terhadap penelitian terdahulu dilakukan untuk memetakan perkembangan literatur yang relevan, mengidentifikasi kesenjangan yang belum terjawab, serta memposisikan kontribusi orisinal dari penelitian ini. Tabel 2.1 merangkum delapan penelitian representatif yang menjadi landasan komparatif.

Tabel 2.1. Ringkasan Penelitian Terdahulu

No	Peneliti (Tahun)	Topik	Metode	Hasil Utama	Perbedaan
1	Pringandana & Kusnawi (2025) [1]	Klasifikasi curah hujan *multiclass* di Jakarta	XGBoost, LightGBM	XGBoost unggul dengan akurasi tertinggi untuk prediksi banjir Jakarta	Penelitian ini menambahkan analisis RF dan interpretabilitas SHAP

Lanjut ke halaman berikutnya...

Tabel 2.1 Penelitian Terdahulu (Lanjutan)

No	Peneliti (Tahun)	Topik	Metode	Hasil Utama	Perbedaan
2	Achmad dkk. (2025) [2]	Pemetaan keparahan banjir DKI Jakarta	Random Forest	RF efektif memetakan zona rawan banjir berbasis data historis	Penelitian ini melakukan perbandingan RF dengan XGBoost secara kuantitatif
3	Imanullah & Fanani (2026) [4]	Prediksi hujan harian di Gunung Ungaran	RF, XGBoost, SMOTE	XGBoost + SMOTE mencapai F1-Score tertinggi pada data tidak seimbang	Penelitian ini menggunakan data BMKG multi-stasiun DKI Jakarta
4	Appiah-Badu dkk. (2022) [10]	Klasifikasi hujan biner di Ghana (IEEE Access)	RF, XGBoost, DT, MLP, KNN	RF dan XGBoost secara konsisten mengungguli model lain dengan ROC-AUC > 0.85	Penelitian ini menerapkan *time series split* dan analisis SHAP
5	Wu dkk. (2024) [11]	Prediksi sensitivitas banjir berbasis ML	RF, XGBoost, Logistic Regression	XGBoost menghasilkan akurasi tertinggi; *feature importance* mengidentifikasi variabel curah hujan dominan	Penelitian ini berfokus pada klasifikasi kejadian hujan harian, bukan pemetaan sensitivitas

Lanjut ke halaman berikutnya...

Tabel 2.1 Penelitian Terdahulu (Lanjutan)

No	Peneliti (Tahun)	Topik	Metode	Hasil Utama	Perbedaan
6	Mdegela dkk. (2023) [12]	Klasifikasi hujan ekstrem untuk EWS banjir sungai	RF, XGBoost, SVM, DT	*Class imbalance* adalah tantangan utama; RF dan XGBoost memberikan Recall terbaik	Penelitian ini menggunakan data BMKG tiga stasiun dan metodologi CRISP-DM
7	Imani dkk. (2025) [13]	Analisis komprehensif RF & XGBoost dengan SMOTE, ADASYN, GNUS	RF, XGBoost	RF + SMOTE dan XGBoost + SMOTE secara konsisten terbaik pada data sangat tidak seimbang	Penelitian ini menggunakan *classweight* dan *scaleposweight* sebagai pengganti oversampling eksplisit
8	Ren dkk. (2024) [14]	Penilaian kerentanan banjir dengan RF dan XGBoost	RF, XGBoost	XGBoost mencapai ROC-AUC tertinggi; *random sampling strategy* meningkatkan stabilitas model	Penelitian ini menambahkan analisis interpretabilitas SHAP untuk transparansi prediksi

Berdasarkan pemetaan tersebut, penelitian ini menutup tiga celah yang belum diatasi secara bersamaan dalam satu kerangka: (1) komparasi RF vs XGBoost yang dilakukan pada data BMKG multi-stasiun DKI Jakarta dengan (2) validasi menggunakan *time series splitting* yang ketat, serta (3) analisis interpretabilitas ganda menggunakan MDI *Feature Importance* dan SHAP.

2.2 Curah Hujan dan Banjir

Curah hujan merupakan salah satu parameter iklim yang sangat krusial dan merujuk pada jumlah volume air yang jatuh ke permukaan bumi dalam periode waktu tertentu, yang secara mendasar dipengaruhi oleh berbagai faktor atmosferik seperti suhu, tekanan udara, dan kelembapan [8]. Dalam siklus hidrologi, presipitasi atau hujan terbagi ke dalam berbagai jenis berdasarkan mekanisme pengangkatannya, seperti hujan konvektif yang sering terjadi di wilayah tropis akibat pemanasan lokal yang intens, hujan orografis yang dipicu oleh topografi pegunungan, serta hujan frontal. Di kawasan perkotaan dengan topografi dataran rendah seperti Daerah Khusus Ibukota (DKI) Jakarta, intensitas curah hujan memiliki korelasi langsung terhadap peningkatan risiko bencana hidrometeorologi, khususnya banjir [4]. Kombinasi antara intensitas hujan yang ekstrem, minimnya daerah resapan air akibat masifnya pembangunan infrastruktur beton, serta sistem drainase yang sering kali kelebihan kapasitas, menyebabkan limpasan permukaan (*surface runoff*) tidak dapat dikendalikan. Banjir di DKI Jakarta tidak hanya merendam kawasan pemukiman, tetapi juga melumpuhkan aktivitas sosial-ekonomi, merusak infrastruktur vital, dan mengancam keselamatan jiwa penduduk yang terdampak [2]. Oleh karena itu, kemampuan untuk memprediksi dan memitigasi dampak dari curah hujan ekstrem menjadi sebuah urgensi yang absolut, di mana pemahaman pola historis kejadian hujan dapat digunakan untuk mengoptimalkan rekayasa hidrologis seperti pengaktifan pompa air secara tepat waktu maupun evakuasi warga sebelum genangan air mencapai tingkat keparahan yang mematikan.

2.3 Klasifikasi Kejadian Hujan

Klasifikasi kejadian hujan merupakan suatu proses pemetaan data meteorologis ke dalam kategori-kategori diskrit guna mengidentifikasi pola cuaca secara matematis dan sistematis. Dalam konteks pemodelan prediksi cuaca, klasifikasi dapat bersifat biner, yakni menentukan apakah akan terjadi hujan atau tidak hujan pada rentang waktu tertentu, maupun bersifat multikelas yang mengategorikan hujan berdasarkan tingkat intensitasnya, seperti ringan, sedang, lebat, dan sangat lebat [3]. Pentingnya klasifikasi ini terletak pada kemampuannya untuk mengonversi data cuaca mentah yang kompleks dan fluktuatif menjadi informasi prediktif yang mudah dipahami dan ditindaklanjuti oleh otoritas terkait.

Dalam ranah pengambilan keputusan strategis mitigasi bencana, hasil klasifikasi ini bertindak sebagai fondasi utama bagi Sistem Peringatan Dini (EWS). Dengan mengetahui probabilitas dan kelas kejadian hujan pada hari berikutnya, instansi terkait seperti Badan Penanggulangan Bencana Daerah (BPBD) dapat merancang intervensi yang proporsional. Kesalahan dalam klasifikasi, khususnya kegagalan dalam mendeteksi kelas hujan lebat (*false negative*), dapat berakibat fatal karena menyebabkan ketiadaan respons mitigasi saat bencana benar-benar terjadi [1]. Oleh karena itu, sistem klasifikasi yang tangguh dan memiliki akurasi prediktif tinggi menjadi prasyarat mutlak untuk menjamin efektivitas upaya pengurangan risiko bencana.

2.4 Data *Time Series* dan *Lag Features* pada Data Cuaca

Data deret waktu atau *time series* didefinisikan sebagai rangkaian observasi variabel kuantitatif yang dicatat secara berurutan berdasarkan interval waktu yang tetap dan kronologis [7]. Data cuaca dan iklim pada dasarnya merupakan entitas yang terikat kuat dengan dimensi waktu, di mana kondisi meteorologis pada hari ini sangat dipengaruhi oleh kondisi atmosferik pada hari-hari sebelumnya. Karakteristik utama dari data cuaca sebagai data *time series* adalah adanya pola tren, musiman (*seasonality*), dan autokorelasi, di mana nilai pada waktu tertentu memiliki dependensi terhadap nilai pada langkah waktu lampau. Implikasi dari sifat sekuensial ini terhadap pemodelan *machine learning* sangatlah besar; model tidak hanya dituntut untuk mengenali hubungan antar-variabel secara independen, tetapi juga harus mampu menangkap dinamika perubahan cuaca dari waktu ke waktu [8].

Untuk mengeksploitasi informasi temporal ini, penelitian ini mengimplementasikan rekayasa fitur berbasis ***lag features*** (*autoregressive features*). *Lag features* adalah mekanisme penciptaan variabel fitur baru dengan cara menggeser nilai suatu variabel cuaca ke belakang sebesar d langkah waktu. Secara matematis, *lag feature* ke-d dari variabel meteorologi X_t didefinisikan sebagai:

$$X_{t-d} = \text{nilai variabel } X \text{ pada waktu } (t-d) \tag{2.1}$$

Dalam penelitian ini, $d = \{1, 2\}$, sehingga untuk setiap variabel cuaca harian (TAVG, RH_AVG, SS, RR) diciptakan dua *lag feature* yang merepresentasikan kondisi atmosfer satu dan dua hari sebelumnya. Pendekatan ini mentransformasi

masalah prediksi deret waktu menjadi masalah regresi/klasifikasi standar yang dapat diproses oleh algoritma *machine learning* berbasis pohon, tanpa harus menggunakan arsitektur *deep learning* rekuren [17]. Selain *lag features*, variabel fitur Bulan (*month*) juga ditambahkan sebagai representasi pola musiman monsun DKI Jakarta, di mana musim hujan aktif terjadi pada bulan Oktober–April (angin barat) dan musim kemarau pada bulan Mei–September (angin timur).

2.5 *Machine Learning*

Machine learning merupakan cabang utama dari kecerdasan buatan (*Artificial Intelligence*/AI) yang menitikberatkan pada pengembangan algoritma komputasional yang mampu belajar dari data historis, mengenali pola tersembunyi, dan membuat keputusan atau prediksi tanpa harus diprogram secara eksplisit untuk setiap kondisi [10]. Dalam penelitian terkait prediksi cuaca, pendekatan yang paling relevan adalah *supervised learning* atau pembelajaran terarah. Pada *supervised learning*, algoritma dilatih menggunakan himpunan data yang telah memiliki label target yang diketahui, misalnya label kejadian hujan aktual dari rekaman stasiun meteorologi BMKG. Peran *machine learning* dalam klasifikasi curah hujan sangatlah revolusioner dibandingkan dengan pendekatan peramalan cuaca tradisional yang berbasis pada persamaan fisika atmosferik. Model *machine learning*, khususnya algoritma berbasis pohon keputusan, mampu memproses dan mengekstraksi informasi dari data meteorologis multivariat yang sangat besar dalam waktu komputasi yang relatif singkat [19]. Keunggulan utama dari teknologi ini adalah kapabilitasnya dalam memodelkan hubungan non-linear dan interaksi kompleks antar-variabel pembentuk cuaca, sehingga mampu menghasilkan probabilitas prediksi kejadian hujan yang jauh lebih presisi dan dapat diandalkan untuk operasional mitigasi.

2.6 Algoritma *Random Forest*

Random Forest merupakan salah satu algoritma pembelajaran mesin berbasis *supervised learning* yang beroperasi menggunakan paradigma *ensemble learning*, khususnya melalui teknik *Bootstrap Aggregating* atau *Bagging* [8]. Paradigma *ensemble* secara fundamental bertujuan untuk menggabungkan banyak model prediktif sederhana (*base learners*) guna membentuk satu model komposit yang memiliki performa prediktif jauh lebih tangguh dan stabil. Fungsi prediktif

ensemble *Random Forest* untuk sebuah instans data x_i direpresentasikan sebagai penjumlahan dari seluruh pohon:

$$\hat{y}_i = \text{mode}\left(\{f_k(x_i)\}_{k=1}^{K}\right) \tag{2.2}$$

di mana K adalah jumlah total pohon dalam ensemble, dan f_k adalah fungsi prediksi dari pohon ke-k.

Secara teknis, algoritma *Random Forest* bekerja dengan cara membangun sejumlah K pohon keputusan secara paralel dan independen. Setiap pohon dilatih menggunakan subsampel data yang diambil secara acak dengan teknik pengembalian (*bootstrapping*) dari dataset asli. Selain pengacakan pada level data, algoritma ini juga menerapkan pengacakan pada level fitur, di mana hanya subset acak dari variabel meteorologi sebesar $\sqrt{p}$ (dengan p adalah total jumlah fitur) yang dievaluasi pada setiap proses pencabangan simpul. Untuk memisahkan kelas kejadian hujan dan tidak hujan pada setiap simpul, algoritma mengukur tingkat kemurnian informasi menggunakan kriteria *Gini Impurity*:

$$G(D) = 1 - \sum_{i=1}^{C} p_i^2 \tag{2.3}$$

Pada formula tersebut, C merepresentasikan jumlah kelas target (dalam penelitian ini $C = 2$: hujan dan tidak hujan), sedangkan p_i merupakan probabilitas empiris dari suatu observasi yang termasuk ke dalam kelas i pada simpul tertentu. Kelebihan utama dari *Random Forest* adalah ketahanannya yang sangat tinggi terhadap masalah *overfitting*, mengingat mekanisme *majority voting* pada akhir prediksi mampu secara efektif mereduksi variansi *error* tanpa meningkatkan bias [2]. Algoritma ini juga terbukti sangat stabil ketika dihadapkan pada dataset cuaca yang kompleks, non-linear, dan sarat akan pencilan (*outliers*) [14]. Meskipun demikian, *Random Forest* memiliki kelemahan berupa kebutuhan sumber daya komputasi yang relatif besar serta sifatnya yang menyerupai "kotak hitam", sehingga sulit diinterpretasikan secara langsung tanpa teknik analisis tambahan seperti SHAP.

2.7 Algoritma XGBoost

Extreme Gradient Boosting atau XGBoost adalah implementasi tingkat lanjut dari algoritma *gradient boosted decision trees* yang dirancang untuk menghadirkan kecepatan komputasi, skalabilitas, dan akurasi prediktif yang ekstrem [8]. Berbeda secara diametral dengan pendekatan paralel pada *Random Forest*, XGBoost mengadopsi konsep *Boosting*, di mana sejumlah pohon keputusan dibangun secara sekuensial. Setiap pohon baru didesain secara khusus untuk mengoreksi residu kesalahan prediksi dari kombinasi pohon-pohon pada iterasi sebelumnya. Fungsi prediksi komposit XGBoost didefinisikan sebagai:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathscr{F} \tag{2.4}$$

di mana $\mathscr{F}$ adalah ruang semua pohon keputusan yang memungkinkan. Optimalisasi pada XGBoost dicapai dengan meminimalkan fungsi objektif pada iterasi ke-t sebagai berikut:

$$\text{Obj}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \tag{2.5}$$

Fungsi l melambangkan fungsi kerugian (*loss function*) yang mengukur deviasi antara nilai aktual y_i dengan prediksi komposit $\hat{y}_i$. Suku $\Omega(f_t)$ mewakili fungsi regularisasi yang memberikan penalti terhadap kompleksitas pohon, sehingga model yang dihasilkan tetap sederhana dan memiliki variansi rendah. Untuk mempercepat pencarian titik optimal, XGBoost menerapkan ekspansi deret Taylor orde kedua yang melibatkan komputasi gradien (turunan pertama) dan hessian (turunan kedua) dari fungsi kerugian. Kelebihan operasional yang paling fundamental dari XGBoost adalah kemampuannya menangani nilai yang hilang (*missing values*) secara bawaan melalui algoritma *sparsity-aware split finding* [6]. Selain itu, kemampuan XGBoost dalam menangani ketidakseimbangan kelas melalui parameter `scale_pos_weight` menjadikannya sangat relevan untuk data cuaca yang imbalanced [11]. Melalui komputasi yang terdistribusi dan efisien, XGBoost menjadi algoritma mutakhir yang sangat diandalkan dalam klasifikasi data cuaca berdimensi tinggi untuk kebutuhan mitigasi bencana.

2.8 Akurasi dan Evaluasi Model

Evaluasi model merupakan serangkaian prosedur analitis yang esensial dalam pembelajaran mesin untuk mengukur sejauh mana sebuah algoritma mampu menggeneralisasi pola dari data latih menuju data uji yang belum pernah dilihat sebelumnya. Dalam kasus klasifikasi biner seperti penentuan kejadian hujan dan tidak hujan, kinerja model diekstraksi dari *Confusion Matrix* (Matriks Kebingungan) [1]. Matriks ini memetakan hasil prediksi ke dalam empat kuadran fundamental: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Berdasarkan empat elemen tersebut, metrik-metrik evaluasi diturunkan sebagai berikut:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.6}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.7}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.8}$$

$$F1\text{-}Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.9}$$

Dalam konteks EWS banjir, *Recall* merupakan metrik yang paling kritis karena mengukur kemampuan model dalam mendeteksi seluruh kejadian hujan yang benar-benar terjadi. Nilai *False Negative* yang tinggi berarti model gagal mengeluarkan peringatan saat hujan lebat benar-benar datang, sebuah konsekuensi yang jauh lebih berbahaya dibandingkan *False Positive* (alarm palsu) [8].

Selain keempat metrik di atas, penelitian ini juga menggunakan ***Receiver Operating Characteristic*–Area Under the Curve** (ROC-AUC) sebagai metrik evaluasi utama. Kurva ROC diperoleh dengan memplot *True Positive Rate* (TPR atau Recall) terhadap *False Positive Rate* (FPR) pada berbagai nilai ambang batas (*threshold*) klasifikasi. FPR didefinisikan sebagai:

$$FPR = \frac{FP}{FP + TN} \tag{2.10}$$

Nilai AUC merupakan integral dari kurva ROC yang merepresentasikan probabilitas bahwa model akan memberikan skor yang lebih tinggi kepada observasi positif (hari hujan) yang dipilih secara acak dibandingkan observasi negatif (hari tidak hujan). Rentang nilai AUC berkisar antara 0 hingga 1, dengan interpretasi kategori sebagai berikut: AUC $\geq$ 0.9 dikategorikan *Excellent*, AUC 0.8–0.9 dikategorikan *Good*, AUC 0.7–0.8 dikategorikan *Fair*, dan AUC $<$ 0.7 dikategorikan *Poor* [14]. Keunggulan ROC-AUC adalah sifatnya yang tidak bergantung pada pemilihan *threshold* tertentu dan tidak terpengaruh oleh ketidakseimbangan kelas, sehingga menjadikannya metrik yang paling objektif untuk membandingkan performa RF dan XGBoost pada data cuaca yang imbalanced.

2.9 Interpretabilitas Model: MDI dan SHAP

Interpretabilitas dalam terminologi pembelajaran mesin merujuk pada derajat sejauh mana seorang manusia khususnya para analis cuaca dan pemangku kebijakan dapat secara rasional memahami logika sebab-akibat di balik keputusan yang diotomasikan oleh suatu model prediktif [9]. Pada sistem EWS yang menyangkut keselamatan jiwa, akurasi komputasional yang tinggi tidak akan memiliki bobot operasional yang maksimal jika algoritma tersebut bertindak sebagai "kotak hitam".

2.9.1 *Mean Decrease in Impurity* (MDI)

Metode interpretabilitas pertama yang diimplementasikan adalah *Mean Decrease in Impurity* (MDI), yang merupakan mekanisme pengukuran kepentingan fitur (*feature importance*) bawaan dari algoritma berbasis pohon keputusan. MDI mengukur rata-rata penurunan *Gini Impurity* yang dikontribusikan oleh setiap fitur di seluruh pohon dalam *ensemble* ketika fitur tersebut digunakan sebagai variabel pemisah pada sebuah simpul. Secara matematis, nilai MDI untuk fitur j didefinisikan sebagai:

$$\text{MDI}(j) = \frac{1}{K} \sum_{k=1}^{K} \sum_{t \in T_k : v(t) = j} p(t) \cdot \Delta G(t) \tag{2.11}$$

di mana K adalah jumlah pohon, T_k adalah himpunan simpul pada pohon ke-k, $v(t)$ adalah fitur yang digunakan pada simpul t, $p(t)$ adalah proporsi observasi yang mencapai simpul t, dan $\Delta G(t)$ adalah penurunan *Gini Impurity* pada simpul tersebut. Seluruh nilai MDI kemudian dinormalisasi sehingga jumlahnya sama dengan 1.0. Fitur dengan nilai MDI tertinggi berkontribusi paling besar dalam daya diskriminatif model [3].

2.9.2 *SHapley Additive exPlanations* (SHAP)

Metode interpretabilitas kedua yang diimplementasikan adalah *SHapley Additive exPlanations* (SHAP), sebuah pendekatan yang didasarkan pada landasan teoritis yang lebih kuat dibandingkan MDI [9]. SHAP berakar pada konsep *Shapley values* dari teori permainan kooperatif (*cooperative game theory*), yang menyediakan cara yang adil dan aksiomatik untuk mendistribusikan "kredit" dari suatu prediksi ke masing-masing fitur yang berkontribusi. *Shapley value* untuk fitur j pada prediksi observasi x didefinisikan sebagai:

$$\phi_j(f, x) = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{j\}}(x_{S \cup \{j\}}) - f_S(x_S) \right] \tag{2.12}$$

di mana F adalah himpunan semua fitur, S adalah subset fitur yang tidak menyertakan fitur j, dan f_S adalah model yang dilatih hanya menggunakan fitur dalam S. Nilai ϕ_j yang positif mengindikasikan bahwa fitur j mendorong prediksi ke arah kelas Hujan, sedangkan nilai negatif mengindikasikan dorongan ke arah kelas Tidak Hujan. Implementasi efisien untuk model berbasis pohon dilakukan melalui algoritma *TreeSHAP* yang memiliki kompleksitas komputasi polinomial, jauh lebih cepat dibandingkan metode brute-force yang bersifat eksponensial [9]. Keunggulan SHAP dibandingkan MDI terletak pada kemampuannya memberikan interpretasi pada tingkat prediksi individual (*local explanation*), bukan hanya gambaran global, serta bebas dari bias kardinalitas yang menjadi kelemahan inheren MDI.

2.10 Variabel Penelitian (Meteorologi)

Variabel penelitian dalam pemodelan klasifikasi hujan berakar pada sekumpulan parameter meteorologis observasional yang saling berinteraksi secara termodinamika. Variabel primer yang digunakan dalam penelitian ini adalah suhu udara rata-rata (TAVG), kelembapan relatif rata-rata (RH_AVG), dan durasi sinar matahari (SS), yang seluruhnya diperoleh dari rekaman harian stasiun BMKG. Suhu udara memiliki pengaruh langsung terhadap kapasitas udara menahan uap air sesuai Hukum Clausius-Clapeyron; peningkatan suhu meningkatkan kapasitas tersebut, sedangkan penurunan suhu mendekati titik embun memicu kondensasi dan presipitasi [17]. Kelembapan relatif menduduki peran sentral: saat kelembapan menyentuh titik jenuh (mendekati 100%), uap air mengalami kondensasi, bertransformasi menjadi partikel penyusun awan, dan akhirnya jatuh sebagai curah hujan [8]. Durasi sinar matahari berfungsi sebagai proksi inversi tutupan awan; hari dengan durasi SS rendah mengindikasikan langit tertutup awan, yang merupakan prasyarat fisik terjadinya hujan. Variabel target biner (Target_Hujan) didefinisikan berdasarkan ambang batas curah hujan (RR) sebesar ≥ 0.5 mm/hari sesuai standar resmi BMKG untuk kategori "hari hujan".

2.11 Data *Splitting* pada *Time Series*

Pemecahan data (*data splitting*) merupakan protokol analitik wajib di mana sebuah dataset diisolasi menjadi himpunan pelatihan (*training set*) untuk menyusun bobot algoritma, dan himpunan pengujian (*testing set*) sebagai arena validasi untuk menguji efikasi generalisasi model. Pada studi klimatologi komputasi yang bersinggungan langsung dengan data berorientasi waktu, prosedur pemisahan menggunakan pengacakan baris data (*random shuffling*) atau validasi silang acak sangat dilarang secara metodologis [3]. Pengacakan akan mengeliminasi tatanan kronologis dan memicu insiden kebocoran data (*data leakage*), sebuah kondisi fatal di mana model mencuri informasi dari masa depan untuk mengenali pola di masa lalu [18].

Oleh karena itu, penelitian ini secara ketat mengaplikasikan teknik *time series splitting* yang berbasis pada indeks waktu temporal. Dataset dibagi secara berurutan, di mana blok waktu masa lampau (80% data awal berurutan) didedikasikan secara eksklusif untuk pelatihan, sementara blok waktu teraktual (20% data terakhir) digunakan murni untuk pengujian [7]. Dalam penelitian

ini, batas pembagian ditetapkan pada indeks $\lfloor 0.8 \times N \rfloor$ di mana N adalah total jumlah observasi, menghasilkan 1.485 baris data latih (Maret 2024 – Oktober 2025) dan 372 baris data uji (November 2025 – Maret 2026). Pentingnya menjaga dependensi temporal ini adalah untuk mensimulasikan kondisi peramalan cuaca yang sesungguhnya di lapangan, sehingga tingkat metrik performa yang dilaporkan oleh model merupakan representasi sahih dari ketangguhannya dalam memprediksi kejadian alam yang akan datang.