

Prediksi Curah Hujan di Kabupaten Sumenep Menggunakan Metode *Extreme Gradient Boosting* (XGBoost) dan Algoritma *Grid Search*

**M. THUFAIL ALWANNABIL SAMAS, NURISSAIDAH ULINNUHA*,
MOH. HAFIYUSHOLEH**

Department of Mathematics, Universitas Islam Negeri Sunan Ampel Surabaya, Indonesia
Email: nuris.ulinnuha@uinsby.ac.id

Received 19 Desember 2025 | *Revised* 16 Maret 2026 | *Accepted* 26 April 2026

ABSTRAK

Curah hujan di Kabupaten Sumenep merupakan variabel meteorologis penting karena memengaruhi kegiatan pertanian dan produksi garam. Penelitian ini bertujuan memprediksi curah hujan harian menggunakan metode Extreme Gradient Boosting (XGBoost) yang dioptimalkan dengan Grid Search. Variabel yang digunakan meliputi suhu udara, durasi sinar matahari, tekanan udara, kelembapan udara, kecepatan angin, dan penguapan. Data yang digunakan berupa data cuaca harian dari BMKG periode 1 Juli 2020 hingga 30 Juni 2024. Proses pemodelan meliputi preprocessing data, pembentukan fitur lag, pembagian data menggunakan Time Series Cross Validation dengan pendekatan expanding window, serta optimasi hyperparameter menggunakan Grid Search. Model dengan kombinasi hyperparameter terbaik menghasilkan MAAPE sebesar 0.9152 dan RMSE sebesar 11.9566.

Kata kunci: Curah Hujan, Grid Search, Kabupaten Sumenep, Prediksi, XGBoost

ABSTRACT

Rainfall in Sumenep Regency is an important meteorological variable because it affects agricultural activities and salt production. This study aims to predict daily rainfall using the Extreme Gradient Boosting (XGBoost) method optimized with Grid Search. The variables used include air temperature, sunshine duration, air pressure, air humidity, wind speed, and evaporation. The data used is daily weather data from BMKG for the period July 1, 2020, to June 30, 2024. The modeling process includes data preprocessing, lag feature formation, data division using Time Series Cross Validation with an expanding window approach, and hyperparameter optimization using Grid Search. The model with the best hyperparameter combination produced an MAAPE of 0.9152 and an RMSE of 11.9566.

Keywords: Rainfall, Grid Search, Sumenep Regency, Prediction, XGBoost

1. PENDAHULUAN

Indonesia merupakan negara kepulauan yang sebagian besar wilayahnya berupa perairan (**Faridah et al., 2022**). Salah satu potensi kelautan yang penting adalah air laut yang dapat dimanfaatkan sebagai bahan baku produksi garam. Garam memiliki peran strategis dalam berbagai sektor seperti pangan, farmasi, perminyakan, pengolahan air, serta industri kimia lainnya (**Sunoko et al., 2022**). Berdasarkan data Kementerian Kelautan dan Perikanan (KKP), Provinsi Jawa Timur menempati posisi pertama sebagai penghasil garam dengan produksi mencapai 399 ribu ton pada tahun 2023. Sebagian besar produksi tersebut berasal dari Pulau Madura, di mana Kabupaten Sumenep menjadi penyumbang terbesar dengan hasil mencapai 208,684.91 ton pada tahun yang sama (**Mahdi, 2024**).

Kebutuhan garam industri nasional terus meningkat sekitar 5–7% setiap tahun. Namun, produktivitas petambak garam sering terkendala oleh berkurangnya lahan, keterbatasan teknologi, dan faktor lingkungan. Proses produksi tradisional di Indonesia masih sangat bergantung pada sinar matahari untuk penguapan air laut, sehingga curah hujan menjadi faktor penentu utama. Perubahan iklim yang ekstrem menyebabkan peningkatan intensitas curah hujan yang berdampak langsung pada kuantitas dan kualitas garam yang dihasilkan (**Faqih et al., 2022**). Kondisi cuaca ideal untuk produksi garam meliputi kecepatan angin di atas 5 m/s, suhu udara lebih dari 32°C, penyinaran matahari maksimal, kelembapan udara rendah, serta curah hujan tahunan 1.000–1.300 mm (**Mashuri et al., 2021**). Ketidakpastian musim dan tingginya curah hujan semakin meningkatkan risiko penurunan produksi di Kabupaten Sumenep.

Untuk mengatasi dampak perubahan iklim tersebut, diperlukan solusi prediktif yang mampu memperkirakan curah hujan secara akurat. Salah satu pendekatan yang efektif adalah metode *machine learning*, yang dapat mengolah data *time series* dengan kompleksitas tinggi (**Anggarda et al., 2023**). Pendekatan ini memungkinkan analisis pola historis cuaca secara lebih akurat dan dapat digunakan untuk memprediksi tren curah hujan di masa mendatang (**Judijanto et al., 2024**).

Salah satu metode *machine learning* yang dapat digunakan dalam prediksi curah hujan adalah *Extreme Gradient Boosting* (XGBoost), yaitu algoritma berbasis pohon keputusan yang dikembangkan untuk menghasilkan performa tinggi pada data kompleks. Penelitian terdahulu menunjukkan bahwa XGBoost memiliki tingkat akurasi yang baik dalam prediksi data *time series*, seperti pada studi prediksi curah hujan di Kota Semarang yang menghasilkan nilai RMSE dan MAE masing-masing sebesar 2.7 dan 8.8 (**Anwar et al., 2021**). Penelitian lain yang membandingkan beberapa metode seperti CatBoost, LightGBM, Lasso, Ridge, dan *Linear Regression* juga menunjukkan bahwa XGBoost memberikan hasil terbaik dengan nilai RMSE dan MAE terkecil, yaitu 0.10 dan 0.02 (**Kumar et al., 2023**). Selain itu, penerapan teknik *Grid Search* dapat dilakukan untuk mencari kombinasi optimal *hyperparameter* pada model XGBoost, sehingga performa model dapat meningkat secara signifikan (**Li et al., 2024**). Penelitian lain menunjukkan bahwa penerapan *Grid Search* mampu meningkatkan akurasi model XGBoost dibandingkan dengan model tanpa optimasi. Hasil penelitian tersebut menunjukkan bahwa akurasi model tanpa optimasi sebesar 83.28%, sedangkan setelah dilakukan *Grid Search* meningkat menjadi 89.09% (**Pramudhyta & Rohman, 2024**).

Meskipun berbagai penelitian menunjukkan bahwa algoritma XGBoost dan teknik optimasi *Grid Search* mampu menghasilkan performa prediksi yang baik pada berbagai kasus data *time series*, penerapan metode tersebut untuk memprediksi curah hujan harian secara spesifik di Kabupaten Sumenep masih terbatas. Padahal wilayah ini merupakan salah satu sentra produksi garam terbesar di Indonesia yang sangat dipengaruhi oleh kondisi curah hujan. Oleh

karena itu, diperlukan penelitian yang mengkaji penerapan model prediksi curah hujan berbasis *machine learning* pada data meteorologi di wilayah tersebut agar dapat memberikan informasi yang lebih relevan bagi kegiatan produksi garam. Berdasarkan latar belakang tersebut, rumusan masalah dalam penelitian ini berkaitan dengan bagaimana memanfaatkan data meteorologi untuk memprediksi curah hujan harian di Kabupaten Sumenep menggunakan metode *machine learning*, bagaimana penerapan algoritma XGBoost yang dioptimalkan dengan teknik *Grid Search* dalam membangun model prediksi curah hujan, serta bagaimana performa model yang dihasilkan dalam memprediksi curah hujan pada wilayah penelitian.

Penelitian ini bertujuan untuk memprediksi curah hujan di Kabupaten Sumenep dengan menerapkan metode XGBoost yang dioptimalkan menggunakan algoritma *Grid Search*. Hasil penelitian ini diharapkan dapat memberikan informasi prediktif yang membantu petani garam dalam merencanakan waktu produksi secara lebih tepat sehingga risiko kerugian akibat ketidakpastian cuaca dapat diminimalkan. Selain itu, penelitian ini juga diharapkan dapat menjadi referensi dalam pengembangan model *machine learning* berbasis *boosting* untuk analisis data klimatologi di wilayah pesisir Indonesia. Penelitian ini berfokus pada pemodelan prediksi curah hujan harian di Kabupaten Sumenep menggunakan data meteorologi yang diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG). Variabel yang digunakan dalam penelitian ini meliputi beberapa parameter cuaca yang berkaitan dengan kondisi atmosfer, sedangkan variabel target yang diprediksi adalah curah hujan harian, dengan proses pemodelan menggunakan algoritma XGBoost yang dioptimalkan melalui teknik *Grid Search* dan dievaluasi menggunakan metrik MAAPE serta RMSE.

2. METODOLOGI PENELITIAN

2.1. Interpolasi Linier

Interpolasi linier adalah metode untuk memperkirakan nilai suatu variabel pada titik di antara dua titik data yang telah diketahui dengan mengasumsikan hubungan linier pada selang tersebut (**Larson et al., 2023**). Dengan kata lain, perubahan nilai di antara dua titik yang diketahui diasumsikan berlangsung secara proporsional mengikuti garis lurus. Persamaan (1) interpolasi linier dapat dinyatakan sebagai berikut:

$$f(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) \quad (1)$$

2.2. Normalisasi *Min-Max*

Normalisasi merupakan teknik yang digunakan untuk melakukan penskalaan atau transformasi nilai data agar berada pada rentang tertentu sebagai bagian dari proses *preprocessing* dalam *machine learning* (**Chairunisa et al., 2020**). Salah satu metode normalisasi yang paling umum digunakan adalah *min-max scaling*. Metode ini bekerja dengan cara memetakan setiap nilai data aktual ke dalam rentang 0 hingga 1, sehingga seluruh fitur memiliki skala yang sebanding dan tidak mendominasi proses pembelajaran model (**Suryanegara et al., 2021**). Persamaan (2) normalisasi min-max dapat dituliskan sebagai berikut:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

2.3. *Grid Search*

Grid Search merupakan metode pencarian *hyperparameter* yang digunakan untuk mengoptimalkan performa model *machine learning*. Teknik ini bekerja dengan cara mengevaluasi seluruh kombinasi nilai dari *hyperparameter* yang telah ditentukan dalam sebuah *grid* atau ruang pencarian. Dengan demikian, *Grid Search* memastikan bahwa semua

kombinasi *hyperparameter* diuji untuk menemukan konfigurasi terbaik berdasarkan metrik evaluasi tertentu (**Castro et al., 2024**).

Metode *Grid Search* sering digabungkan dengan *cross validation* untuk memastikan hasil optimasi *hyperparameter* tetap valid ketika diterapkan pada data baru yang belum pernah dilihat sebelumnya. Proses ini memungkinkan model *machine learning* memperoleh kombinasi *hyperparameter* yang optimal untuk menghasilkan prediksi yang lebih akurat (**El-Shahat et al., 2024**). Selain itu, *Grid Search* banyak digunakan karena kesederhanaan implementasinya dan kemampuannya dalam mengeksplorasi seluruh ruang *hyperparameter* secara sistematis. Metode ini dapat diterapkan secara fleksibel pada berbagai algoritma *machine learning*. Dengan demikian, pendekatan ini membantu peneliti memperoleh kombinasi *hyperparameter* terbaik untuk meningkatkan performa model secara optimal.

2.4. Time Series Cross Validation (TSCV)

Time Series Cross Validation adalah metode validasi model yang dirancang khusus untuk data berurutan atau data dengan ketergantungan temporal seperti data *time series*. Berbeda dengan *K-Fold Cross Validation* pada data independen, metode ini memastikan bahwa data *training* selalu berada di masa lalu relatif terhadap data *testing* untuk menjaga kausalitas. Pendekatan ini mencegah terjadinya *data leakage*, yaitu kondisi di mana informasi dari masa depan secara tidak sengaja digunakan untuk memprediksi masa lalu, yang dapat menghasilkan estimasi performa model yang bias (**Ajay et al., 2025**).

Dalam TSCV, dataset dibagi menjadi beberapa subset secara berurutan berdasarkan waktu. Data *testing* selalu diambil dari periode waktu setelah data *training* untuk menjaga urutan temporal. Salah satu pendekatan yang umum digunakan adalah *expanding window*, di mana ukuran data *training* bertambah secara bertahap untuk merepresentasikan ketersediaan data yang meningkat seiring waktu. Metode ini sangat berguna untuk mengevaluasi performa model dalam memprediksi data masa depan, terutama dalam aplikasi seperti *forecasting* cuaca, keuangan, dan analisis tren (**Torgbor et al., 2023**).

2.5. Extreme Gradient Boosting (XGBoost)

Extreme Gradient Boosting (XGBoost) merupakan algoritma *machine learning* berbasis pohon keputusan yang menggunakan pendekatan *boosting* secara bertahap untuk meningkatkan akurasi prediksi model. Metode ini menggabungkan beberapa model lemah (*weak learners*) menjadi satu model kuat (*strong learner*), di mana setiap model baru dibangun untuk memperbaiki kesalahan (*residual error*) dari model sebelumnya. XGBoost banyak digunakan karena kemampuannya dalam menangani data non-linier, efisiensi komputasi yang tinggi, serta fleksibilitas penerapannya pada berbagai permasalahan regresi dan klasifikasi (**Kiangala & Wang, 2021**).

Prediksi akhir yang dihasilkan oleh XGBoost merupakan penjumlahan dari seluruh pohon keputusan (*decision trees*) yang dibangun pada setiap iterasi. Prediksi akhir dapat dinyatakan pada Persamaan (3):

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i) \quad (3)$$

Untuk meminimalkan kesalahan prediksi, XGBoost menggunakan fungsi objektif (*objective function*) yang menggabungkan *loss function* dan *regularization term* guna mengontrol kompleksitas model. Fungsi objektif dapat dinyatakan pada Persamaan (4):

$$\mathcal{L}(\phi) = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (4)$$

Regularisasi pada XGBoost dirancang untuk mencegah *overfitting* dengan memberikan penalti terhadap ukuran dan kompleksitas model. Bentuk regularisasi dapat dinyatakan pada Persamaan (5):

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (5)$$

Keunggulan utama XGBoost terletak pada kemampuannya melakukan proses pembelajaran secara bertahap untuk meningkatkan akurasi prediksi. Algoritma ini mampu menghasilkan model yang efisien, stabil, dan adaptif terhadap pola data yang kompleks. Oleh karena itu, XGBoost banyak digunakan pada berbagai penelitian yang memerlukan model prediksi dengan performa tinggi.

2.6. Denormalisasi *Min-Max*

Denormalisasi merupakan proses kebalikan dari normalisasi, yaitu mengembalikan nilai data yang sebelumnya telah dinormalisasi ke skala aslinya (**Astutik & Syafii, 2024**). Metode ini bekerja dengan cara mentransformasi kembali nilai data yang semula berada pada rentang 0 hingga 1 setelah normalisasi, menjadi rentang nilai data aslinya sesuai dengan skala awal. Rumus denormalisasi *min-max scaling* dapat dinyatakan pada Persamaan (6):

$$x = x'(x_{max} - x_{min}) + x_{min} \quad (6)$$

2.7. Mean Arctangent Absolute Percentage Error (MAAPE)

Mean Arctangent Absolute Percentage Error (MAAPE) merupakan metrik evaluasi yang digunakan untuk mengukur tingkat *error* prediksi pada data regresi, yang merupakan pengembangan dari *Mean Absolute Percentage Error* (MAPE). MAAPE menggunakan fungsi *arctangent* untuk menangani nilai aktual yang mendekati nol, sehingga menghasilkan estimasi kesalahan yang lebih stabil dan *robust* dibandingkan MAPE (**Xu et al., 2024**). Rumus MAAPE dapat dinyatakan pada Persamaan (7):

$$MAAPE = \frac{1}{n} \sum_{i=1}^n \arctan \left(\left| \frac{y_i - \hat{y}_i}{y_i} \right| \right) \quad (7)$$

2.8. Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) merupakan metrik evaluasi yang digunakan untuk mengukur tingkat *error* pada model regresi. RMSE menghitung akar kuadrat dari rata-rata kuadrat selisih antara nilai aktual dan nilai hasil prediksi, sehingga memberikan ukuran kesalahan dalam satuan yang sama dengan data aslinya (**Sanjaya & Heksaputra, 2020**). Rumus RMSE dapat dinyatakan pada Persamaan (8):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (8)$$

2.9. Data Penelitian

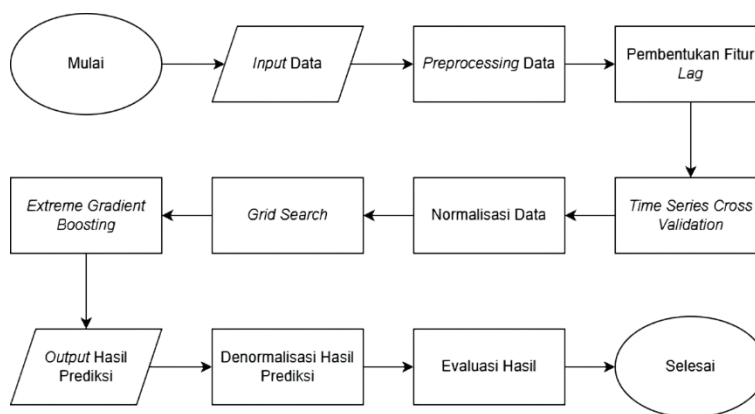
Data yang digunakan dalam penelitian ini adalah data sekunder variabel cuaca harian di Kabupaten Sumenep yaitu suhu udara rata-rata (x_1), durasi sinar matahari (x_2), tekanan udara rata-rata (x_3), kelembapan udara rata-rata (x_4), kecepatan angin rata-rata (x_5), dan penguapan (x_6) sebagai variabel bebas, sedangkan data curah hujan (y) sebagai variabel terikat. Data yang digunakan berjumlah 1461 data yang terhitung dari tanggal 01 Juli 2020 sampai 30 Juni 2024. Data didapatkan dari stasiun Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) Trunojoyo. Data penelitian dapat dilihat pada Tabel 1 berikut:

Tabel 1. Data Penelitian

No	Tanggal	x_1	x_2	x_3	x_4	x_5	x_6	y
1	01-07-2020	28.650	8.1	1010.0	74.50	3.9	5.3	0.0
2	02-07-2020	28.850	8.0	1010.5	73.25	3.6	5.6	0.0
3	03-07-2020	28.725	9.9	1011.4	69.00	4.1	3.4	0.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
730	30-06-2022	29.600	8.2	1008.5	72.25	3.9	3.7	0.0
731	01-07-2022	29.150	9.3	1008.7	71.75	4.3	5.3	0.0
732	02-07-2022	27.850	8.6	1008.1	86.00	2.6	5.2	1.6
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1459	28-06-2024	29.200	9.7	1011.2	78.50	2.3	5.0	0.0
1460	29-06-2024	28.800	10.2	1011.6	82.75	2.6	4.3	0.0
1461	30-06-2024	29.125	9.0	1010.2	72.50	1.6	6.4	0.0

2.10. Alur Penelitian

Langkah-langkah pada penelitian kali ini akan ditampilkan dalam bentuk diagram alir atau *flowchart* pada Gambar 1 berikut:



Gambar 1. Flowchart Penelitian

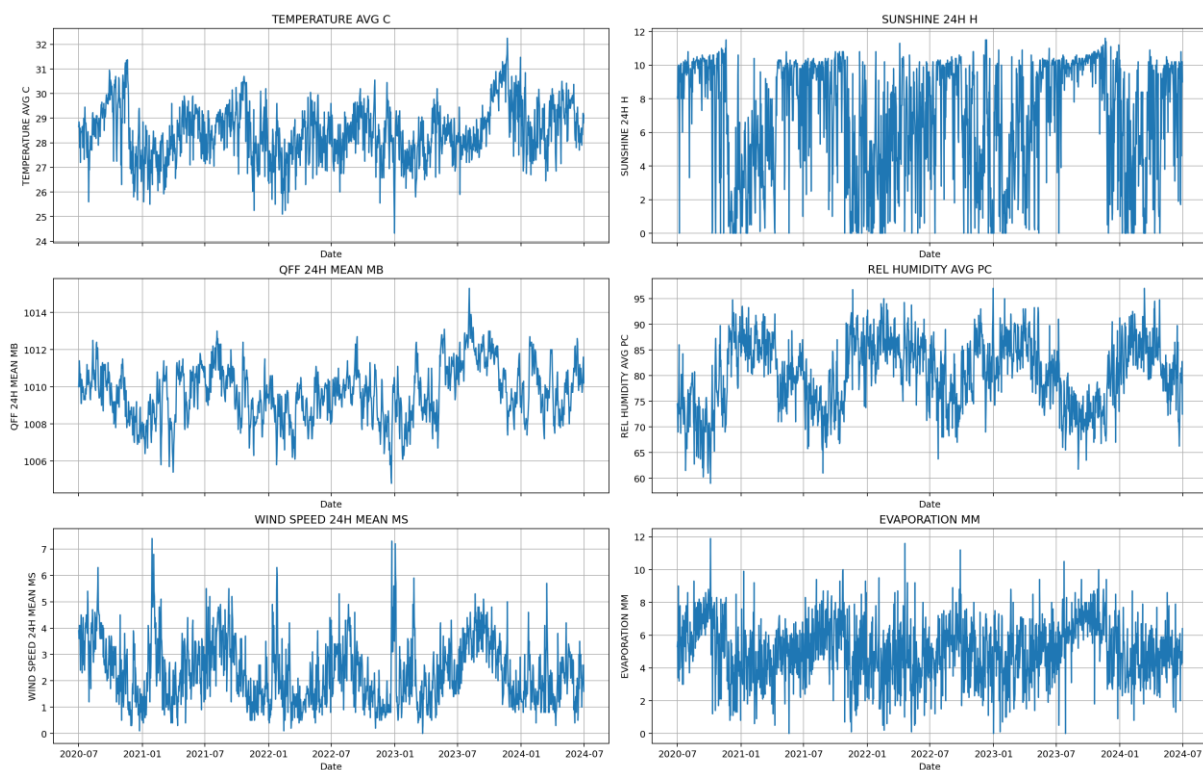
Proses dimulai dengan memasukkan data curah hujan sebagai variabel terikat dan data cuaca harian lain sebagai variabel bebas. Data curah hujan yang kosong diisi dengan nilai 0, sedangkan data kosong pada variabel bebas diisi menggunakan interpolasi linier. Nilai tidak terukur (8888 dan 9999) dikonversi menjadi 0 untuk variabel y dan dikonversi menjadi data kosong untuk variabel x karena tidak merepresentasikan kondisi aktual. Fitur *lag-1* kemudian dibentuk sehingga seluruh variabel pada hari sebelumnya digunakan untuk memprediksi curah hujan pada hari berikutnya. Pemilihan *lag-1* dilakukan karena penelitian ini berfokus pada prediksi curah hujan satu hari ke depan (*one-step ahead forecasting*), sehingga kondisi cuaca pada hari sebelumnya diasumsikan memiliki pengaruh langsung terhadap curah hujan pada hari berikutnya, meskipun eksplorasi beberapa nilai *lag* lain belum dilakukan secara sistematis

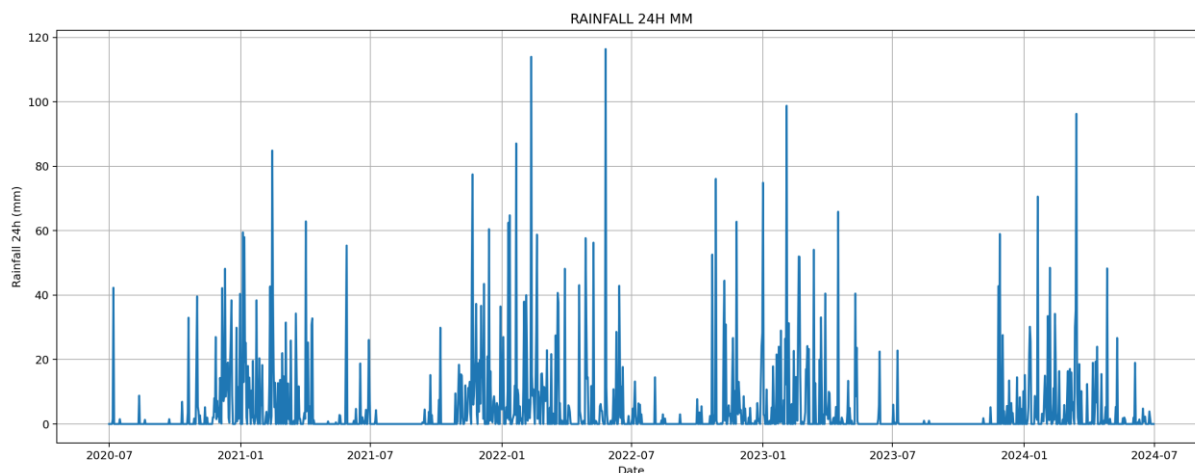
dan menjadi salah satu keterbatasan penelitian. Data kemudian dibagi menggunakan *Time Series Cross Validation* sebanyak 10 *fold* dengan pendekatan *expanding window* untuk menjaga urutan temporal data sehingga evaluasi model lebih merepresentasikan kondisi prediksi pada data masa depan. Pada setiap *fold*, data *training* dinormalisasi menggunakan *min-max scaling*, sementara data *testing* dinormalisasi menggunakan nilai *min* dan *max* dari data *training* untuk mencegah *data leakage*. Selanjutnya *Grid Search* digunakan untuk melakukan *hyperparameter tuning* pada model XGBoost agar diperoleh kombinasi parameter yang optimal. Hasil prediksi kemudian dinormalisasi dan dievaluasi menggunakan MAAPE dan RMSE serta divisualisasikan untuk analisis performa model secara komprehensif.

3. HASIL DAN PEMBAHASAN

3.1. *Preprocessing Data*

Data pada penelitian ini memiliki beberapa nilai kosong dan nilai tidak terukur, sehingga diperlukan *preprocessing* data sebelum dilakukan pemodelan. Pada variabel x , data kosong diisi menggunakan metode interpolasi linier. Selanjutnya pada variabel y , data kosong diisi dengan nilai 0 karena dianggap tidak ada hujan di hari tersebut. Nilai 8888 dan 9999 yang muncul pada variabel y diubah menjadi 0 sebagai penanda bahwa curah hujan pada hari tersebut tidak terukur dan bukan merupakan nilai yang sebenarnya. Selanjutnya pada variabel x , nilai 8888 dan 9999 dianggap sebagai data kosong dan diisi menggunakan metode interpolasi linier. Dalam dataset meteorologi BMKG, nilai 8888 dan 9999 umumnya digunakan sebagai kode untuk data yang tidak tercatat atau tidak terukur sehingga tidak merepresentasikan kondisi meteorologis aktual. Oleh karena itu, nilai tersebut perlu ditangani pada tahap *preprocessing* agar tidak mempengaruhi proses pembelajaran model. Berikut adalah visualisasi *plot* data setelah dilakukan *preprocessing* data.





Gambar 2. Visualisasi *Plot* Data setelah Dilakukan *Preprocessing* Data

Berdasarkan visualisasi *plot* pada Gambar 2, variabel curah hujan menunjukkan fluktuasi musiman yang cukup tinggi dengan banyak nilai nol serta beberapa puncak ekstrem. Pola tersebut menegaskan adanya ketergantungan waktu yang kuat pada data. Untuk memberikan gambaran kuantitatif mengenai karakteristik masing-masing variabel setelah dilakukan *preprocessing* data, dilakukan analisis statistika deskriptif sebagaimana ditunjukkan pada Tabel 2 berikut:

Tabel 2. Statistika Deskriptif

	x_1	x_2	x_3	x_4	x_5	x_6	y
count	1461	1461	1461	1461	1461	1461	1461
mean	28.420	6.8	1009.7	80.38	2.3	5.0	4.8
std	1.115	3.5	1.5	6.83	1.2	1.9	12.4
min	24.325	0.0	1004.8	59.00	0.0	0.0	0.0
25%	27.675	4.0	1008.6	75.00	1.3	3.8	0.0
50%	28.350	8.0	1009.7	81.00	2.2	5.0	0.0
75%	29.175	10.0	1010.7	85.50	3.2	6.3	2.8
max	32.250	11.6	1015.3	97.00	7.4	11.9	116.4

Berdasarkan tabel di atas, jumlah data pada seluruh variabel telah sama, yaitu 1461 observasi, yang menandakan tidak terdapat data kosong. Nilai maksimum yang tidak lagi menunjukkan angka 8888 maupun 9999 juga mengindikasikan bahwa seluruh nilai tak terukur telah berhasil ditangani melalui *preprocessing* data. Sebagian besar variabel menunjukkan variasi harian yang relatif stabil, kecuali curah hujan (y) yang memiliki distribusi sangat tidak merata dengan nilai median sebesar 0, lebih dari separuh data curah hujan bernilai 0 mm. Kondisi ini menunjukkan bahwa data curah hujan sangat tidak merata dengan didominasi hari tanpa hujan, namun disertai beberapa hari dengan intensitas hujan yang tinggi.

3.2. Pembentukan Fitur *Lag*

Pembentukan fitur *lag* dilakukan dengan menggeser nilai seluruh variabel bebas dan variabel terikat (x_1, x_2, \dots, x_6, y) pada hari sebelumnya ke hari berikutnya. Proses ini bertujuan untuk menangkap ketergantungan temporal pada data dan sesuai dengan karakteristik data *time series*, di mana kondisi pada hari sebelumnya dapat memengaruhi kondisi hari berikutnya. Pada penelitian ini digunakan *lag-1*, sehingga setiap baris data berisi nilai semua variabel pada hari sebelumnya beserta nilai curah hujan pada hari berikutnya. Hasil pembentukan fitur *lag-1* ditampilkan pada Tabel 3 berikut:

Tabel 3. Hasil Pembentukan Fitur *Lag-1*

No	Tanggal	$x_{1(t-1)}$	$x_{2(t-1)}$	$x_{3(t-1)}$	$x_{4(t-1)}$	$x_{5(t-1)}$	$x_{6(t-1)}$	$y_{(t-1)}$	y
1	02-07-2020	28.650	8.1	1010.0	74.50	3.9	5.3	0.0	0.0
2	03-07-2020	28.850	8.0	1010.5	73.25	3.6	5.6	0.0	0.0
3	04-07-2020	28.725	9.9	1011.4	69.00	4.1	3.4	0.0	0.0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
730	01-07-2022	29.60	8.2	1008.5	72.25	3.9	3.7	0.0	0.0
731	02-07-2022	29.15	9.3	1008.7	71.75	4.3	5.3	0.0	1.6
732	03-07-2022	27.85	8.6	1008.1	86.00	2.6	5.2	1.6	4.8
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1458	28-06-2024	28.675	4.6	1010.6	81.50	2.0	4.2	0.0	0.0
1459	29-06-2024	29.200	9.7	1011.2	78.50	2.3	5.0	0.0	0.0
1460	30-06-2024	28.800	10.2	1011.6	82.75	2.6	4.3	0.0	0.0

3.3. Time Series Cross Validation

Data yang sudah melalui proses pembentukan fitur *lag* akan dibagi menggunakan metode *Time Series Cross Validation* dengan pendekatan *expanding window* sebanyak 10-*fold*. Pada setiap *fold*, jumlah data *testing* dibuat sama, sedangkan data *training* akan bertambah secara bertahap sebesar jumlah data *testing* pada setiap iterasi. Hasil pembagian data menggunakan TSCV ditampilkan pada Tabel 4 berikut:

Tabel 4. Hasil Pembagian Data Menggunakan *Time Series Cross Validation*

<i>Fold</i>	Jumlah Data <i>Training</i>	Jumlah Data <i>Testing</i>
1	140 (02/07/2020 – 18/11/2020)	132 (19/11/2020 – 30/03/2021)
2	140 + 132 = 272 (02/07/2020 – 30/03/2021)	132 (31/03/2021 – 09/08/2021)
3	272 + 132 = 404 (02/07/2020 – 09/08/2021)	132 (10/08/2021 – 19/12/2021)
⋮	⋮	⋮
8	932 + 132 = 1064 (02/07/2020 – 31/05/2023)	132 (01/06/2023 – 10/10/2023)
9	1064 + 132 = 1196 (02/07/2020 – 10/10/2023)	132 (11/10/2023 – 19/02/2024)
10	1196 + 132 = 1328 (02/07/2020 – 19/02/2024)	132 (20/02/2024 – 30/06/2024)

3.4. Normalisasi Data

Data yang telah dibagi menggunakan *Time Series Cross Validation* akan dinormalisasi pada setiap *fold* menggunakan metode *min-max scaling*. Proses normalisasi dilakukan berdasarkan nilai *min* dan *max* dari data *training* di setiap *fold*, kemudian nilai tersebut juga digunakan untuk menormalisasi data *testing* agar skala kedua dataset tetap konsisten dan menghindari *data leakage*. Karena proses normalisasi pada setiap *fold* dilakukan dengan pola yang sama, maka pada bagian ini hanya ditampilkan hasil normalisasi pada *fold* ke-1 sebagai contoh. Hasil normalisasi pada data *training* dan *testing* pada *fold* ke-1 ditunjukkan berturut-turut pada Tabel 5 dan Tabel 6 berikut:

Tabel 5. Hasil Normalisasi pada Data *Training* di *Fold* ke-1

No	Tanggal	$x_{1(t-1)}$	$x_{2(t-1)}$	$x_{3(t-1)}$	$x_{4(t-1)}$	$x_{5(t-1)}$	$x_{6(t-1)}$	$y_{(t-1)}$	y
1	02-07-2020	0.5304	0.7364	0.4681	0.5041	0.5862	0.3832	0.0000	0.0000
2	03-07-2020	0.5652	0.7273	0.5745	0.4634	0.5345	0.4112	0.0000	0.0000
3	04-07-2020	0.5435	0.9000	0.7660	0.3252	0.6207	0.2056	0.0000	0.0000
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
138	16-11-2020	0.8087	0.8091	0.4255	0.6016	0.3103	0.3271	0.0473	0.0000
139	17-11-2020	1.0000	0.7091	0.1915	0.3252	0.4310	0.6355	0.0000	0.0000
140	18-11-2020	0.9739	1.0000	0.0000	0.3496	0.4138	0.6355	0.0000	0.0000

Tabel 6. Hasil Normalisasi Pada Data *Testing* di *Fold* ke-1

No	Tanggal	$x_{1(t-1)}$	$x_{2(t-1)}$	$x_{3(t-1)}$	$x_{4(t-1)}$	$x_{5(t-1)}$	$x_{6(t-1)}$	$y_{(t-1)}$	y
141	19-11-2020	0.9043	0.9636	0.0426	0.3659	0.4655	0.4860	0.0000	0.0000
142	20-11-2020	1.0043	1.0455	0.0851	0.3740	0.2931	0.6636	0.0000	0.0000
143	21-11-2020	0.8609	0.7909	0.0638	0.4878	0.1034	0.4206	0.0000	0.0000
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
270	28-03-2021	0.6957	0.7273	0.0000	0.6992	0.1552	0.3645	0.0000	0.0000
271	29-03-2021	0.4957	0.6818	-0.0638	0.7967	0.0690	0.2430	0.0000	0.0000
272	30-03-2021	0.5391	0.6455	-0.2766	0.8211	-0.0172	0.2150	0.0000	0.0473

3.5. XGBoost dan *Grid Search*

Data yang telah dinormalisasi pada setiap *fold* kemudian digunakan pada tahap *hyperparameter tuning* untuk memperoleh kombinasi *hyperparameter* terbaik pada model XGBoost. Proses ini dilakukan menggunakan metode *Grid Search*, yang secara sistematis menguji seluruh kombinasi nilai *hyperparameter* berdasarkan kinerja model di setiap *fold*. Evaluasi dilakukan menggunakan metrik MAAPE dan RMSE yang dihitung setelah hasil prediksi dinormalisasi ke skala aslinya. Hasil *hyperparameter tuning* secara lengkap dapat dilihat pada Tabel 7 berikut:

Tabel 7. Hasil *Hyperparameter Tuning* Menggunakan *Grid Search*

<i>Hyperparameter</i>	Nilai	Nilai Terbaik	MAAPE	RMSE
<i>colsample_bytree</i>	0.6, 0.8, 1.0	0.8	0.9152	11.9566
<i>learning_rate</i>	0.05, 0.1, 0.3	0.3		
<i>max_depth</i>	4, 6, 8	4		
<i>n_estimators</i>	100, 300, 500	500		
<i>subsample</i>	0.6, 0.8, 1.0	0.6		

Sebagai pembandingan, model XGBoost juga dilatih menggunakan *default hyperparameter*, yang menghasilkan nilai MAAPE sebesar 1.0041 dan RMSE sebesar 12.7680. Hasil tersebut menunjukkan bahwa proses *tuning* mampu meningkatkan performa model, ditunjukkan oleh nilai MAAPE dan RMSE yang lebih rendah dibandingkan model tanpa optimasi. Dengan demikian, model XGBoost hasil *hyperparameter tuning* menghasilkan estimasi curah hujan yang lebih mendekati data aktual, baik dari sisi kesalahan relatif maupun kesalahan dalam satuan milimeter. Kombinasi *hyperparameter* terbaik yang diperoleh pada tahap ini kemudian digunakan untuk melatih model akhir XGBoost dalam melakukan prediksi curah hujan di Kabupaten Sumenep.

3.6. Hasil Prediksi dan Visualisasi *Plot*

Hasil kombinasi *hyperparameter* terbaik yang didapatkan dari *Grid Search* akan digunakan untuk melatih ulang model XGBoost agar menghasilkan prediksi curah hujan. Hasil prediksi yang diperoleh kemudian dinormalisasi kembali ke skala aslinya (mm) agar dapat dibandingkan langsung dengan data aktual. Perbandingan antara data aktual dan hasil prediksi ditampilkan pada Tabel 8 berikut:

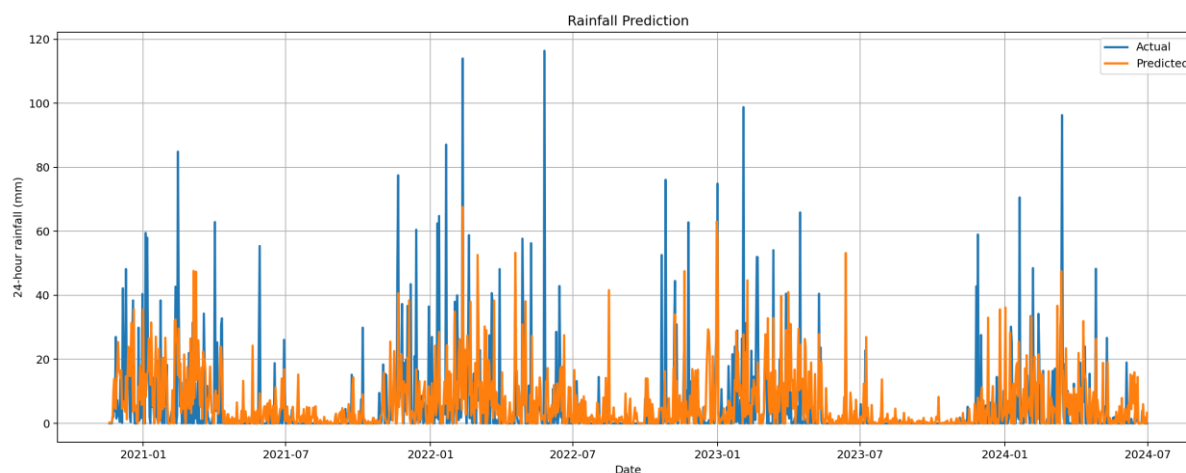
Tabel 8. Hasil Prediksi Curah Hujan

No	Tanggal	Data Aktual	Hasil Prediksi
141	19-11-2020	0.0	0.2179
142	20-11-2020	0.0	0.1038
143	21-11-2020	0.0	0.1855
⋮	⋮	⋮	⋮

Prediksi Curah Hujan di Kabupaten Sumenep Menggunakan Metode *Extreme Gradient Boosting* (XGBoost) dan Algoritma *Grid Search*

No	Tanggal	Data Aktual	Hasil Prediksi
800	08-09-2022	0.0	1.6337
801	09-09-2022	0.0	0.4072
802	10-09-2022	0.0	0.0000
⋮	⋮	⋮	⋮
1458	28-06-2024	0.0	1.8516
1459	29-06-2024	0.0	0.0000
1460	30-06-2024	0.0	3.3022

Berikutnya, hasil prediksi yang telah dinormalisasi divisualisasikan dalam bentuk *plot* untuk menunjukkan pola perbandingan antara nilai aktual dan hasil prediksi model, sebagaimana ditampilkan pada Gambar 3.



Gambar 3. Visualisasi *Plot* Perbandingan Data Aktual dan Hasil Prediksi

Dari gambar di atas dapat dilihat bahwa hasil prediksi model XGBoost mampu mengikuti pola fluktuasi curah hujan aktual dengan cukup baik, terutama pada periode hujan dengan intensitas ringan hingga sedang. Model juga mampu mengenali pola musiman yang berulang setiap tahun, meskipun masih terdapat beberapa deviasi pada nilai puncak curah hujan ekstrem. Kondisi ini menunjukkan bahwa pendekatan berbasis *boosting* yang digunakan oleh XGBoost efektif dalam mempelajari hubungan non-linier antar variabel meteorologi terhadap curah hujan. Secara keseluruhan, penerapan XGBoost yang dioptimalkan melalui *Grid Search* mampu meningkatkan generalisasi model, menekan nilai kesalahan prediksi, dan menghasilkan estimasi curah hujan yang mengikuti pola musiman pada data aktual. Hasil ini juga sejalan dengan beberapa penelitian sebelumnya yang menunjukkan bahwa algoritma XGBoost memiliki performa yang baik dalam memodelkan data *time series* meteorologi yang kompleks. Dengan demikian, model ini dapat dikatakan cukup baik untuk memprediksi curah hujan di Kabupaten Sumenep.

4. KESIMPULAN

Penerapan metode XGBoost dengan *Grid Search* pada prediksi curah hujan di Kabupaten Sumenep menghasilkan model terbaik dengan kombinasi *hyperparameter* yaitu *colsample_bytree* = 0.8, *learning_rate* = 0.3, *max_depth* = 4, *n_estimators* = 500, dan *subsample* = 0.6. Model tersebut menghasilkan nilai MAAPE sebesar 0.9152 dan RMSE sebesar 11.9566, yang menunjukkan bahwa model mampu memberikan estimasi curah hujan dengan performa prediksi yang lebih baik dibandingkan model tanpa optimasi. Informasi prediksi curah

hujan yang dihasilkan dari model ini dapat dimanfaatkan sebagai bahan pertimbangan bagi petani garam maupun pemangku kebijakan daerah dalam menentukan waktu produksi dan pengelolaan tambak garam agar risiko kerugian akibat hujan dapat diminimalkan. Hasil ini mengindikasikan bahwa model cukup baik dalam mempelajari pola curah hujan yang fluktuatif dan tidak merata. Untuk penelitian selanjutnya, performa model dapat ditingkatkan dengan menambahkan fitur meteorologis lain, menambahkan nilai *hyperparameter* lain, menggunakan *plot ACF* dan *PACF* untuk menentukan fitur *lag*, dan melakukan transformasi logaritma untuk mengatasi jauhnya rentang nilai pada data curah hujan.

DAFTAR RUJUKAN

- Ajay, B. S., Pavan K, P., & Rao, M. (2025). MC-QDSNN: Quantized Deep evolutionary SNN with Multi-Dendritic Compartment Neurons for Stress Detection using Physiological Signals. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *44*(4), 1–13. <https://doi.org/10.1109/TCAD.2024.3484353>
- Anggarda, M. F., Kustiawan, I., Nurjannah, D. R., & Hakim, N. F. A. (2023). Pengembangan Sistem Prediksi Waktu Penyiraman Optimal pada Perkebunan: Pendekatan Machine Learning untuk Peningkatan Produktivitas Pertanian. *Jurnal Budidaya Pertanian*, *19*(2), 124–136. <https://doi.org/10.30598/jbdp.2023.19.2.124>
- Anwar, M. T., Winarno, E., Hadikurniawati, W., & Novita, M. (2021). Rainfall Prediction Using Extreme Gradient Boosting. *Journal of Physics: Conference Series*, *1869*(1), 1–5. <https://doi.org/10.1088/1742-6596/1869/1/012078>
- Astutik, L. Y., & Syafii, I. (2024). Penerapan Artificial Neural Network untuk Memprediksi Error dalam Perancangan Aplikasi Monitoring Tetes Cairan Infus. *Jurnal Media Informatika Budidarma*, *8*(3), 1408–1418. <https://doi.org/10.30865/mib.v7i3.6368>
- Castro, L. G., Chávez, M., Dufлот, P., Bleret, V., Fiol, G. Del, & López-Nores, M. (2024). Impact of Hyperparameter Optimization to Enhance Machine Learning Performance: A Case Study on Breast Cancer Recurrence Prediction. *Applied Sciences*, *14*(13), 1–9. <https://doi.org/10.3390/app14135909>
- Chairunisa, R., Adiwijaya, & Astuti, W. (2020). Perbandingan CART dan Random Forest untuk Deteksi Kanker Berbasis Klasifikasi Data Microarray. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *4*(5), 805–812. <https://doi.org/10.29207/resti.v4i5.2083>
- El-Shahat, D., Tolba, A., Abouhawwash, M., & Abdel-Basset, M. (2024). Machine Learning and Deep Learning Models Based Grid Search Cross Validation for Short-term Solar Irradiance Forecasting. *Journal of Big Data*, *11*(134), 1–39. <https://doi.org/10.1186/s40537-024-00991-w>
- Faqih, Rodiaminollah, M., & Kodariyah, L. (2022). Analisis Penentuan Harga Pokok Produksi Garam dalam Menetapkan Harga Jual Menurut Perspektif Islam. *IQTISODINA: Jurnal*

- Ekonomi Syariah Dan Hukum Islam*, 5(1), 29–52.
<https://doi.org/10.35127/iqtisodina.v5i1.5806>
- Faridah, A., Kurniawan, A., & Dewi, A. K. (2022). Perancangan Film Pendek sebagai Media Kampanye Wisata Historical Sailing and Conservation di Kepulauan Seribu kepada Generasi Z. *FAD*, 1(1), 1–13. <https://eproceeding.itenas.ac.id/index.php/fad/article/view/1139/1077>
- Judijanto, L., Amin, A., & Nurhakim, L. (2024). Implementasi Teknologi Artificial Intelligence dan Machine Learning dalam Praktik Akuntansi dan Audit: Sebuah Revolusi atau Evolusi. *COSMOS: Jurnal Ilmu Pendidikan, Ekonomi Dan Teknologi*, 1(6), 470–483. <https://doi.org/10.35194/ajaki.v4i1.5032>
- Kiangala, S. K., & Wang, Z. (2021). An Effective Adaptive Customization Framework for Small Manufacturing Plants Using Extreme Gradient Boosting-XGBoost and Random Forest Ensemble Learning Algorithms in an Industry 4.0 Environment. *Machine Learning with Applications*, 4(100024), 1–15. <https://doi.org/10.1016/j.mlwa.2021.100024>
- Kumar, V., Kedam, N., Sharma, K. V., Khedher, K. M., & Alluqmani, A. E. (2023). A Comparison of Machine Learning Models for Predicting Rainfall in Urban Metropolitan Cities. *Sustainability*, 15(18), 1–27. <https://doi.org/10.3390/su151813724>
- Larson, D. M., Bungula, W., Lee, A., Stockdill, A., McKean, C., Miller, F. F., Davis, K., Erickson, R. A., & Hlavacek, E. (2023). Reconstructing Missing Data by Comparing Interpolation Techniques: Applications for Long-term Water Quality Data. *Limnology and Oceanography: Methods*, 21(7), 435–449. <https://doi.org/10.1002/lom3.10556>
- Li, W., Peng, Y., & Peng, K. (2024). Diabetes Prediction Model Based on GA-XGBoost and Stacking Ensemble Algorithm Wenguang. *PLoS ONE*, 19(9), 1–29. <https://doi.org/10.1371/journal.pone.0311222>
- Mahdi, I. (2024). *Produksi Garam Rakyat Sumenep Tahun 2023 Capai 208 Ribu Ton*. Kabar Madura. <https://kabarmadura.id/produksi-garam-rakyat-sumenep-tahun-2023-capai-208-ribu-ton/>
- Mashuri, Losu, H. Z., Nurhadi, H., Hakim, M. L., & Sampurno, B. (2021). Perancangan Sistem Model Scale Alat Pencegah Bercampurnya Air Hujan dengan Air Laut Menggunakan Sistem Kontrol Otomatis Sensor Suhu guna Menjaga Kestabilan Produksi Garam pada Musim Hujan. *Jurnal Nasional Aplikasi Mekatronika, Otomasi Dan Robot Industri (AMORI)*, 2(1), 22–28. <https://doi.org/10.12962/j27213560.v2i1.9125>
- Pramudhyta, N. A., & Rohman, M. S. (2024). Perbandingan Optimasi Metode Grid Search dan Random Search dalam Algoritma XGBoost untuk Klasifikasi Stunting. *Jurnal Media Informatika Budidarma*, 8(1), 19–29. <https://doi.org/10.30865/mib.v8i1.6965>

- Sanjaya, F. I., & Heksaputra, D. (2020). Prediksi Rerata Harga Beras Tingkat Grosir Indonesia dengan Long Short Term Memory. *Jurnal Teknik Informatika Dan Sistem Informasi*, 7(2), 163–174. <https://doi.org/10.35957/jatisi.v7i2.388>
- Sunoko, R., Saefuddin, A., Syarief, R., & Zulbainarni, N. (2022). Proteksionisme dan Standardisasi Garam Konsumsi Beryodium. *Jurnal Kebijakan Sosial Ekonomi Kelautan Dan Perikanan*, 12(2), 101–111. <http://dx.doi.org/10.15578/jksekp.v12i2.11077>
- Suryanegara, G. A. B., Adiwijaya, & Purbolaksono, M. D. (2021). Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(1), 114–122. <https://doi.org/10.29207/resti.v5i1.2880>
- Torgbor, B. A., Rahman, M. M., Brinkhoff, J., Sinha, P., & Robson, A. (2023). Integrating Remote Sensing and Weather Variables for Mango Yield Prediction Using a Machine Learning Approach. *Remote Sensing*, 15(12), 1–26. <https://doi.org/10.3390/rs15123075>
- Xu, K., Zhang, J., Huang, J., Tan, H., Jing, X., & Zheng, T. (2024). Forecasting Visitor Arrivals at Tourist Attractions: A Time Series Framework with the N-BEATS for Sustainable Tourism. *Sustainability*, 16(18), 1–28. <https://doi.org/10.3390/su16188227>