Jurnal Pendidikan dan Teknologi Indonesia (JPTI)
Vol. 5, No. 9, September 2025, Hal. 2478-2489
DOI: https://doi.org/10.52436/1.jpti.967
p-ISSN: 2775-4227
e-ISSN: 2775-4219

Perbandingan Algoritma Generalized Linear Model Dan Linear Regression Untuk Prediksi Hujan Berbasis Data Kaggle

Muhamad Anggi Irawan[*1], Ayu Ratna Juwita[2], Elsa Elvira Awal[3], Tatang Rohana[4]

[1,2,3,4]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Buana Perjuangan Karawang, Indonesia
Email: [1]if21.muhamadirawan@mhs.ubpkarawang.ac.id, [2]ayurj@ubpkarawang.ac.id, [3]elsaelvira@ubpkarawang.ac.id, [4]tatang.rohana@ubpkarawang.ac.id

Abstrak

Prediksi curah hujan sangat penting bagi berbagai aktivitas yang dipengaruhi kondisi cuaca, khususnya di negara beriklim tropis mengalami kondisi ini secara signifikan. Prediksi curah hujan yang akurat sangat penting untuk mendukung berbagai aspek perencanaan kota, termasuk pengelolaan sumber daya air dan mitigasi risiko bencana banjir. Penelitian ini membandingkan dua algoritma *machine learning*, *Generalized Linear Model* (GLM) dan *Linear Regression*, dalam memprediksi curah hujan berdasarkan fitur cuaca seperti suhu, kelembaban, tekanan, angin, tutupan awan, dan data historis. Selanjutnya diproses melalui *encoding* yang dimana akan mengubah nilai kategorikal menjadi nilai numerik, normalisasi yang melibatkan penyesuaian ulang nilai nilai dalam dataset, dan penanganan *class imbalance* untuk melakukan duplikasi sample pada kelas minotitas. Setelah dibagi menjadi data latih dan uji, kedua algoritma diterapkan dan dievaluasi menggunakan akurasi, RMSE, dan MAE. Hasilnya, GLM memiliki akurasi sebesar 90.17% lalu untuk RMSE sebesar 0.3949 dan MAE 0.3836, se dangkan *Linear Regression* lebih baik dalam nilai MAE sebesar 0.2656 dan RMSE 0.3218 untuk akurasi sebesar 89.26%. Dengan pendekatan analisis yang tepat, pola tersebut dapat dimanfaatkan untuk mendukung keputusan dan perencanaan secara lebih terarah.

Kata kunci: *Curah Hujan, Evaluasi Model, Generalized Linear Model, Linear Regression, Machine Learning, Prediksi Hujan, Preprocessing Data.*

Comparison of Generalized Linear Model Algorithms and Linear Regression for Rain Prediction

Abstract

Rainfall prediction is very important for various activities that are influenced by weather conditions, especially in tropical countries that experience this condition significantly. Accurate rainfall prediction is very important to support various aspects of city planning, including water resource management and flood disaster risk mitigation. This study compares two machine learning algorithms, Generalized Linear Model (GLM) and Linear Regression, in predicting rainfall based on weather features such as temperature, humidity, pressure, wind, cloud cover, and historical data. Furthermore, it is processed through encoding which will change categorical values into numeric values, normalization which involves readjusting the values in the dataset, and handling class imbalance to duplicate samples in the minority class. After being divided into training and test data, both algorithms are applied and evaluated using accuracy, RMSE, and MAE. As a result, GLM has an accuracy of 90.17% then for RMSE of 0.3949 and MAE of 0.3836, while Linear Regression is better in MAE values of 0.2656 and RMSE of 0.3218 for an accuracy of 89.26%. With the right analysis approach, the pattern can be used to support decisions and planning in a more targeted manner.

Keywords: *Data Preprocessing, Generalized Linear Model, Linear Regression, Machine Learning, Model Evaluation, Rain Prediction, Rainfall.*

1. PENDAHULUAN

Cuaca adalah keadaan udara pada saat tertentu dan pada wilayah tertentu dalam jangka waktu singkat [1]. Prediksi curah hujan yang akurat sangat penting untuk mendukung berbagai aspek perencanaan kota, termasuk pengelolaan sumber daya air dan mitigasi risiko bencana banjir [2]. Kondisi cuaca di Indonesia saat ini

p-ISSN: 2775-4227
e-ISSN: 2775-4219

cenderung tidak stabil, dimana daerah yang awalnya terlihat cerah dapat berubah menjadi hujan atau badai dalam waktu singkat [3].Arah hembusan angin dan kecepatan angin dengan arah yang tidak menentu juga menyebabkan terjadinya curah hujan yang tidak menentu sehingga sering disebut dengan musim peralihan yang juga meyebabkan terjadinya perubahan pola curah hujan [4].

Naskah Proses perencanaan dan pengambilan keputusan menjadi lebih efektif ketika informasi cuaca tersedia dengan lengkap, akurat, dan cepat [5]. Memperkirakan hujan sangat penting karena memungkinkan masyarakat untuk melakukan antisipasi terhadap dampak yang mungkin terjadi dikarenakan hujan [6]. Namun perubahan iklim global menyebabkan pola cuaca yang tidak menentu, dengan pergantian musim kemarau dan hujan yang sulit diprediksi [7]. Kemajuan di bidang teknologi modern telah memicu inovasi pesat dalam pendekatan estimasi curah hujan, mencakup metode *deterministik* hingga *stokastik* [8].

Pengembangan berbagai algoritma terus dilakukan, di mana pembelajarannya dipandang sebagai bidang riset *machine learning* yang sangat menjanjikan ke depannya [9]. Secara umum, *machine learning* merupakan bagian dari Kecerdasan Buatan (*Artificial Intelligence* / AI) yang mencakup berbagai teknik dan strategi untuk menciptakan sistem yang dapat meniru manusia [10]. *Generalized Linear Model* (GLM) dan *Linear Regression* adalah dua algoritma umum digunakan dalam analisis data, generalisasi fleksibel dari regresi linear yang mampu menangani variabel respon dengan distribusi kesalahan normal [11].

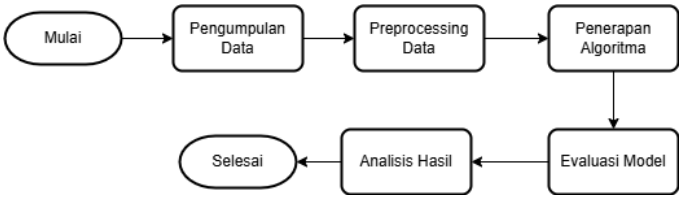
Penelitian sebelumnya menunjukkan bahwa metode berbasis data dapat meningkatkan ketepatan prediksi hujan. Misalnya, penelitian oleh [12] menunjukan bahwa algoritma *Random Forest* dengan *resampling* menggunakan SMOTE mencapai akurasi hingga 95.59% dalam prediksi hujan. Lalu penelitian lain oleh [7] beberapa metode *Machine Learning* individu telah menunjukkan performa yang cukup baik dalam prediksi cuaca, bahkan dengan penggunaan parameter *default*. Metode *Naive Bayes* mencapai akurasi tertinggi sebesar 99,00%, sementara *Logistic Regression* mencatat akurasi terendah sebesar 70,85%.

Adapun penelitian sebelumnya oleh [13] Model ANN331 terpilih sebagai model paling optimal dalam studi ini, dengan input berupa data cuaca permukaan dan struktur jaringan yang terdiri atas tiga neuron pada *hidden layer*. Model tersebut menunjukkan *error* sebesar 9,7 mm serta korelasi 0,5. Selain itu penelitian oleh [14] pada studi kasus Kota Medan, metode *Bayesian Vector Autoregressive* (BVAR) memberikan hasil estimasi yang menunjukkan curah hujan tertinggi sebesar 571,87 mm di bulan September dan yang terendah 54,59 mm di bulan Januari. Tingkat akurasi model sebesar 4,75% menandakan bahwa BVAR memiliki kinerja estimasi yang sangat baik.

Kehidupan sehari-hari tidak lepas dari peristiwa musiman yang acap kali diabaikan, namun dengan pendekatan analisis yang tepat, pola tersebut dapat dimanfaatkan untuk mendukung keputusan dan perencanaan secara lebih terarah [15]. Dalam konteks ini, GLM dapat menangani data dengan distribusi yang berbeda dan cocok untuk klasifikasi, sedangkan Linear Regression efektif untuk prediksi nilai kontinu. Dengan membandingkan keduanya, saya ingin mengetahui metode mana yang lebih optimal untuk prediksi curah hujan berdasarkan data yang tersedia. Diharapkan hasil penelitian ini dapat memberikan kontribusi signifikan dalam pengembangan sistem prediksi hujan yang lebih akurat dan efisien.

2. METODE PENELITIAN

Untuk memastikan keluaran penelitian sesuai dengan yang direncanakan, diperlukan prosedur penelitian yang jelas. Berikut tahapan yang dilakukan digambarkan dalam Gambar 1.



Gambar 1. Kerangka Penelitian

Penelitian ini diawali dengan pengumpulan dataset, dilanjutkan dengan tahap *preprocessing* untuk menyiapkan data. Setelah itu, data dibagi menjadi data latih dan data uji. Algoritma *Generalized Linear Model* dan *Linear Regression* kemudian diterapkan dalam proses klasifikasi menggunakan sejumlah fitur yang tersedia. Kinerja masing-masing algoritma dievaluasi berdasarkan nilai akurasi yang dihitung melalui program.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

2.1. Pengumpulan Dataset

Data historis cuaca ini sebanyak 2.500 diperoleh dari platform Kaggle. Data ini mencakup berbagai parameter, seperti suhu, kelembaban, kecepatan angin, tutupan awan, tekanan dan hujan. Selain itu, dilakukan pemeriksaan kelengkapan untuk memastikan data sesuai dengan lokasi, periode waktu, dan relevansi penelitian. Proses validasi awal juga dijalankan guna menjamin kualitas data dalam hal akurasi dan konsistensi.

2.2. *Preprocessing*

Prerpocessing merupakan tahapan untuk menyiapkan data mentah menjadi data yang siap olah. Tahapannya mencakup *encoding*, normalisasi dan *class imbalance*.

2.2.1. *Encoding*

Pada tahapan ini akan melakukan *encoding* yang dimana akan mengubah nilai kategorikal menjadi nilai numerik dengan menggunakan *LabelEncoder* dari modul *sklearn* [16]. Variabel kategorikal merupakan jenis data yang terdiri atas nilai berupa kategoriatau label teks, seperti "*rain*" dan "*no rain*" pada kolom *Rain* dalam dataset. Untuk dapat digunakan dalam algoritma pembelajaran mesin, yang biasanya hanya mendukung data numerik, variabel ini perlu dikonversi ke dalam format angka, misalnya "*rain*" diberi label 1 dan "*no rain*" diberi label 0.

2.2.2. Normalisasi

Normalisasi data adalah suatu teknik penting dalam *preprocessing* data yang melibatkan penyesuaian ulang nilai-nilai dalam dataset [17]. Hal ini diperlukan karena perbedaan besar dalam rentang nilai antar atribut dapat mengganggu kinerja optimal atribut dalam analisis data, normalisasi data menjadi suatu langkah krusial untuk memastikan data yang diolah lebih konsisten dan akurat [17]. Dalam penelitian ini dilakukan normalisasi menggunakan metode normalisasi *Z-Score* atau *StandarScaler* untuk mengolah data [17].

$$Z = \frac{x-\mu}{\sigma} \qquad (1)$$

Keterangan:

- Z = nilai *Z-Score* yang dihasilkan.
- X = nilai dari data yang akan dinormalisasi.
- μ = rata-rata (*mean*) dari seluruh data.
- σ = simpangan baku (*standard deviation*) dari seluruh data.

2.2.3. *Class Imbalance*

Resampling merupakan teknik dimana mencoba melakukan peneyimbangan data asli melalu proses algoritma *sampling* dengan menyesuaikan jumlah sampel dalam kelas berbeda [18]. Pada penelitian ini menggunakan metode *Random oversampling* untuk melakukan duplikasi sampel pada kelas minoritas, seperti kasus di mana jumlah sampel untuk kelas "*no rain*" jauh lebih banyak dibandingkan dengan kelas "*rain*". Kondisi tidak seimbang tersebut bisa menyebabkan bias pada model terhadap kelompok data yang jumlahnya lebih besar, sehingga menghasilkan prediksi yang kurang akurat, terutama untuk kelas minoritas.

2.3. Penerapan Algoritma

Pada tahap penerapan algoritma, dataset kemudian dipisahkan menjadi dua kelompok, yaitu *training set* dan *testing set*, dengan rasio tertentu seperti 80:20. Data pelatihan digunakan untuk membangun model, sedangkan data pengujian berfungsi untuk menilai kinerja model tersebut. Algoritma *Generalized Linear Model* (GLM) diterapkan untuk menganalisis hubungan linier antara variabel cuaca, dengan *Linear Regression* sebagai metode pembanding. Untuk memaksimalkan performa model, dilakukan penyesuaian *hyperparameter*, seperti jumlah iterasi dan pilihan fungsi aktivasi. Proses ini dilakukan secara terstruktur, mulai dari pembangunan model, pengujian validitas hasil, hingga analisis performa menggunakan data pengujian.

2.3.1. Algoritma Generalized Linear Model

Generalized Linear Model (GLM) merupakan perpanjangan dari model *Linier Regression* dengan asumsi bahwa predictor memiliki efek linier tetapi tidak mengasumsikan distribusi spesifik dari variabel respon dan digunakan ketika variabel respon anggota dari keluarga eksponensial [19]. GLM dibuat untuk mengelola data

p-ISSN: 2775-4227
e-ISSN: 2775-4219

dengan distribusi probabilitas yang tidak terbatas pada distribusi normal. *Model linier Generalized* dikembangkan oleh *John Nelder* dan *Robert Wedderburn* sebagai suatu metode untuk mengintegrasikan berbagai model statistik lainnya, seperti regresi linier, regresi logistik, dan regresi *Poisson* [11]. Untuk model statistik yg menggabungkan hubungan linear dengan distribusi proabilitas non-normal dapat dilihat pada Rumus berikut.

$$g(\mu) = \eta = X\beta \quad (2)$$

Keterangan :

- $\mu = E[Y]$: Rata-rata dari variabel respons Y
- $g(\mu)$: Menghubungkan rata – rata respons μ
- X : *Matriks predictor*
- $\eta = X\beta$: Hubungan linear
- β : Vektor parameter model yang akan diestimasi

2.3.2. Algoritma *Linear Regression*

Regresi linier merupakan metode untuk menganalisis hubungan antara dua variabel atau lebih, yang digunakan dalam prediksi data dengan menggambarkannya dalam bentuk garis lurus [20]. Hubungan variabel dependen dan varibel independen tergantung dalam beberapa bentuk persamaan, linear, eksponensial dan yang terakhir berganda [19]. *Linear Regression* memodelkan hubungan antara variabel dependen Y dan independen X, untuk model statistik yg menggabungkan hubungan linear dengan distribusi proabilitas *non-normal* dapat dilihat pada rumus berikut.

$$Y = \beta_0 + \beta_1 X + \epsilon \quad (3)$$

Keterangan :

- $\beta 0$: Intersep (kostanta) yang mewakili nilai Y saat X = 0
- $\beta 1$X : Koefisien regresi menunjukkan perubahan rata-rata Y
- ϵ : *Error* (residual), yaitu selisih antar nilai prediksi dan nilai aktual Y

2.4. Evaluasi

Evaluasi dilakukan untuk menilai kinerja model yang telah dibangun. Evaluasi model dilakukan menggunakan indikator statistik seperti *Root Mean Squared Error*(RMSE), *Mean Absolute Error*(MAE), dan koefisien determinasi (*R-squared*) untuk menilai akurasi dan validitas prediksi, yang bertujuan untuk mendukung perencanaan pertanian yang lebih baik [21]. Hasil evaluasi dan prediksi kemudian disajikan dalam bentuk grafik atau diagram untuk mempermudah analisis dan pengambilan keputusan [21].

a. *Accuracy*

$$\text{Akurasi} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (4)$$

Keterangan:

- yi : Nilai yang sebenarnya.
- $\hat{yi}$: Nilai yang diprediksi.
- $\bar{y}$: Rata-rata dari nilai aktual.
- n : Total Jumlah data

b. *RMSE*

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (5)$$

Keterangan:

- yi : Nilai yang sebenarnya.
- Y' : Nilai yang prediksi.
- n : Total Jumlah data.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

c. *MAE*

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad (6)$$

Keterangan:

- n : Jumlah data
- fi : Nilai aktual pada data ke- i
- f^i : Nilai prediksi pada data ke-i

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pengumpulan Dataset

Pada proses penelitian ini menggunakan data *Weather Forecast Data* yang berasal dari platform Kaggle dengan format data csv, dan memiliki 6 fitur dengan jumlah total 2500 data yang telah diklasifikasikan berdasarkan kriteria tertentu serta dilengkapi dengan label keterangan hujan. Untuk penjelasan lebih lanjut mengenai dataset *Weather Forecast Data* dapat dilihat pada gambar 2.

	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
0	23.720338	89.592641	7.335604	50.501694	1032.378759	rain
1	27.879734	46.489704	5.952484	4.990053	992.614190	no rain
2	25.069084	83.072843	1.371992	14.855784	1007.231620	no rain
3	23.622080	74.367758	7.050551	67.255282	982.632013	rain
4	20.591370	96.858822	4.643921	47.676444	980.825142	no rain
...	...	...	...	...	...	...
2495	21.791602	45.270902	11.807192	55.044682	1017.686181	no rain
2496	27.558479	46.481744	10.884915	39.715133	1008.590961	no rain
2497	28.108274	43.817178	2.897128	75.842952	999.119187	no rain
2498	14.789275	57.908105	2.374717	2.378743	1046.501875	no rain
2499	26.554356	97.101517	18.563084	81.357508	1001.729176	no rain

2500 rows × 6 columns

Gambar 2. Dataset Prediksi Hujan

3.2. *Preprocessing*

Tahap *preprocessing* dilakukan untuk menyiapkan data agar dapat digunakan secara optimal dalam proses klasifikasi. Proses ini meliputi beberapa langkah, di antaranya yaitu *encoding*, normalisasi dan *class imbalance*.

3.2.1. *Encoding*

Encoding digunakan untuk mengubah label hujan yang semula berupa data kategorikal (*rain* dan *no rain*) menjadi bentuk numerik, yaitu 1 untuk *rain* dan 0 untuk *no rain*, agar dapat dikenali oleh algoritma klasifikasi. Dataset setelah proses *encoding* dapat dilihat pada gambar 3.

	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
0	23.720338	89.592641	7.335604	50.501694	1032.378759	1
1	27.879734	46.489704	5.952484	4.990053	992.614190	0
2	25.069084	83.072843	1.371992	14.855784	1007.231620	0
3	23.622080	74.367758	7.050551	67.255282	982.632013	1
4	20.591370	96.858822	4.643921	47.676444	980.825142	0
...	...	...	...	...	...	...
2495	21.791602	45.270902	11.807192	55.044682	1017.686181	0
2496	27.558479	46.481744	10.884915	39.715133	1008.590961	0
2497	28.108274	43.817178	2.897128	75.842952	999.119187	0
2498	14.789275	57.908105	2.374717	2.378743	1046.501875	0
2499	26.554356	97.101517	18.563084	81.357508	1001.729176	0

2500 rows × 6 columns

Gambar 3. Dataset Setelah *Proses Encoding*

3.2.2. Normalisasi

Normalisasi dilakukan dengan memanfaatkan teknik *StandardScaler* untuk menyeragamkan skala pada masing-masing fitur. Proses ini dilakukan dengan mengkonversi setiap nilai dalam suatu fitur menjadi skor standar (*z-score*), yaitu dengan mengurangkan nilai tersebut dari rata-rata fitur, lalu membaginya dengan deviasi standar dari fitur yang sama. Transformasi ini menghasilkan distribusi data dengan nilai tengah nol dan deviasi standar satu, sehingga skala antar fitur menjadi konsisten dan setara. Langkah ini krusial mengingat perbedaan rentang

p-ISSN: 2775-4227
e-ISSN: 2775-4219

nilai pada setiap fitur, sehingga normalisasi memungkinkan seluruh fitur berada dalam skala yang konsisten dan berpotensi meningkatkan performa model klasifikasi. Dengan melakukan normalisasi, model dapat menghindari dominasi fitur tertentu yang memiliki nilai numerik lebih besar, sehingga meningkatkan stabilitas proses pelatihan dan akurasi hasil prediksi. Untuk hasil data yang telah di normalisasi dapat dilihat pada gambar 4.

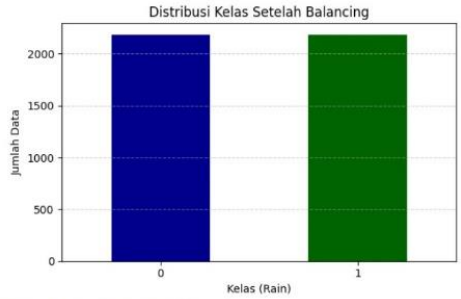
```
Data Train setelah normalisasi:
   Temperature  Humidity  Wind_Speed  Cloud_Cover  Pressure
0    -0.301358  1.079458   -0.159029     0.108881  0.073228
1     0.071158 -1.911866    1.234465    -1.131059  0.615442
2    -0.708144  1.247489    1.655399     0.725863 -0.999108
3     0.207487 -1.849584   -1.544831    -2.200392  1.654451
4    -1.513431  0.562584   -0.888248     0.162467 -1.323046

Data Test setelah normalisasi:
   Temperature  Humidity  Wind_Speed  Cloud_Cover  Pressure
0     0.980197 -0.242633    1.472548     0.778576  1.675564
1    -0.736134 -1.729549    0.274667    -0.228054 -1.468094
2     0.801258 -0.753453   -0.333486    -1.271980 -1.374157
3    -1.126936 -1.857377   -1.697757    -2.143366 -1.353019
4     0.626021  0.246403   -1.668958    -0.232760 -1.400123
```

Gambar 4. Dataset Setelah Proses Normalisasi

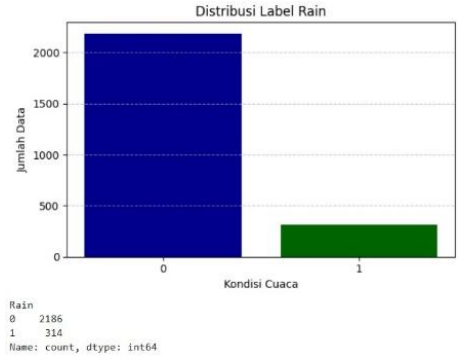
3.2.3. *Class Imbalance*

Pada saat penghitungan masing masing kategori pada kolom *Rain* dataset awal memiliki ketidakseimbangan kelas antara label "*rain*" dengan jumlah 2186 dan "*no rain*" berjumlah 314 , yang ini nanti akan dapat memengaruhi kinerja model. Untuk ketidakseimbangan data dapat dilihat pada gambar 5.



Gambar 5. Dataset Sebelum Proses Class Imbalance

Untuk mengatasi hal ini, dilakukan proses *class imbalance*, menggunakan teknik *Random Oversampling*, yaitu dengan memperbanyak data dari kelas minoritas (kelas "*rain*") secara acak hingga jumlahnya setara dengan kelas mayoritas. Langkah ini dilakukan dengan cara mengambil indeks data dari kelas minoritas secara acak menggunakan fungsi *np.random.choice*, kemudian menambahkannya ke dalam dataset. Setelah proses ini selesai, data dari kedua kelas digabung kembali sehingga distribusinya menjadi seimbang. Akibatnya, jumlah data meningkat dari 2500 menjadi 4372. Untuk hasil proses setelah dilakukan *class imbalance* dapat dilihat pada gambar 6.

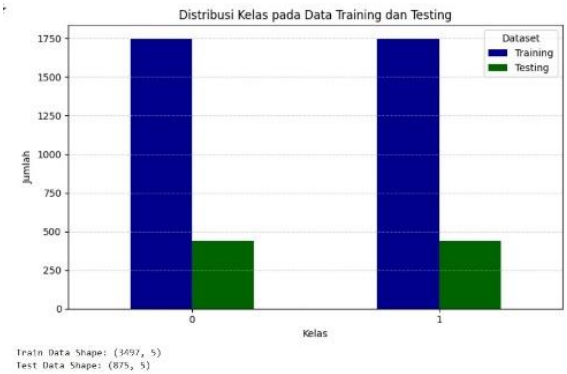


Gambar 6. Dataset Setelah Proses Class Imbalance

p-ISSN: 2775-4227
e-ISSN: 2775-4219

3.3. Penerapan Algoritma

Pada proses penerapan algoritma, Dataset dibagi menjadi dua kelompok, yaitu data untuk pelatihan dan data untuk pengujian, dengan rasio tertentu seperti 80:20 pada penelitian ini data latih sebesar 3497 dan data uji sebesar 875. Sebagian besar data, yaitu 80%, digunakan untuk melatih model klasifikasi, sementara 20% sisanya digunakan untuk menguji akurasi dan performa model. Hasil proses *splitting data* dapat dilihat pada gambar 7.



Gambar 7. Dataset Setelah Proses Splitting Data

Data pelatihan dan data pengujian yang ditampilkan merupakan *output* dari tahap *preprocessing* pada dataset cuaca, di mana lima fitur utama yaitu *temperature*, *humidity*, *wind speed*, *cloud cover*, dan *pressure* telah melalui proses normalisasi agar memiliki skala nilai yang konsisten. Tujuan dari normalisasi ini adalah untuk membantu model dalam mengenali pola dengan lebih efektif. Data pelatihan dimanfaatkan untuk membentuk model, sementara data pengujian digunakan untuk mengukur akurasi prediksi yang dihasilkan. Lima contoh baris pertama dari tabel 1 dan tabel 2 masing-masing data menunjukkan bahwa nilai-nilai fitur sudah berada dalam kisaran yang sesuai setelah proses normalisasi dilakukan.

Tabel 1. Fitur Data Latih Setelah Preprocessing

No	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure
0	-1.392491	0.124036	1.474321	1.443809	1.135657
1	-0.101871	-1.173502	-0.967838	0.516164	-0.639219
2	-0.205768	0.749073	0.174455	0.669427	-0.357446
3	-1.424449	1.386872	0.715992	-1.197624	-1.328893
4	-0.277612	0.066846	-1.140207	0.054985	1.193317
3496	-0.547792	-1.909650	1.298072	1.472304	-0.759359

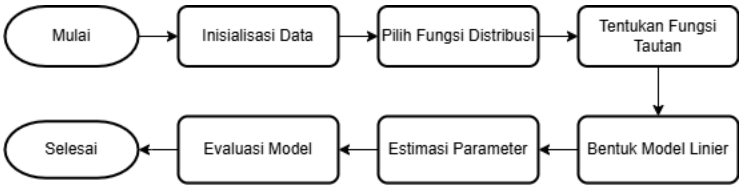
Tabel 2. Fitur Data Uji Setelah Preprocessing

No	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure
0	-0.607676	0.905009	-0.762483	0.549580	-0.112550
1	-0.950115	-1.998641	0.468100	0.853258	-1.069926
2	1.258155	-2.008984	1.628483	-0.274802	-0.345098
3	1.750851	-1.092264	1.591385	0.503243	-1.056096
4	-1.357345	0.158629	-1.525656	-0.287364	-0.557872
874	-0.135285	1.087277	-1.395732	-0.133810	-1.348407

3.3.1. Algoritma *Generalized Linear Model*

Setelah proses pembagian data, dilakukan pengujian terhadap algoritma *Generalized Linear Model*. Implementasi algoritma *Generalized Linear Model* (GLM) dalam penelitian ini dilakukan dengan pendekatan yang terstruktur, mengikuti urutan langkah-langkah yang dijelaskan dalam *flowchart* pada Gambar 8. *Flowchart* tersebut menggambarkan langkah-langkah utama dalam penerapan GLM, mulai dari proses awal (inisialisasi data) hingga tahap akhir (evaluasi model).

p-ISSN: 2775-4227
e-ISSN: 2775-4219



Gambar 8. Algoritma Generalized Linear Model

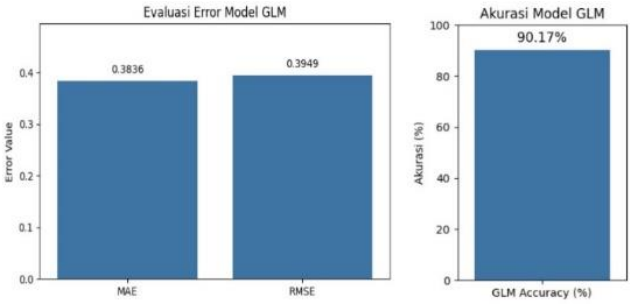
Proses implementasi dimulai dengan langkah "Mulai", diikuti oleh tahap Inisialisasi Data, yang melibatkan pemuatan dataset *weather_forecast_data.csv* yang berisi informasi prakiraan cuaca. Dataset ini mencakup fitur-fitur seperti suhu, kelembaban, tekanan udara, tutupan awan, dan kecepatan angin, dengan variabel target berupa curah hujan.

Setelah data dilakukan tahap *preprocessing*, yang meliputi normalisasi fitur numerik dengan *scaler* serta pembagian dataset menjadi dua bagian yaitu data latih (*X_train_scaled, y_train*) dan data uji (*X_test_scaled*, *y_test*). Dalam kode implementasi, tahap ini tercermin pada penggunaan variabel *X_input = X_train_scaled* dan *y_input = y_train*, yang sesuai dengan blok "Inisialisasi Data" dalam *flowchart*. Selanjutnya, pada tahap pemilihan fungsi distribusi, dipilih distribusi *Tweedie* dengan parameter *power*=1, yang setara dengan distribusi *Poisson*. Distribusi *Poisson* sangat tepat untuk memodelkan data berupa angka hitung, contohnya jumlah curah hujan dalam interval waktu tertentu. Pada tahap berikutnya, yaitu tentukan fungsi tautan, digunakan fungsi tautan *log*, yang mengaitkan rata-rata nilai respons dengan kombinasi linier variabel-variabel input. Fungsi *log* ini menjamin bahwa hasil prediksi model selalu bernilai positif, yang merupakan hal krusial untuk data seperti curah hujan yang tidak boleh memiliki nilai negatif. Implementasi ini dilakukan dengan memanggil model *TweedieRegressor(power=1, link='log')*.

Pada tahap selanjutnya, dibentuk model linier GLM, diikuti dengan proses estimasi parameter, yang dilakukan dengan melatih model menggunakan data latih melalui *glm_model.fit(X_input, y_input)*. Proses ini menentukan nilai optimal dari vektor parameter β, sehingga model dapat mewakili hubungan terbaik antara variabel prediktor (X) dan respons (Y).

Setelah model terlatih, dilakukan Prediksi Nilai pada data uji menggunakan fungsi *glm_model.predict(X_test_scaled)*. Hasil prediksi berupa nilai kontinu kemudian diklasifikasikan menjadi dua kategori (hujan atau tidak hujan) dengan menggunakan *threshold* 0.5, melalui *glm_predictions_class = (glm_predictions >= threshold).astype(int)*.

Setelah proses pembentukan model dan estimasi parameter selesai, tahap selanjutnya adalah menguji performa model menggunakan data uji. Akurasi model dihitung dengan menggunakan *accuracy_score*, *Root Mean Square Error* (RMSE), dan *Mean Absolute Error* (MAE). Seluruh proses ini ditutup dengan penyimpulan, menandai selesainya penerapan algoritma GLM dalam prediksi curah hujan. Pada proses pengujian klasifikasi *Generalized Linear Model*, hasil evaluasi Model GLM memiliki nilai akurasi sebesar 90.17%, MAE sebesar 0.3836 dan RMSE sebesar 0.3949. Grafik dapat dilihat pada gambar 9.



Gambar 9. Hasil Evaluasi Generalized Linear Model

Tabel ini memperlihatkan hasil pemodelan *Generalized Linear Model* (GLM) pada data cuaca dan curah hujan. Setiap baris berisi pengamatan dengan variabel input seperti suhu, kelembapan, kecepatan angin, tutupan awan, dan tekanan udara yang sudah dinormalisasi. Kolom *Actual_Rainfall* menunjukkan data hujan asli (1 untuk hujan, 0 untuk tidak), sementara *Predicted_Rainfall* berisi probabilitas hujan yang diprediksi model. *Predicted_Class* dan *Actual_Class* mengkategorikan hasil prediksi dan data asli secara biner. Sepuluh baris pertama memberikan gambaran bagaimana model memprediksi hujan berdasarkan kondisi cuaca. Dataset ini memiliki total 875 baris, menyediakan data yang cukup untuk evaluasi model. Yang dapat dilihat pada tabel 3.

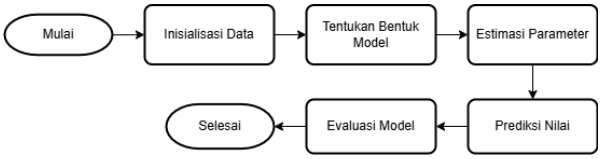
p-ISSN: 2775-4227
e-ISSN: 2775-4219

Tabel 3. Hasil Prediksi dengan Proses penerapan GLM

No	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Actual_Rainfall	Predicted_Rainfall	Predicted_Class	Actual_Class
0	-0.607676	0.905009	-0.762483	0.549580	-0.112550	1	0.670706	1	1
1	-0.950115	-1.998641	0.468100	0.853258	-1.069926	0	0.417741	0	0
2	1.258155	-2.008984	1.628483	-0.274802	-0.345098	0	0.259585	0	0
3	1.750851	-1.092264	1.591385	0.503243	-1.056096	0	0.329021	0	0
4	-1.357345	0.158629	-1.525656	-0.287364	0.557872	1	0.562299	1	1
874	-0.135285	1.087277	-1.395732	-0.133810	1.348407	1	0.587289	1	1

3.3.2. Algoritma *Linear Regression*

Implementasi algoritma *Linear Regression* dalam penelitian ini dilakukan dengan langkah-langkah terstruktur sesuai dengan urutan yang ditunjukkan dalam flowchart pada Gambar 10. Flowchart tersebut menggambarkan tahapan utama dalam penerapan regresi linier, dimulai dari tahap awal (inisialisasi data) hingga tahap akhir (evaluasi model). Proses implementasi dimulai dengan tahap "Mulai", diikuti oleh Inisialisasi Data, yang meliputi pemuatan dataset *weather_forecast_data.csv* yang berisi informasi tentang prakiraan cuaca. Dataset ini mencakup fitur-fitur seperti suhu, kelembaban, tekanan udara, tutupan awan, kecepatan angin, serta curah hujan yang menjadi target prediksi.



Gambar 10. Algoritma Regression Linear

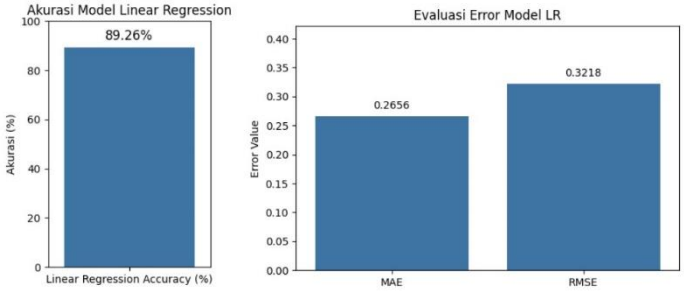
Setelah data dilakukan tahap *preprocessing*, yang mencakup normalisasi fitur numerik dan pembagian dataset menjadi dua bagian yaitu data latih *(X_train_scaled, y_train)* dan data uji *(X_test_scaled, y_test)*. Pada implementasi kode, hal ini dilakukan dengan menetapkan variabel *X_input = X_train_scaled* dan *y_input = y_train*. Tahap ini sejalan dengan bagian "Inisialisasi Data" dalam *flowchart*, yang menunjukkan bahwa data perlu disiapkan terlebih dahulu sebelum proses pemodelan. Selanjutnya, pada tahap Penentuan Bentuk Model, dipilih model regresi linier dari *library sklearn.linear_model*, yaitu dengan memanggil *lr_model = LinearRegression()*. Model ini dipilih karena kemampuannya dalam mengidentifikasi hubungan linier antara fitur cuaca dan target (curah hujan). Ini sesuai dengan tahap "Tentukan Bentuk Model" dalam *flowchart*, yang merupakan dasar untuk langkah pelatihan selanjutnya.

Tahap berikutnya adalah Estimasi Parameter, di mana model dilatih menggunakan data latih melalui *lr_model.fit(X_input, y_input)*. Proses ini bertujuan untuk menghitung koefisien regresi dan intersep. Setelah parameter diperoleh, model digunakan untuk melakukan Prediksi Nilai pada data uji dengan fungsi *lr_model.predict(X_test_scaled)*. Hasil prediksi ini kemudian diklasifikasikan menjadi dua kelas berdasarkan *threshold* 0.5 *(lr_predictions_class = (lr_predictions >= threshold). astype(int)),* yang menunjukkan apakah curah hujan terjadi atau tidak. Tahap ini sesuai dengan kotak "Prediksi Nilai" dalam *flowchart* yang menggambarkan penggunaan model untuk menghasilkan *output*.

Selanjutnya, pada tahap Evaluasi Model, hasil prediksi dibandingkan dengan nilai aktual menggunakan metrik evaluasi seperti akurasi (*accuracy_score*), *Mean Absolute Error* (MAE), dan *Root Mean Square Error* (RMSE). Selain perhitungan numerik, evaluasi juga divisualisasikan dengan menggunakan *bar chart*, yang memperlihatkan perbandingan antara data aktual dan hasil prediksi, serta kesalahan model. Ini mendukung bagian "Evaluasi Model" dalam *flowchart*, yang merupakan komponen penting untuk menilai kinerja model sebelum dianggap final.

Pada proses pengujian klasifikasi Regression Linear, hasil evaluasi Model memiliki nilai akurasi sebesar 89.26%, MAE sebesar 0.2656 dan RMSE sebesar 0.3218. Grafik dapat dilihat pada gambar 11.

p-ISSN: 2775-4227
e-ISSN: 2775-4219



Gambar 11. Hasil Evaluasi Regression Linear

Tabel ini menunjukkan hasil pemodelan *Linear Regression* pada data cuaca dan curah hujan. Setiap baris merepresentasikan pengamatan dengan variabel input seperti suhu, kelembapan, kecepatan angin, tutupan awan, dan tekanan udara yang telah dinormalisasi. Kolom *Actual_Rainfall* berisi data hujan asli (1 untuk hujan, 0 untuk tidak), sedangkan *Predicted_Rainfall* adalah prediksi model dalam bentuk angka kontinu. *Predicted_Class* dan *Actual_Class* mengelompokkan hasil prediksi dan data asli secara biner. Contoh 10 baris pertama menggambarkan cara model memperkirakan curah hujan berdasarkan kondisi cuaca. Dataset terdiri dari 875 baris, cukup luas untuk evaluasi performa model. Yang dapat dilihat pada tabel 3.

Tabel 4. Hasil Prediksi dengan Proses penerapan Regresi Linear

No	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Actual_Rainfall	Predicted_Rainfall	Predicted_Class	Actual_Class
0	-0.607676	0.905009	-0.762483	0.549580	-0.112550	1	0.906121	1	1
1	-0.950115	-1.998641	0.468100	0.853258	-1.069926	0	0.320385	0	0
2	1.258155	-2.008984	1.628483	-0.274802	-0.345098	0	-0.211790	0	0
3	1.750851	-1.092264	1.591385	0.503243	-1.056096	0	0.079253	0	0
4	-1.357345	0.158629	-1.525656	-0.287364	-0.557872	1	0.690140	1	1
874	-0.135285	1.087277	-1.395732	-0.133810	-1.348407	1	0.765184	1	1

3.4. Evaluasi

Berdasarkan hasil pengujian, algoritma *Linear Regression* menghasilkan nilai *Mean Absolute Error* (MAE) memiliki nilai 0.2656, dan *Root Mean Squared Error* (RMSE) memiliki nilai 0.3218, dengan tingkat akurasi klasifikasi sebesar 89.26%. Sementara itu, algoritma *Generalized Linear Model* (GLM) menunjukkan performa yang sedikit berbeda, dengan nilai MAE sebesar 0.3836, dan RMSE sebesar 0.3949, serta tingkat akurasi klasifikasi sebesar 90.17%.

Tabel 5. Hasil Evaluasi

Metrik Evaluasi	Linear Regression	Generalized Linear Model
Mean Absolute Error (MAE)	0.2656	0.3836
Root Mean Squared Error (RMSE)	0.3218	0.3949
Akurasi (%)	89.26%	90.17%

Perbedaan hasil akurasi, MAE, dan RMSE antara GLM dan *Linear Regression* disebabkan oleh karakteristik masing-masing model. GLM mampu menangani hubungan *non-linier* dan distribusi data yang tidak normal, sehingga menghasilkan akurasi klasifikasi yang lebih tinggi. Sedangkan Linear Regression mengasumsikan hubungan linier sederhana, sehingga lebih efektif dalam meminimalkan kesalahan prediksi seperti MAE dan RMSE. Secara keseluruhan, model GLM dan *Linear Regression* pada penelitian ini memberikan hasil yang baik dengan pendekatan yang sederhana dan mudah diinterpretasi, sedangkan metode *Random Forest* pada penelitian [12] berpotensi memberikan performa lebih baik pada data yang kompleks dan *non-linear*. Oleh karena itu,

p-ISSN: 2775-4227
e-ISSN: 2775-4219

pemilihan model sebaiknya disesuaikan dengan kebutuhan, apakah mengutamakan interpretabilitas dan akurasi klasifikasi GLM, minimisasi kesalahan prediksi *Linear Regression*, atau kemampuan menangani kompleksitas data *Random Forest*.

4. KESIMPULAN

Perkiraan curah hujan memainkan peran penting dalam berbagai kegiatan manusia, terutama di wilayah beriklim tropis yang memiliki pola cuaca yang dinamis. Guna meningkatkan keakuratan prediksi tersebut, studi ini membandingkan performa dua algoritma *machine learning*, yakni *Generalized Linear Model* dan *Linear Regression*. Data ini menggunakan data *Weather Forecast Data.csv* yang berasal dari platform Kaggle, kemudian melalui tahapan *preprocessing* seperti label *encoding*, normalisasi, serta *class imbalance*. Dataset selanjutnya dipisah menjadi data pelatihan dan pengujian sebelum diterapkan ke masing-masing algoritma. *Generalized Linear Model* merupakan model regresi yang dirancang untuk menangani distribusi non-normal pada variabel target dan menghubungkannya dengan variabel prediktor melalui fungsi tautan. Sementara itu, *Linear Regression* merupakan pendekatan sederhana yang menggambarkan hubungan linier antara variabel bebas dan terikat. Hasilnya, *Generalized Linear Model* memiliki nilai akurasi sebesar 90.17% lalu untuk RMSE sebesar 0.3949 dan MAE 0.3836, sedangkan *Linear Regression* memiliki nilai MAE sebesar 0.2656 dan RMSE 0.3218 lalu untuk akurasi sebesar 89.26%. Dengan demikian, hasil penelitian ini tidak hanya memberikan gambaran efektivitas kedua model dalam prediksi curah hujan, tetapi juga membuka peluang pengembangan model yang lebih kompleks dan adaptif di masa depan, sehingga dapat mendukung pengambilan keputusan yang lebih tepat dalam mitigasi risiko bencana atau kebutuhan lainnya.

DAFTAR PUSTAKA

[1] D. A. H. Panggabean, F. M. Sihombing, and N. M. Aruan, "Prediksi Tinggi Curah Hujan Dan Kecepatan Angin Berdasarkan Data Cuaca Dengan Penerapan Algoritma Artificial Neural Network (Ann)," *PROSIDING SEMINASTIKA*, vol. 3, no. 1, pp. 1–7, 2021.

[2] I. Farisi, J. Shadiq, W. Priyadi, D. Maulana, A. Acep, and S. F. Gusril, "Penerapan Model Recurrent Neural Network (RNN) untuk Prediksi Curah Hujan Berbasis Data Historis," *INFORMATION SYSTEM FOR EDUCATORS AND PROFESSIONALS: Journal of Information System*, vol. 9, no. 2, pp. 217–226, 2024.

[3] R. Tanjung, A. Listiani, and F. Lestari, "Prediksi Multivariate Time Series Parameter Cuaca Menggunakan Long Short-Term Memory (LSTM)," in *PROSIDING SEMINAR NASIONAL SAINS DATA*, 2024, pp. 445–456.

[4] D. Safitri, S. S. Hilabi, and F. Nurapriani, "Analisis Penggunaan Algoritma Klasifikasi Dalam Prediksi Kelulusan Menggunakan Orange Data Mining," *RABIT: Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 8, no. 1, pp. 75–81, 2023.

[5] T. Rohana, E. Nurlaelasari, E. E. Awal, and H. Y. Novita, "Kajian Model Jaringan Syaraf Tiruan Untuk Memprediksi Secara Dini Tingkat Kelulusan Mahasiswa," *Technologia: Jurnal Ilmiah*, vol. 15, no. 4, pp. 629–640, 2024.

[6] N. Nursobah, S. Lailiyah, B. Harpad, and M. Fahmi, "Penerapan Data Mining Untuk Prediksi Perkiraan Hujan dengan Menggunakan Algoritma K-Nearest Neighbor," *Building Of Informatics, Technology And Science (Bits)*, vol. 4, no. 3, pp. 1395–1400, 2022.

[7] S. Joses, D. Yulvida, and S. Rochimah, "Pendekatan Metode Ensemble Learning untuk Prakiraan Cuaca menggunakan Soft Voting Classifier," *Journal of Applied Computer Science and Technology*, vol. 5, no. 1, pp. 72–80, 2024.

[8] B. Susilo, *Mengenal Iklim dan Cuaca di Indonesia*. Diva Press, 2021.

[9] S. A. Pratiwi, A. Fauzi, S. A. P. Lestari, and Y. Cahyana, "Prediksi Persediaan Obat Pada Apotek Menggunakan Algoritma Decision Tree," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 4, pp. 2381–2388, 2024.

[10] D. Kurniawan, *Pengenalan machine learning dengan python*. Elex Media Komputindo, 2022.

[11] A. M. Siregar, "Klasifikasi Untuk Prediksi Cuaca Menggunakan Esemble Learning," *Petir*, vol. 13, no. 2, p. 522607, 2020.

[12] G. A. Mursianto, I. M. Falih, M. Irfan, T. Sakinah, and D. S. Prasvita, "Perbandingan Metode Klasifikasi Random Forest dan XGBoost Serta Implementasi Teknik SMOTE pada Kasus Prediksi Hujan," *J. Senamika*, vol. 2, no. 2, pp. 41–50, 2021.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

[13] R. M. Putra and N. A. Rani, "Prediksi Curah Hujan Harian di Stasiun Meteorologi Kemayoran Menggunakan Artificial Neural Network (ANN)," *Buletin GAW Bariri (BGB)*, vol. 1, no. 2, pp. 101–108, 2020.

[14] W. Y. Ambarita, S. Dur, and S. Harleni, "ANALISIS DIAGNOSTIK VARIABEL CUACA UNTUK ESTIMASI POLA CURAH HUJAN DI MEDAN MENGGUNAKAN MODEL BAYESIAN VECTOR AUTOREGRESSIVE," *Jurnal Lebesgue: Jurnal Ilmiah Pendidikan Matematika, Matematika dan Statistika*, vol. 4, no. 3, pp. 1688–1701, 2023.

[15] A. R. Juwita, T. Al Mudzakir, A. R. Pratama, B. Nugraha, and N. Heryana, "Penerapan Algoritma Apriori untuk Memprediksi Pembayaran UKT," *Syntax: Jurnal Informatika*, vol. 13, no. 01, pp. 35–43, 2024.

[16] A. F. Istianto, A. I. Hadiana, and F. R. Umbara, "Prediksi curah hujan menggunakan metode categorical boosting (Catboost)," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 4, pp. 2930–2937, 2023.

[17] A. Suarisman, A. Nazir, F. Syafria, and L. Afriyanti, "PERBANDNGAN JARAK METRIK PADA KLASIFIKASI JAMUR BERACUN MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR (K-NN)," *PERBANDNGAN JARAK METRIK PADA KLASIFIKASI JAMUR BERACUN MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR (K-NN)*, vol. 5, no. 1, pp. 10–19, 2023.

[18] A. R. I. Pratama, S. A. Latipah, and B. N. Sari, "Optimasi klasifikasi curah hujan menggunakan support vector machine (svm) dan recursive feature elimination (rfe)," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 2, pp. 314–324, 2022.

[19] F. H. Hamdanah and D. Fitrianah, "Analisis Performansi Algoritma Linear Regression dengan Generalized Linear Model untuk Prediksi Penjualan pada Usaha Mikra, Kecil, dan Menengah," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 10, no. 1, pp. 23–32, 2021.

[20] D. S. Rahayu, J. Afifah, and S. Intan, "Classification of Diabetes Mellitus Using C4. 5 Algorithm, Support Vector Machine (SVM) and Linear Regression Klasifikasi Penyakit Diabetes Melitus Menggunakan Algoritma C4. 5, Support Vector Machine (SVM) dan Regresi Linear," *SENTIMAS Semin. Nas. Penelit. dan Pengabdi. Masy*, vol. 1, no. 1, pp. 56–63, 2023.

[21] L. Fatimah, M. Martanto, A. R. Dikananda, and A. Rifa'i, "ALGORITMA REGRESI LINEAR UNTUK PREDIKSI HASIL PANEN DAN STRATEGI PRODUKSI PADI DI KABUPATEN CIREBON," *Jurnal Informatika Teknologi dan Sains (Jinteks)*, vol. 7, no. 2, pp. 464–472, 2025.