

## **Pengembangan Sistem Prediksi Waktu Penyiraman Optimal pada Perkebunan: Pendekatan Machine Learning untuk Peningkatan Produktivitas Pertanian**

*Development of Optimal Watering Time Prediction System in Plantation: A Machine Learning Approach for Improved Agricultural Productivity*

**Mohammad F. Anggarda<sup>1,\*</sup>, Iwan Kustiawan<sup>2</sup>, Deasy R. Nurjannah<sup>3</sup>, Nurul F. A. Hakim<sup>1</sup>**

<sup>1</sup>Program Studi Pendidikan Teknik Otomasi Industri dan Robotika, Fakultas Pendidikan Teknologi dan Kejuruan, Universitas Pendidikan Indonesia, Jl. Dr. Setiabudi No.229, Isola, Kec. Sukasari, Kota Bandung, Jawa Barat 40154 Indonesia

<sup>2</sup> Program Studi Teknik Elektro, Fakultas Pendidikan Teknologi dan Kejuruan, Universitas Pendidikan Indonesia, Jl. Dr. Setiabudi No.229, Isola, Kec. Sukasari, Kota Bandung, Jawa Barat 40154 Indonesia

<sup>3</sup> Program Studi Teknik Elektronika Politeknik TEDC, Jl. Politeknik Jl. Pesantren No. 2, Cibabat, Kec. Cimahi Utara, Kota Cimahi, Jawa Barat 40513

\*E-mail Penulis Korespondensi: [anggarda@upi.edu](mailto:anggarda@upi.edu)

---

### **ABSTRACT**

*Modern agriculture relies heavily on technology, especially in irrigation management and crop watering. Several previous studies have applied field data-based predictive techniques to improve crop yields. This research aims to develop a prediction system for optimal watering time in plantations and agriculture using a machine learning approach. The rigorous methodology includes data capture, pre-processing, model evaluation and testing, validation, and visualization. High accuracy demonstrates the system's reliability in determining optimal watering needs to improve resource efficiency and crop yields in agriculture. The data obtained from the automatic weather station (AWS) via thingsboard is processed sequentially, starting from data retrieval in json format using postman to transformation into csv files with proper timestamp adjustment. The pre-processing stage includes data cleaning, variable selection, data integration, and generating a clean dataset. In the evaluation stage, the dataset is divided into training data and test data, with the application and comparison of logistic regression, random forest and decision tree models applied as classifiers. Furthermore, the validation and results stage includes prediction, performance testing using the confusion matrix, and visualization of prediction results in the form of text and icons that aim to increase interpretability for users through Google Collaboratory. The results of this research provide an overview of the optimal watering time based on the dataset from the automatic weather station. Further analysis shows that the implementation of machine learning models significantly improves the prediction accuracy, proving the effectiveness of the system in providing more precise watering time recommendations to increase agricultural productivity. The main objective is to develop a machine learning-based watering time prediction system using data from the automatic weather station and evaluate various classifier algorithms to select the best model.*

**Keywords:** Development of prediction system; optimal watering time; machine learning and agricultural productivity.

### **ABSTRAK**

Pertanian modern sangat bergantung pada teknologi, khususnya dalam pengelolaan irigasi dan penyiraman tanaman. Beberapa penelitian terdahulu telah menerapkan teknik prediktif berbasis data lapangan untuk meningkatkan hasil panen. Penelitian ini bertujuan untuk mengembangkan sistem prediksi waktu penyiraman optimal pada perkebunan maupun pertanian dengan menggunakan pendekatan machine learning. Metodologi yang ketat mencakup pengambilan data, pra-pemrosesan, evaluasi dan pengujian model, validasi, dan visualisasi. Akurasi yang tinggi menunjukkan keandalan sistem dalam menentukan kebutuhan penyiraman optimal untuk meningkatkan efisiensi sumber daya dan hasil panen dalam pertanian. Data yang diperoleh dari automatic weather station (AWS) melalui thingsboard diproses secara berurutan, mulai dari pengambilan data dalam format json menggunakan postman hingga transformasi menjadi file csv dengan penyesuaian timestamp yang tepat. Tahap pra-pemrosesan mencakup pembersihan data, pemilihan variabel,

integrasi data, dan menghasilkan dataset yang bersih. Pada tahap evaluasi, dataset dibagi menjadi data latih dan data uji, dengan penerapan dan perbandingan model logistic regression, random forest dan decision tree diterapkan sebagai classifier. Selanjutnya pada tahap validasi dan hasil mencakup prediksi, pengujian kinerja menggunakan matriks konfusi, serta visualisasi hasil prediksi dalam bentuk teks dan icon yang bertujuan untuk meningkatkan interpetabilitas bagi pengguna melalui google colaboratory. Hasil penelitian ini memberikan gambaran tentang waktu penyiraman optimal berdasarkan dataset dari automatic weather station. Analisis lebih lanjut menunjukkan bahwa implementasi model machine learning secara signifikan meningkatkan akurasi prediksi, membuktikan efektivitas sistem dalam memberikan rekomendasi waktu penyiraman yang lebih presisi untuk meningkatkan produktivitas pertanian. Tujuan utamanya untuk mengembangkan sistem prediksi waktu penyiraman berbasis machine learning menggunakan data dari automatic weather station serta mengevaluasi berbagai algoritma klasifikator untuk memilih model terbaik.

**Kata Kunci:** Pengembangan sistem prediksi; waktu penyiraman optimal; machine learning dan produktivitas pertanian.

---

## PENDAHULUAN

Pertanian merupakan sektor penting dalam perekonomian suatu negara. Seiring dengan pertumbuhan populasi dan peningkatan kebutuhan pangan, produktivitas pertanian menjadi salah satu fokus utama dalam upaya memenuhi kebutuhan pangan yang terus meningkat (Prema *et al.*, 2022; Pallathadka *et al.*, 2023; Veeragandham & Santhi, 2020). Salah satu faktor kunci dalam meningkatkan produktivitas pertanian adalah pengelolaan irigasi yang efisien dan optimal.

Di banyak daerah, sistem penyiraman pada perkebunan masih dilakukan secara manual dan tradisional. Petani sering kali hanya mengandalkan pengalaman dan pengetahuan mereka dalam menentukan waktu dan jumlah air yang diperlukan untuk menyiram tanaman (Sharma *et al.*, 2021; Sharma *et al.*, 2023). Pendekatan ini tidak efisien dan dapat menyebabkan pemborosan air serta penurunan produktivitas pertanian.

Dalam beberapa tahun terakhir, perkembangan teknologi dan kecerdasan buatan (*artificial intelligence*) telah membawa perubahan besar dalam berbagai sektor, termasuk pertanian. Salah satu teknologi yang telah banyak digunakan dalam pertanian adalah machine learning. Machine learning adalah salah satu metode dalam kecerdasan buatan yang memungkinkan komputer untuk belajar dari data dan melakukan prediksi atau pengambilan keputusan berdasarkan pola yang ditemukan dalam data tersebut (Ferrag *et al.*, 2022; Morota *et al.*, 2018; Priyani *et al.*, 2023).

Pada perkebunan, penggunaan machine learning dapat membantu dalam pengembangan sistem prediksi waktu penyiraman optimal (Mohan *et al.*, 2023). Dengan menggunakan data historis tentang curah hujan, suhu udara, kelembaban tanah, dan faktor-faktor lain yang mempengaruhi kebutuhan air tanaman, sistem prediksi dapat mempelajari pola dan tren yang ada (Menaha & Lavanya, 2023a). Dengan demikian, sistem ini dapat memberikan rekomendasi tentang waktu dan jumlah air yang optimal untuk penyiraman tanaman.

Pengembangan sistem prediksi waktu penyiraman optimal pada perkebunan menggunakan pendekatan machine learning memiliki beberapa keuntungan. Pertama, dengan menggunakan pendekatan ini, petani dapat menghemat waktu dan tenaga dalam menentukan waktu dan jumlah air yang diperlukan untuk penyiraman tanaman. Kedua, dengan akurasi prediksi yang tinggi, sistem ini dapat membantu petani dalam mengoptimalkan penggunaan air dan mengurangi pemborosan air. Ketiga, dengan memberikan rekomendasi yang tepat, sistem ini dapat meningkatkan produktivitas pertanian dengan memastikan tanaman mendapatkan air yang cukup pada waktu yang tepat (Indu *et al.*, 2022; Menaha & Lavanya, 2023b; Saba *et al.*, 2023).

Pertanian modern sangat bergantung pada teknologi, khususnya dalam pengelolaan irigasi dan penyiraman tanaman. Beberapa penelitian terdahulu telah menerapkan teknik prediktif berbasis data lapangan untuk meningkatkan hasil panen. Memanfaatkan *deep learning* untuk memprediksi hasil panen gandum di Tiongkok dengan akurasi tinggi (Wang *et al.*, 2020). Kayad juga menerapkan *machine learning* untuk memantau variabilitas hasil jagung menggunakan citra satelit (Kayad *et al.*, 2019). Sementara itu, (Chlingaryan *et al.*, 2018), melakukan literatur review tentang penerapan *machine learning* dalam memprediksi hasil panen dan status nitrogen. Meskipun demikian, belum banyak dilakukan kajian mendalam terkait prediksi waktu penyiraman tanaman yang optimal. Penelitian (Husdi & Lasena., 2020), tentang analisis berbasis IoT untuk prediksi iklim lahan pertanian menunjukkan prototipe yang memungkinkan pemantauan kondisi tanah dan lingkungan untuk mengendalikan sistem irigasi hanya saja prediksi waktu penyiraman secara spesifik tidak dibahas.

Meskipun memiliki potensi yang besar, pengembangan sistem prediksi waktu penyiraman optimal pada perkebunan menggunakan pendekatan machine learning juga memiliki beberapa tantangan (Araújo *et al.*, 2023; Mohinur & Azharuddin, 2022). Pertama, diperlukan data yang akurat dan lengkap tentang curah hujan, suhu udara, kelembaban tanah, dan faktor-faktor lain yang relevan. Data ini harus dikumpulkan secara teratur dan dalam jumlah yang cukup untuk melatih sistem prediksi dengan baik. Kedua, diperlukan pemahaman yang mendalam tentang hubungan antara faktor-faktor cuaca dan kebutuhan air tanaman. Hal ini memerlukan penelitian dan pengembangan yang intensif untuk memastikan sistem prediksi memiliki akurasi yang tinggi. Ketiga, sistem ini juga harus mudah digunakan dan dapat diakses oleh petani dengan tingkat literasi teknologi yang beragam (Araújo *et al.*, 2023; Mohinur & Azharuddin, 2022).

Dalam konteks ini, pengembangan sistem prediksi waktu penyiraman optimal pada perkebunan menggunakan pendekatan machine learning menjadi sebuah tantangan yang menarik. Dengan memanfaatkan teknologi dan kecerdasan buatan, sistem ini dapat menjadi alat yang efektif dalam meningkatkan produktivitas pertanian (Ang & Seng, 2021; Attri *et al.*, 2023). Dalam hal ini, penelitian dan pengembangan lebih lanjut diperlukan untuk mengoptimalkan performa sistem prediksi dan memastikan implementasinya yang sukses di lapangan. Oleh karena itu, tujuan penelitian ini adalah mengembangkan sistem prediksi waktu penyiraman optimal dengan pendekatan *machine learning*. Data cuaca dari *automatic weather station* (AWS) akan dimanfaatkan dalam melakukan klasifikasi waktu penyiraman yang tepat dapat memberikan rekomendasi yang lebih presisi untuk meningkatkan efisiensi penggunaan air dan sumber daya pertanian. Penelitian ini diharapkan memberi kontribusi penting terhadap pengembangan teknologi pertanian cerdas dan adaptif di era modern.

## METODE PENELITIAN

Pengerjaan penelitian ini bertujuan untuk mengembangkan sistem prediksi waktu penyiraman yang optimal di perkebunan. Langkah-langkah metodologis yang terstruktur telah diikuti dengan cermat dan disusun secara sistematis untuk memastikan keakuratan dan keefektifan hasil prediksi.

Data collection diawali dengan akuisisi data cuaca dari AWS yang dilengkapi sensor untuk membaca parameter seperti: *ecvalue*, *humidity*, *rainfall*, *soil moisture*, *soil temperature*, *solar radiation*, dan *temperature*. Data hasil pembacaan sensor cuaca ini kemudian ditampilkan secara real-time pada platform internet of things (IoT) bernama *thingsboard*, sebagai proses *collecting data* secara kontinyu.

*Thingsboard* berperan sebagai platform IoT yang mengumpulkan data dari sensor *automatic weather station* secara otomatis. Data mentah hasil pembacaan sensor ditampilkan pada *thingsboard* tanpa adanya pemrosesan lebih lanjut. Selanjutnya, peneliti mengambil data historis yang dibutuhkan dari *thingsboard* menggunakan aplikasi *postman*. Rentang waktu pengambilan data dimulai dari 1 Januari 2022 hingga 31 Desember 2022. Data diambil dalam format *json* karena merupakan format natif yang disimpan *thingsboard*.

Data dalam format *json* yang diperoleh dari *thingsboard* kemudian ditransformasikan menjadi format *csv* untuk memudahkan *preprocessing* dan pemodelan menggunakan *machine learning*. Format *csv* dipilih karena struktur datanya yang terorganisir dalam kolom dan baris memudahkan analisis data *timeseries*. Dengan demikian, dataset cuaca dalam format *csv* siap digunakan untuk pemrosesan dan analisis lebih lanjut menggunakan teknik *machine learning* untuk prediksi waktu penyiraman.

Data *pre-processing* diawali dengan tahap data *cleansing*, data *cleansing* merupakan tahap yang lebih luas daripada data *cleaning*. Proses ini tidak hanya melibatkan tindakan teknis untuk memperbaiki data, tetapi juga mencakup pendekatan yang lebih holistik dalam memastikan kualitas keseluruhan data. Selain itu, data *cleansing* juga melibatkan identifikasi serta penanganan masalah struktural atau konseptual yang mendasarinya, seperti proses membersihkan dataset dari data yang tidak akurat, tidak lengkap, atau tidak konsisten. Data *cleansing* melibatkan identifikasi dan perbaikan masalah-masalah seperti *missing values*, *outliers*, duplikasi data, dan inkonsistensi representasi data. Proses ini lebih luas daripada sekadar data *cleaning* karena juga memperhatikan aspek integritas dan konsistensi data dari waktu ke waktu.

Tahap selanjutnya adalah data *selection* yaitu proses pemilihan subset variabel atau fitur dari dataset yang relevan dan berkontribusi terhadap tujuan analisis. Variabel yang tidak berkorelasi kuat dengan target analisis akan disaring supaya model *machine learning* dapat berfokus pada prediktor penting dan menghasilkan akurasi yang lebih tinggi.

Kemudian tahap data *integration* merupakan proses penggabungan kembali data yang telah diproses pada tahap sebelumnya. Data *selection* menyebabkan berkurangnya variabel, oleh karena itu integrasi diperlukan untuk menggabungkannya kembali menjadi dataset utuh namun sudah terbebas dari data yang bermasalah dan hanya memuat variabel relevan.

Keseluruhan proses ini menjadi prasyarat penting untuk analisis data yang akurat dan dapat diandalkan, berdasarkan informasi yang tepat juga akan membantu dalam membuat keputusan yang lebih baik. Hasil akhir dari data *pre-processing* adalah dihasilkannya *cleaned dataset*, yaitu *dataset* bersih yang terbebas dari berbagai permasalahan data sehingga siap digunakan untuk analisis dan modeling *machine learning* lebih lanjut.

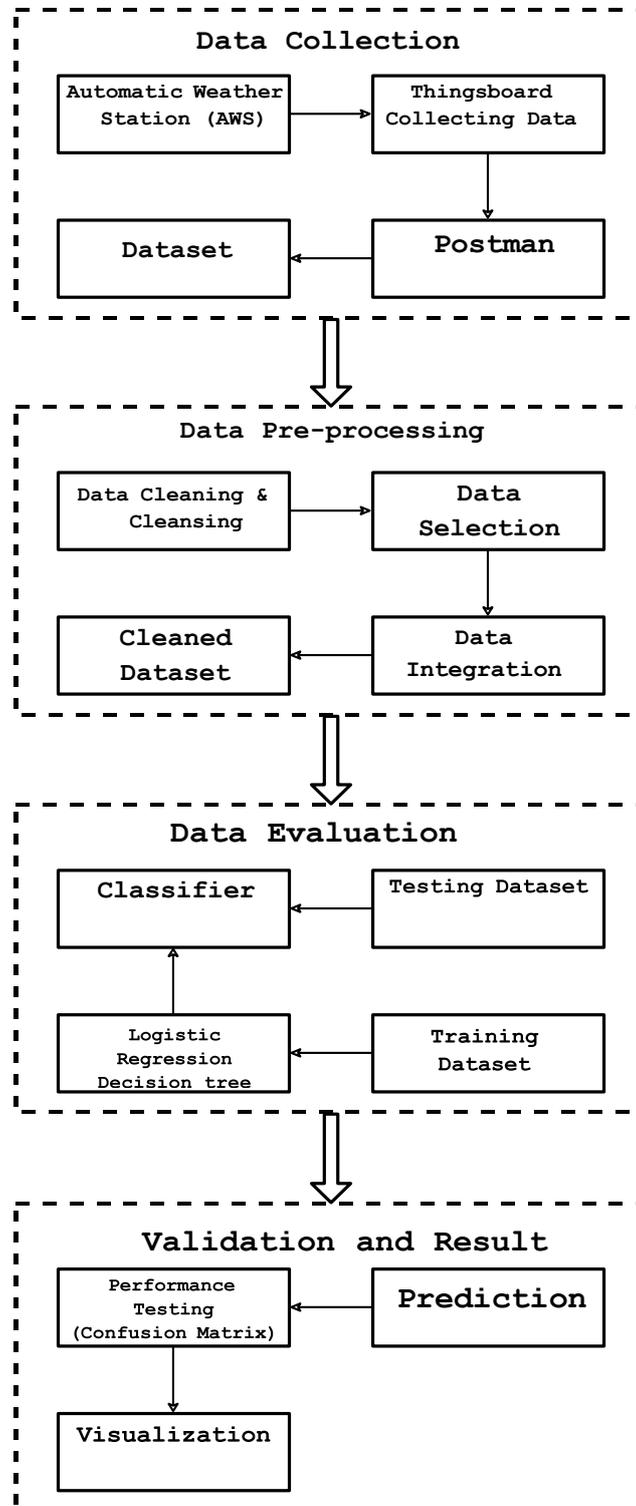
Setelah data melalui proses pra-pemrosesan, langkah selanjutnya adalah evaluasi dengan membagi dataset menjadi data latih (*training dataset*) dan data uji (*testing dataset*). Pembagian ini bertujuan untuk proses pelatihan dan pengujian beberapa model *machine learning*, yaitu *logistic regression*, *decision tree*, dan *random forest*. Ketiga model ini kemudian berperan sebagai *classifier*.

Training dataset digunakan untuk melatih *model classifier* agar dapat memetakan pola hubungan antara variabel input dengan target output. *Logistic regression* memodelkan probabilitas melalui fungsi logistik. *Decision tree* membangun sejumlah aturan *if-then* dari node dan branch. Sedangkan *random forest* membentuk banyak *decision tree* untuk kemudian menggabungkan hasil prediksinya.

Sementara itu, *testing dataset* berfungsi untuk menguji akurasi prediksi dari model yang telah dilatih sebelumnya dengan cara memberikan data input baru yang belum pernah dilihat dan memeriksa apakah output prediksinya sesuai

dengan target aktual. Pengujian terhadap testing dataset dilakukan setelah classifier selesai dilatih menggunakan training dataset.

*Classifier* sendiri merupakan algoritma pembelajaran mesin yang memetakan input data ke dalam kategori diskrit. Dalam konteks ini, *classifier* digunakan untuk mengklasifikasikan waktu penyiraman tanaman ke dalam kategori yang menunjukkan apakah waktu penyiraman tersebut optimal atau tidak. *Classifier* akan memprediksi kebutuhan penyiraman berdasarkan pola yang dipelajari dari training dataset sebelumnya.



Gambar 1. Diagram Alir Alur Kerja

Pada tahap validation and result, beberapa proses dilakukan untuk memvalidasi model classifier yang telah dibangun sebelumnya. Proses tersebut diawali dengan prediction. Prediction dilakukan dengan menerapkan beberapa model *machine learning*, yaitu *logistic regression*, *decision tree*, dan *random forest*. Ketiga model tersebut dievaluasi performanya untuk melihat model mana yang paling akurat dalam memprediksi waktu penyiraman optimal. Logistic regression adalah model klasifikasi yang cocok untuk kasus biner seperti memprediksi waktu penyiraman (perlu/tidak perlu), karena model ini dapat memetakan hubungan non-linear antara variabel input dan output (Budianto *et al.*, 2017). Selain itu, logistic regression dapat menangani korelasi antar variabel prediktor dengan lebih baik dibanding decision tree atau random forest yang cenderung memperlakukan variabel secara terpisah (Putra *et al.*, 2019). Lebih lanjut, logistic regression lebih mudah diinterpretasikan dan parameter-parameternya memiliki makna statistik, sedangkan decision tree dan random forest seringkali lebih sulit untuk dijelaskan (Listina *et al.*, 2022). Oleh karena itu hasil prediksi yang divisualisasikan berasal dari penerapan model *Logistic Regression* yang memiliki performa paling optimal.

Selanjutnya confusion matrix menampilkan accuracy, precision, recall, dan matrix penting lainnya. Hal ini memungkinkan investigasi secara rinci pada kesalahan prediksi serta contoh data mana yang paling menantang bagi classifier. Confusion matrix sangat bermanfaat dalam evaluasi kinerja dan menemukan cara untuk meningkatkan akurasi lebih lanjut.

Visualisasi hasil prediksi "Perlu Penyiraman" dan "Tidak Perlu Penyiraman" divisualisasikan menggunakan google colab. Visualisasi ini menampilkan hasil klasifikasi dari model classifier untuk masing-masing kategori perlu atau tidak perlu penyiraman. Representasi visual sangat membantu untuk memahami dan mengevaluasi kinerja prediksi dari model. Melalui visualisasi di colab ini diperoleh gambaran yang lebih intuitif mengenai seberapa akurat classifier memprediksi kebutuhan penyiraman berdasarkan pola-pola yang terdapat pada data input. Sehingga hasil prediksi waktu penyiraman optimal dapat disajikan kepada pengguna dan pihak yang membutuhkan informasi tersebut secara lebih mudah dipahami melalui visualisasi, dibandingkan harus memahami confusion matrix atau metrik hasil evaluasi model.

## HASIL DAN PEMBAHASAN

### Pengambilan dan Konversi Data Cuaca dari AWS

Pengambilan data cuaca dari AWS dimulai dengan mengonfigurasi beberapa parameter penting pada aplikasi Postman, di antaranya:

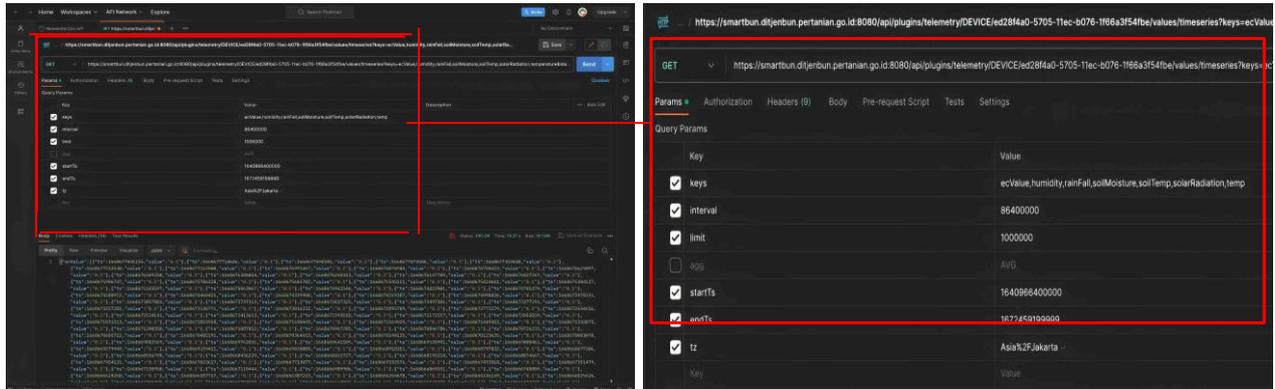
1. Keys: Merupakan daftar variabel cuaca yang akan diambil datanya. Dalam hal ini keys yang digunakan adalah:
  - a. ecValue: Electrical conductivity
  - b. humidity: Kelembapan udara
  - c. rainFall: Curah hujan
  - d. soilMoisture: Kelembapan tanah
  - e. soilTemp: Suhu tanah
  - f. solarRadiation: Radiasi matahari
  - g. temp: Temperatur udara
2. Interval: Nilai 86400000 millidetik, setara dengan periode 1 hari. Ini adalah interval waktu dalam pengambilan data.
3. Limit: Jumlah maksimum data yang akan diambil, dalam hal ini diisi 1000000 data.
4. startTs: Timestamp awal pengambilan data, diisi nilai 1640966400000 yang setara dengan 1 Januari 2022.
5. endTs: Timestamp akhir pengambilan data, diisi 1672459199999 yang setara dengan 31 Desember 2022.
6. tz: Zona waktu, diisi Asia%2FJakarta sesuai lokasi AWS berada di Indonesia bagian barat.

Pengaturan parameter tersebut mengkonfigurasi proses pengambilan data melalui postman. Data yang berhasil diambil akan berbentuk json.

Pengaturan parameter ini sangat penting untuk memastikan data yang diambil sesuai dengan kebutuhan dalam hal variabel, rentang waktu, dan zona waktu yang diinginkan. Selanjutnya data tersebut ditransformasikan ke csv agar mudah diolah dengan machine learning. Hasil transformasi data cuaca dari format json ke dalam format csv yang dilengkapi dengan penambahan timestamp ditampilkan dalam Gambar 3. Data yang ditampilkan berisi nilai dari setiap parameter yang telah diambil sebelumnya, yaitu ecValue, humidity, rainFall, soilMoisture, soilTemp, solarRadiation, dan temp. Langkah kedua ini melibatkan transformasi dari data json yang diperoleh dari postman menjadi format csv yang lebih mudah dibaca. Dalam format csv, setiap baris mewakili satu set data dan setiap kolom mewakili satu variabel. Selain itu, penambahan timestamp juga dilakukan untuk menyesuaikan data cuaca dengan waktu pengambilan yang akurat.

Tujuan dari konversi ke format csv serta penambahan timestamp ialah agar data dapat dibaca dengan mudah oleh siapapun yang memerlukan informasi cuaca tersebut. Format csv yang terstruktur dan dilengkapi dengan timestamp memudahkan analisis data secara time series, sehingga pola data cuaca dapat dipahami dengan lebih baik, dengan

demikian transformasi dari format json ke csv dengan penambahan timestamp menjadi tahapan penting dalam mempersiapkan data cuaca agar siap digunakan untuk proses pre-processing dan analisis lebih lanjut.



Gambar 2. Pengambilan Data menggunakan Postman

The image shows a Microsoft Excel spreadsheet with a table of weather data. The table has 9 columns: TimeStamp, ts, ecValue, humidity, rainFall, soilMoisture, soilTemp, solarRadiation, and temperature. The data rows show daily weather records for Saturday, 01 January 2022, with values for each parameter ranging from approximately 6.0 to 25.6.

TimeStamp	ts	ecValue	humidity	rainFall	soilMoisture	soilTemp	solarRadiation	temperature
Saturday, 01 January 2022	1.641E+12 0.0	99.0	0.0	32.0	26.0	6.0	24.1	
Saturday, 01 January 2022	1.641E+12 0.0	98.9	0.0	32.0	26.0	6.0	24.2	
Saturday, 01 January 2022	1.641E+12 0.0	98.9	0.0	32.0	26.0	7.0	24.3	
Saturday, 01 January 2022	1.641E+12 0.0	99.1	0.0	32.0	26.0	7.0	24.3	
Saturday, 01 January 2022	1.641E+12 0.0	99.1	0.0	32.0	26.0	7.0	24.4	
Saturday, 01 January 2022	1.641E+12 0.0	99.1	0.0	32.0	26.0	7.0	24.5	
Saturday, 01 January 2022	1.641E+12 0.0	99.0	0.0	32.0	26.0	7.0	24.6	
Saturday, 01 January 2022	1.641E+12 0.0	99.0	0.0	32.0	26.0	8.0	24.7	
Saturday, 01 January 2022	1.641E+12 0.0	99.0	0.0	32.0	26.0	8.0	24.8	
Saturday, 01 January 2022	1.641E+12 0.0	99.2	0.0	32.0	26.0	10.0	24.9	
Saturday, 01 January 2022	1.641E+12 0.0	99.8	0.0	32.0	26.0	9.0	25.0	
Saturday, 01 January 2022	1.641E+12 0.0	98.6	0.0	32.0	26.0	9.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	98.7	0.0	32.0	26.0	10.0	25.2	
Saturday, 01 January 2022	1.641E+12 0.0	98.5	0.0	32.0	26.0	10.0	25.3	
Saturday, 01 January 2022	1.641E+12 0.0	98.0	0.0	32.0	26.0	10.0	25.4	
Saturday, 01 January 2022	1.641E+12 0.0	97.8	0.0	32.0	26.0	10.0	25.5	
Saturday, 01 January 2022	1.641E+12 0.0	97.8	0.0	32.0	26.0	10.0	25.5	
Saturday, 01 January 2022	1.641E+12 0.0	97.8	0.0	32.0	26.0	11.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	97.4	0.0	32.0	26.0	11.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	97.1	0.0	32.0	26.0	12.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	96.8	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	96.6	0.0	32.0	26.0	14.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	96.7	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	96.0	0.0	32.0	26.0	12.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	94.8	0.0	32.0	26.0	12.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	94.7	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	94.1	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	94.0	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	93.6	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	93.5	0.0	32.0	26.0	13.0	25.7	
Saturday, 01 January 2022	1.641E+12 0.0	93.2	0.0	32.0	26.0	13.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	93.0	0.0	32.0	26.0	14.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.9	0.0	32.0	26.0	14.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.9	0.0	32.0	26.0	20.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.3	0.0	32.0	26.0	20.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.3	0.0	32.0	26.0	20.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.3	0.0	32.0	26.0	21.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	92.0	0.0	32.0	26.0	21.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	91.9	0.0	32.0	26.0	20.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	91.6	0.0	32.0	26.0	14.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	90.9	0.0	32.0	26.0	14.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	90.3	0.0	32.0	26.0	14.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	89.9	0.0	32.0	27.0	17.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	89.9	0.0	32.0	27.0	17.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	91.0	0.0	32.0	27.0	20.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	90.7	0.0	32.0	27.0	22.0	25.1	
Saturday, 01 January 2022	1.641E+12 0.0	90.8	0.0	32.0	27.0	22.0	25.2	
Saturday, 01 January 2022	1.641E+12 0.0	90.9	0.0	32.0	27.0	22.0	25.4	
Saturday, 01 January 2022	1.641E+12 0.0	90.6	0.0	32.0	27.0	23.0	25.4	
Saturday, 01 January 2022	1.641E+12 0.0	89.2	0.0	32.0	27.0	22.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	89.7	0.0	32.0	27.0	24.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	90.1	0.0	32.0	27.0	25.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	89.7	0.0	32.0	27.0	24.0	25.6	
Saturday, 01 January 2022	1.641E+12 0.0	89.6	0.0	32.0	27.0	42.0	25.7	
Saturday, 01 January 2022	1.641E+12 0.0	89.1	0.0	32.0	27.0	34.0	25.8	
Saturday, 01 January 2022	1.641E+12 0.0	90.3	0.0	32.0	27.0	38.0	25.9	
Saturday, 01 January 2022	1.641E+12 0.0	89.8	0.0	32.0	27.0	38.0	26.2	
Saturday, 01 January 2022	1.641E+12 0.0	89.4	0.0	32.0	27.0	24.0	26.2	
Saturday, 01 January 2022	1.641E+12 0.0	89.4	0.0	32.0	26.0	19.0	26.4	

Gambar 3. Transformasi data CSV yang diperkaya dengan timestamp

### Enhancing Dataset Quality

Proses pre-processing melibatkan beberapa langkah esensial dalam rangka meningkatkan kualitas dataset yang digunakan. Tahapan awal pada data pre-processing tersebut yaitu melakukan data cleansing untuk meningkatkan kualitas dataset hasil ekstraksi sebelumnya. Data cleansing bertujuan membersihkan data dari berbagai masalah, seperti nilai yang hilang (missing values), duplikasi data, mencakup membersihkan dataset dari anomali, noise, serta melakukan pemurnian data untuk memastikan keakuratan informasi maupun inkonsistensi tipe data.

Pertama, tipe data kolom diubah seragam menjadi float64 agar lebih mudah diolah. Kemudian missing values ditangani dengan mengisinya menggunakan nilai rata-rata dari setiap kolom sehingga tidak ada sel kosong. Selanjutnya, seluruh data duplikat dihapus untuk mencegah bias dalam pemodelan machine learning. Terakhir dilakukan scaling dan normalization agar distribusi data dan rentang nilai pada setiap kolom menjadi seragam.

All variables classified into correct types.

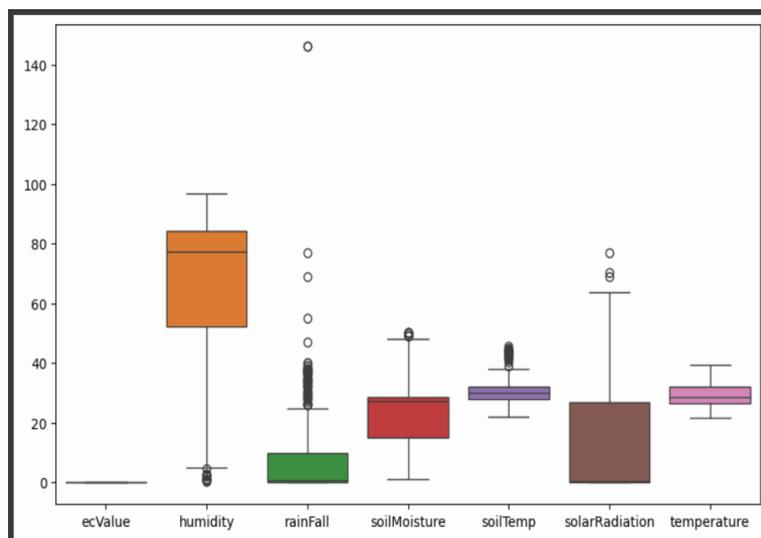
	Data Type	Missing Values%	Unique Values%	Minimum Value	Maximum Value
ecValue	float64	0.000000	NA	0.000000	0.141000
humidity	float64	0.000000	NA	0.500000	96.900000
rainFall	float64	0.000000	NA	0.000000	146.000000
soilMoisture	float64	0.000000	NA	1.000000	50.000000
soilTemp	float64	0.000000	NA	22.000000	45.700000
solarRadiation	float64	0.000000	NA	0.000000	77.000000
temperature	float64	0.000000	NA	21.600000	39.500000

	Data Type	Missing Values%	Unique Values%	Minimum Value	Maximum Value
ecValue	float64	0.0	NA	0.0	0.141
humidity	float64	0.0	NA	0.5	96.900
rainFall	float64	0.0	NA	0.0	146.000
soilMoisture	float64	0.0	NA	1.0	50.000
soilTemp	float64	0.0	NA	22.0	45.700
solarRadiation	float64	0.0	NA	0.0	77.000
temperature	float64	0.0	NA	21.6	39.500

Gambar 4. Pembersihan Data dengan menggunakan Python

Selanjutnya salah satu tahap penting dalam pre-processing data adalah handling outliers. Proses penanganan data outlier menggunakan metode IQR (Interquartile Range) ditunjukkan dalam Gambar 5. Outlier merupakan data observasi yang muncul dengan nilai-nilai ekstrim jauh berbeda dari sebagian besar data lainnya. Keberadaan outlier dapat mengganggu analisis statistik dan menyebabkan model machine learning menjadi bias. Oleh karena itu, penanganan outlier penting dilakukan dalam data cleansing.

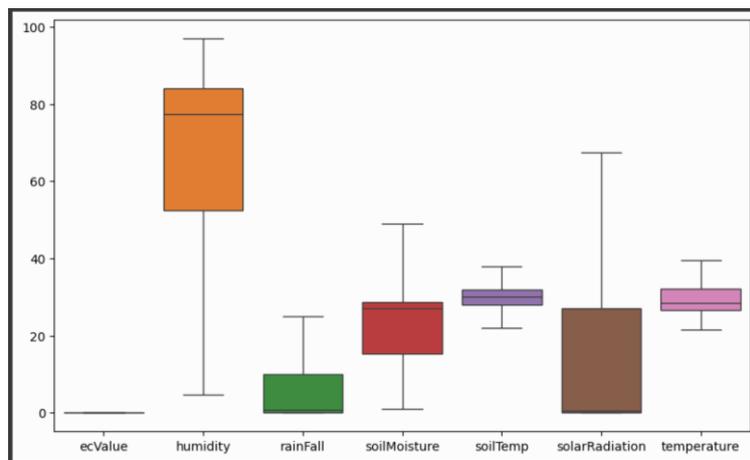


Gambar 5. Boxplot sebelum Handling Outliers

Salah satu metode populer untuk mendeteksi dan menghilangkan outlier adalah interquartile range (IQR). Berikut ini langkah-langkah menangani outlier dengan IQR: Pertama, hitung nilai kuartil bawah ( $Q_1$ ) dan kuartil atas ( $Q_3$ ) dari dataset. Kemudian tentukan nilai IQR dengan mengurangkan  $Q_3$  dan  $Q_1$ . Selanjutnya, tentukan batas bawah (lower fence) dan batas atas (upper fence) outlier. Lower fence bernilai  $Q_1$  dikurangi  $1,5 \times IQR$  sedangkan upper fence bernilai  $Q_3$  ditambah  $1,5 \times IQR$ . Kedua, deteksi outlier dengan mengidentifikasi data yang memiliki nilai lebih kecil dari lower fence atau lebih besar dari upper fence. Data tersebut merupakan outlier. Terakhir, outlier yang terdeteksi kemudian dihapus secara permanen dari dataset untuk dilanjutkan ke tahap analisis berikutnya. Selanjutnya dengan menghilangkan outlier, kualitas data semakin baik sehingga performa machine learning pun ikut meningkat.

Metode ini digunakan untuk mendeteksi dan menghapus nilai-nilai ekstrim atau outlier yang signifikan dalam dataset. Outlier dapat menyebabkan bias dalam analisis data dan harus dihilangkan untuk meningkatkan kualitas dataset.

Setelah proses penanganan outlier menggunakan metode IQR (Interquartile Range) dilakukan, hasilnya dapat dilihat pada Gambar 6. Terlihat dalam gambar tersebut, bahwasanya nilai-nilai ekstrim telah dihapus dari dataset, menghasilkan distribusi data yang lebih normal dan berkualitas tinggi.



**Gambar 6.** Boxplot hasil setelah Handling Outliers

Rangkaian proses data cleansing ini menghasilkan dataset yang jauh lebih bersih dan berkualitas. Data bersih yang diperoleh dari tahapan penting ini siap digunakan untuk analisis dan pemodelan machine learning pada tahap berikutnya. Dengan data cleansing yang tepat, diharapkan model prediksi yang dihasilkan dapat lebih akurat dan dapat diandalkan.

### Evaluating Machine Learning Models

Setelah dataset melewati proses pembersihan, langkah selanjutnya dalam penelitian ini adalah evaluasi dengan membagi dataset menjadi data latih dan data uji. Pembagian ini memungkinkan pengujian performa tiga model machine learning yang berbeda, yaitu logistic regression, random forest, dan decision tree, untuk mendapatkan rekomendasi waktu penyiraman yang optimal. Model yang terbaik akan diterapkan secara konsisten pada dataset training, dan hasilnya kemudian dievaluasi menggunakan dataset testing untuk mengukur sejauh mana kemampuannya dalam melakukan klasifikasi.

Hasil evaluasi model logistic regression yang diterapkan pada training dataset ditampilkan dalam Gambar 7, di mana dilakukan analisis terperinci terhadap matriks-matriks seperti accuracy, precision, recall, dan f1-score untuk masing-masing kelas target. Confusion matrix juga disajikan untuk memberikan gambaran yang lebih jelas tentang prediksi model. Confusion matrix merupakan alat evaluasi yang penting dalam konteks machine learning yang digunakan untuk mengukur kinerja model klasifikasi. Confusion matrix menggambarkan performa model dengan membandingkan prediksi yang dilakukan oleh model dengan nilai sebenarnya dari data yang diamati. Matriks ini terdiri dari empat sel: true positive (TP), true negative (TN), false positive (FP), dan false negative (FN), yang mewakili hasil prediksi yang benar dan yang salah oleh model. Keuntungan utama dari confusion matrix adalah memberikan gambaran yang jelas tentang seberapa baik atau buruk model memprediksi kelas data tertentu, serta memungkinkan kita untuk menghitung metrik evaluasi seperti accuracy, precision, recall, dan f1-score.

Angka-angka dalam confusion matrix muncul sebagai hasil dari perbandingan antara prediksi model dengan nilai sebenarnya (ground truth). Pertama, accuracy mengindikasikan proporsi data yang diklasifikasikan dengan benar oleh model, dihitung dengan membagi jumlah prediksi yang benar dengan total data. Kemudian, presisi menunjukkan proporsi

data yang diprediksi sebagai kelas tertentu yang benar-benar merupakan kelas tersebut. Hal ini diukur dengan membagi jumlah prediksi positif yang benar dengan total prediksi positif.

Sementara itu, recall menggambarkan proporsi data yang sebenarnya merupakan kelas tertentu yang teridentifikasi dengan benar oleh model, dihitung dengan membagi jumlah prediksi positif yang benar dengan total data yang sebenarnya merupakan kelas positif. f1-score, sebagai keseimbangan antara precision dan recall, dihitung dengan dua kali mengalikan precision dan recall, kemudian dibagi dengan jumlah precision dan recall.

Macro avg dan weighted avg memberikan rata-rata precision, recall, dan f1-score untuk kedua kelas. Macro avg dihitung dengan menjumlahkan nilai precision, recall, dan f1-score untuk kedua kelas, lalu dibagi dengan dua. Di sisi lain, weighted avg mempertimbangkan jumlah data di setiap kelas. Matriks ini dihitung dengan menjumlahkan perkalian nilai precision, recall, dan f1-score dengan jumlah data di setiap kelas, kemudian dibagi dengan total data.

```

Logistic Regression:
Accuracy: 0.8426966292134831

Classification Report:
              precision    recall  f1-score   support

     0           0.88       0.88       0.88         59
     1           0.77       0.77       0.77         30

 accuracy                0.84         89
 macro avg              0.82         89
 weighted avg           0.84         89

Confusion Matrix:
[[52  7]
 [ 7 23]]

```

**Gambar 7.** Confusion matrix dari model logistic regression

Secara keseluruhan, angka-angka dalam confusion matrix memberikan gambaran yang komprehensif tentang kinerja model klasifikasi, meliputi accuracy, precision, recall, dan f1-score untuk setiap kelas. Informasi ini sangat berharga bagi para ilmuwan data dalam memilih model terbaik dan meningkatkan kinerja model secara keseluruhan.

Selain model logistic regression, penelitian ini juga melakukan evaluasi perbandingan dengan menerapkan model decision tree dan random forest sebagai classifier dalam prediksi waktu penyiraman optimal.

Ilustrasi penerapan kedua model tersebut pada dataset training yang sama untuk pengujian performa dan analisis ditampilkan pada Gambar 8. Decision tree bekerja dengan membentuk sejumlah aturan if-then dari kombinasi kondisi yang ada, sedangkan random forest membuat banyak pohon keputusan (decision tree) untuk kemudian menggabungkan hasil prediksi dari masing-masing pohon tersebut.

```

# Membuat model Decision Tree
dt_model = DecisionTreeClassifier(random_state=42)

# Melatih model pada data pelatihan
dt_model.fit(X_train, y_train)

# Membuat prediksi pada data pengujian
dt_pred = dt_model.predict(X_test)

# Evaluasi performa model Decision Tree
print("\nDecision Tree:")
print("Accuracy:", accuracy_score(y_test, dt_pred))
print("\nClassification Report:\n", classification_report(y_test, dt_pred))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, dt_pred))

[ ] # Membuat model RandomForest
rf_model = RandomForestClassifier(random_state=42)

# Melatih model pada data pelatihan
rf_model.fit(X_train, y_train)

# Membuat prediksi pada data pengujian
rf_pred = rf_model.predict(X_test)

# Evaluasi performa model RandomForest
print("\nRandom Forest:")
print("Accuracy:", accuracy_score(y_test, rf_pred))
print("\nClassification Report:\n", classification_report(y_test, rf_pred))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, rf_pred))

```

**Gambar 8.** Machine learning model decision tree dan random forest

Evaluasi perbandingan dengan model decision tree dan random forest ini tetap bermanfaat untuk memperkaya pemahaman tentang model mana yang paling cocok digunakan dalam kasus prediksi berbasis data cuaca ini. Meskipun hasil evaluasi sebelumnya menunjukkan bahwa logistic regression memiliki akurasi yang lebih tinggi, analisis ini tetap bermanfaat untuk memperkaya pemahaman tentang model mana yang paling cocok digunakan dalam kasus prediksi berbasis data cuaca ini.

Setelah membandingkan performa decision tree, random forest, dan logistic regression secara keseluruhan, penelitian ini dapat memberikan rekomendasi yang lebih tepat mengenai model terbaik untuk diimplementasikan dalam sistem prediksi waktu penyiraman optimal di perkebunan.

### Machine Learning Test Result

Pada tahap ini, hasil eksperimen dari pengembangan sistem prediksi waktu penyiraman optimal dievaluasi. Setelah menerapkan *model-machine learning*, *logistic regression*, *random forest*, dan *decision tree* pada *dataset*, peneliti melakukan analisis terhadap kinerja masing-masing model. Hasil menunjukkan bahwa model *logistic regression* memberikan akurasi tertinggi dalam mengklasifikasikan waktu penyiraman optimal.

*Logistic regression* adalah model klasifikasi yang cocok untuk kasus biner seperti memprediksi waktu penyiraman (perlu/tidak perlu), karena model ini dapat memetakan hubungan non-linear antara variabel input dan output (H. Pratama Budianto, B. D. Setiawan, and P. P. Adikara., 2017). Selain itu, *logistic regression* dapat menangani korelasi antar variabel prediktor dengan lebih baik dibanding *decision tree* atau *random forest* yang cenderung memperlakukan variabel secara terpisah (R. S. A. Putra, N. Hidayat, and M. A. Fauzi., 2019). Lebih lanjut, *logistic regression* lebih mudah diinterpretasikan dan parameter-parameternya memiliki makna statistik, sedangkan *decision tree* dan *random forest* seringkali lebih sulit untuk dijelaskan (S. A. Listina, R. M. Sampurno, D. Ciptaningtyas, and A. Thoriq., 2022). Maka dari itu, *logistic regression* terbukti menjadi model yang tepat dalam konteks penelitian ini.

Penelitian ini bertujuan untuk menganalisis akurasi prediksi waktu penyiraman optimal menggunakan model *logistic regression*. Hasil penelitian menunjukkan bahwa model *logistic regression* mampu mencapai tingkat akurasi yang signifikan, yaitu 84%. Temuan ini memberikan bukti empiris yang kuat terhadap kemampuan model *logistic regression* dalam mengklasifikasikan waktu penyiraman berdasarkan dataset yang digunakan.

Tingginya tingkat akurasi menunjukkan bahwa model *logistic regression* sesuai dan efektif untuk diterapkan dalam konteks penelitian ini. Keberhasilan ini menggambarkan potensi besar dalam meningkatkan efisiensi dan produktivitas di perkebunan. Rekomendasi waktu penyiraman yang tepat, berdasarkan pada model *logistic regression* yang presisi, dapat memberikan panduan yang berharga bagi para praktisi dan pemangku kepentingan dalam pengambilan keputusan terkait manajemen penyiraman tanaman.

Implementasi model ini dalam praktik pertanian diprediksikan akan menghasilkan peningkatan hasil akhir secara keseluruhan. Hal ini dapat dicapai melalui optimalisasi penggunaan air, pencegahan stres tanaman akibat penyiraman yang tidak tepat, dan peningkatan kualitas panen.

Penelitian ini memberikan kontribusi penting dalam pengembangan sistem manajemen penyiraman tanaman yang presisi dan efisien. Penggunaan model *logistic regression* dapat menjadi solusi inovatif untuk meningkatkan produktivitas dan keberlanjutan di sektor pertanian.

### Heatmap Correlation Validation

*Heatmap correlation validation* merupakan alat analisis yang esensial dalam penelitian untuk menggambarkan hubungan antar variabel dengan memvisualisasikan kekuatan serta arah korelasi menggunakan skema warna yang telah ditentukan. Selanjutnya dalam konteks ini, warna merah menandakan korelasi positif yang signifikan, sementara warna biru menunjukkan korelasi negatif yang kuat, dan warna putih menandakan ketiadaan korelasi yang bermakna antara dua variabel yang diamati. Sebagai contoh, dalam studi mengenai waktu penyiraman optimal, heatmap mampu menggambarkan korelasi positif yang signifikan antara tingkat kelembaban tanah dengan waktu penyiraman, serta korelasi negatif yang kuat antara suhu udara dengan waktu penyiraman.

Selain itu, tahap evaluasi kinerja sensor yang paling mempengaruhi dalam prediksi penyiraman yang optimal dilakukan menggunakan *correlation matrix*. Analisis ini memungkinkan identifikasi sensor-sensor kunci yang memberikan kontribusi signifikan terhadap akurasi model. *Heatmap correlation matrix* ini merepresentasikan tingkat pengaruh berbagai variabel sensor terhadap prediksi waktu penyiraman yang optimal. Terlihat bahwa rainfall, solarRadiation, dan humidity menjadi 3 variabel yang paling berpengaruh signifikan, ditunjukkan dengan warna merah tua pada baris ketiga variabel tersebut. Hujan (*rainfall*) secara langsung mempengaruhi kebutuhan penyiraman. Sedangkan radiasi matahari (*solarRadiation*) dan kelembapan (*humidity*) ikut menentukan tingkat evapotranspirasi potensial tanaman. Ketiga variabel inilah yang paling dominan dalam sistem prediksi waktu penyiraman tanaman. *Heatmap* digunakan untuk evaluasi dan memilih subset variabel sensor yang relevan.

Penentuan tiga variabel yang memiliki pengaruh paling signifikan terhadap waktu penyiraman optimal memiliki implikasi yang substansial dalam praktik pertanian. Pertama, penelitian ini memungkinkan peneliti atau praktisi untuk mengarahkan upaya peningkatan efisiensi penyiraman dengan penekanan faktor-faktor yang paling berpengaruh, seperti

pengelolaan kelembaban tanah. Kedua, dengan memperoleh pemahaman yang lebih mendalam terkait hubungan antar variabel, para peneliti dapat mengembangkan strategi penyiraman yang lebih cermat dan adaptif, yang memperhitungkan variasi suhu udara serta tingkat kelembaban tanah secara simultan. Ketiga, dengan mengidentifikasi waktu penyiraman optimal secara lebih tepat, hasil panen dan kualitas tanaman dapat ditingkatkan, karena tanaman akan mendapatkan pasokan air yang optimal untuk pertumbuhan dan perkembangannya.

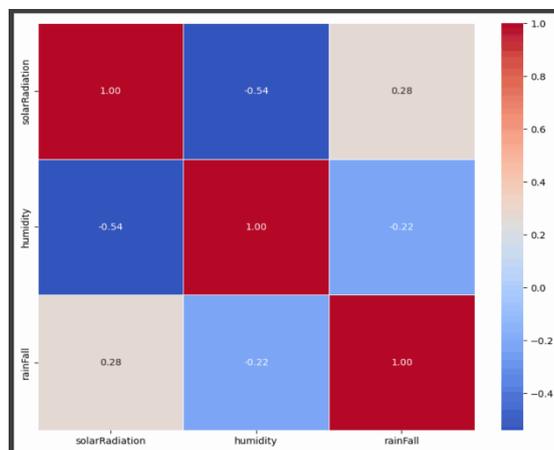


**Gambar 10.** Dashboard indicates 'No need for irrigation'

Maka dari itu, interpretasi warna heatmap serta identifikasi variabel yang paling berpengaruh tidak hanya memberikan kontribusi penting dalam pengambilan keputusan untuk meningkatkan efisiensi penyiraman, namun juga berpotensi untuk memperkaya pengembangan strategi penyiraman yang tepat dan meningkatkan hasil panen secara keseluruhan dalam konteks praktik pertanian yang berkelanjutan.

**Visualization Dashboard**

Selain analisis metrik evaluasi, penelitian ini juga memanfaatkan visualisasi data untuk menyajikan hasil pengujian secara lebih intuitif melalui berbagai representasi grafis. Pendekatan ini bertujuan memberikan pemahaman yang lebih mendalam dan terpadu tentang tingkat keandalan model dalam memprediksi waktu penyiraman optimal. Gambar tersebut menunjukkan tampilan dashboard pada platform thingsboard yang digunakan sebagai antarmuka sistem prediksi waktu penyiraman. Terdapat panel khusus "Rekomendasi Penyiraman" yang memberikan prediksi terkini dari sistem.



**Gambar 11.** Validasi heatmap correlation

Prediksi menunjukkan bahwa "Tidak perlu penyiraman" yang dibuat 1 minggu yang lalu, lengkap dengan icon tanaman. Sistem prediksi waktu penyiraman ini terintegrasi outputnya ke dalam dashboard thingsboard agar petani pengguna dapat dengan mudah memantau rekomendasi terkini dari sistem untuk keperluan penyiraman di perkebunan mereka. Visualisasi yang intuitif dan user-friendly menjadikan hasil pengujian dapat disajikan secara lebih menarik dan memberikan gambaran yang jelas tentang kinerja model dalam memprediksi waktu penyiraman optimal berdasarkan kondisi cuaca terkini.

Berikutnya pada gambar 12 menampilkan contoh lain tampilan dashboard sistem prediksi waktu penyiraman. Terdapat panel khusus "Rekomendasi Penyiraman" yang memberikan prediksi terkini dari sistem.

Terlihat panel "Rekomendasi Penyiraman" kini menunjukkan sistem memprediksi bahwa tanaman membutuhkan penyiraman berdasarkan kondisi terkini yang terdeteksi oleh sensor. Teks "Perlu penyiraman" ditampilkan lengkap dengan icon animasi tanaman yang diwarnai kuning mengindikasikan tanaman haus dan perlu disiram. Prediksi dan

rekomendasi visual ini diharapkan memudahkan petani dalam menentukan kapan waktu yang tepat untuk melakukan irigasi di perkebunan tanpa harus melakukan pengecekan manual di lapangan.



**Gambar 12.** Dashboard indicates 'Need for irrigation'

Oleh karena itu, dengan menggabungkan prediksi, evaluasi kinerja sensor, dan visualisasi data, kedua tahap ini bersinergi untuk membentuk pemahaman menyeluruh tentang kemampuan dan kehandalan model dalam mendukung keputusan waktu penyiraman yang optimal.

## KESIMPULAN

Kesimpulan dari penelitian ini adalah keberhasilan dalam pengembangan sistem prediksi waktu penyiraman optimal menggunakan pendekatan machine learning berdasarkan data cuaca dari automatic weather station (AWS) dengan mengikuti langkah-langkah metodologis yang terstruktur, penelitian ini berhasil menghasilkan model classifier, seperti logistic regression, yang mampu memberikan prediksi waktu penyiraman dengan tingkat keakuratan yang baik. Evaluasi dan validasi menunjukkan bahwa logistic regression terpilih sebagai model terbaik karena kemampuannya menangani kasus klasifikasi biner serta menganalisis korelasi antar variabel prediktor. Hasil prediksi kemudian ditampilkan secara visual dalam bentuk rekomendasi "Perlu Penyiraman" atau "Tidak Perlu Penyiraman" yang disertai representasi icon tanaman. Hasil evaluasi dan validasi juga menunjukkan bahwa sistem prediksi waktu penyiraman optimal ini dapat diandalkan dalam memberikan petunjuk yang berguna untuk kebutuhan penyiraman tanaman. Visualisasi data yang intuitif pada hanya meningkatkan pemahaman terhadap pola penyiraman optimal, tetapi juga memberikan pendekatan interaktif dan mudah digunakan bagi pengguna. Secara keseluruhan, penelitian ini memberikan kontribusi penting pada optimalisasi sistem irigasi presisi di pertanian modern yang adaptif. Sistem prediktif ini dapat menjadi alat yang berharga bagi para petani dalam upaya meningkatkan efisiensi penggunaan air dan sumber daya sekaligus memaksimalkan produktivitas hasil panen.

## DAFTAR PUSTAKA

- Ang, K. L.-M., & Seng, J. K. P. (2021). Big Data and Machine Learning With Hyperspectral Information in Agriculture. *IEEE Access*, 9(2), 36699–36718. <https://doi.org/10.1109/ACCESS.2021.3051196>
- Araújo, S. O., Peres, R. S., Ramalho, J. C., Lidon, F., & Barata, J. (2023). Machine Learning Applications in Agriculture: Current Trends, Challenges, and Future Perspectives. *Agronomy*, 13(12), 2976. <https://doi.org/10.3390/agronomy13122976>
- Attri, I., Awasthi, L. K., & Sharma, T. P. (2023). Machine learning in agriculture: a review of crop management applications. *Multimedia Tools and Applications*, 83(5), 12875–12915. <https://doi.org/10.1007/s11042-023-16105-2>
- Deng, P., Gao, Y., Mu, L., Hu, X., Yu, F., Jia, Y., Wang, Z., & Xing, B. (2023). Development potential of nanoenabled agriculture projected using machine learning. *Proceedings of the National Academy of Sciences*, 120(25). <https://doi.org/10.1073/pnas.2301885120>
- Ferrag, M. A., Shu, L., Friha, O., & Yang, X. (2022). Cyber Security Intrusion Detection for Agriculture 4.0: Machine Learning-Based Solutions, Datasets, and Future Directions. *IEEE/CAA Journal of Automatica Sinica*, 9(3), 407–436. <https://doi.org/10.1109/JAS.2021.1004344>
- Indu, Baghel, A. S., Bhardwaj, A., & Ibrahim, W. (2022). Optimization of Pesticides Spray on Crops in Agriculture using Machine Learning. *Computational Intelligence and Neuroscience*, 2022(1), 1–10. <https://doi.org/10.1155/2022/9408535>
- Menaha, M., & Lavanya, J. (2023a). Machine Learning Techniques in Agriculture. *REST Journal on Data Analytics and Artificial Intelligence*, 2(3), 23–25. <https://doi.org/10.46632/jdaai/2/3/5>
- Menaha, M., & Lavanya, J. (2023b). Machine Learning Techniques in Agriculture. *REST Journal on Data Analytics and Artificial Intelligence*, 2(3), 23–25. <https://doi.org/10.46632/jdaai/2/3/5>
- Menaha, M., & Lavanya, J. (2023c). Machine Learning Techniques in Agriculture. *REST Journal on Data Analytics and Artificial Intelligence*, 2(3), 23–25. <https://doi.org/10.46632/jdaai/2/3/5>
- Mohan, D. S., Dhote, V., Mishra, P., Singh, P., & Srivastav, A. (2023). IoT Framework for Precision Agriculture: Machine Learning Crop Prediction. *International Journal of Intelligent Systems and Applications in Engineering*.

- Mohinur Rahaman, M., & Azharuddin, M. (2022). Wireless sensor networks in agriculture through machine learning: A survey. *Computers and Electronics in Agriculture*, 197(1), 106928. <https://doi.org/10.1016/j.compag.2022.106928>
- Morota, G., Ventura, R. V., Silva, F. F., Koyama, M., & Fernando, S. C. (2018). Big data analytics and precision animal agriculture symposium: Machine learning and data mining advance predictive big data analysis in precision animal agriculture1. *Journal of Animal Science*, 96(4), 1540–1550. <https://doi.org/10.1093/jas/sky014>
- Prema, P., Veeramani, A. & Sivakumar, T. (2022). Machine learning applications in agriculture. *Journal of Agriculture Research and Technology, Special(01)*, 126–129. <https://doi.org/10.56228/JART.2022.SP120>
- Pallathadka, H., Mustafa, M., Sanchez, D. T., Sekhar Sajja, G., Gour, S., & Naved, M. (2023). Impact of machine learning on management, healthcare and agriculture. *Materials Today: Proceedings*, 80, 2803–2806. <https://doi.org/10.1016/j.matpr.2021.07.042>
- Priyani, S., Hakani, R., & Jhala, D. K. (2023). Smart Agriculture system Using Machine Learning. *Researchgate.Net*.
- Saba, T., Rehman, A., Haseeb, K., Bahaj, S. A., & Lloret, J. (2023). Trust-based decentralized blockchain system with machine learning using Internet of agriculture things. *Computers and Electrical Engineering*, 108(2), 108674. <https://doi.org/10.1016/j.compeleceng.2023.108674>
- Sharma, A., Jain, A., Gupta, P., & Chowdary, V. (2021). Machine learning applications for precision agriculture: A Comprehensive Review. *IEEE Access*, 9(2), 4843–4873. <https://doi.org/10.1109/ACCESS.2020.3048415>
- Sharma, P., Dadheech, P., Aneja, N., & Aneja, S. (2023). Predicting agriculture yields based on machine learning using regression and deep learning. *IEEE Access*, 11, 111255–111264. <https://doi.org/10.1109/ACCESS.2023.3321861>
- Veeragandham, S., & H, S. (2020). A review on the role of machine learning in agriculture. *Scalable Computing: Practice and Experience*, 21(4), 583–589. <https://doi.org/10.12694/scpe.v21i4.1699>