BAB II
TINJAUAN PUSTAKA

2.1 Kajian Relevan

Kualitas udara di daerah perkotaan menjadi masalah kritis yang langsung memengaruhi kesehatan masyarakat. Dengan bertambahnya jumlah penduduk dan meningkatnya aktivitas industri, polusi udara semakin mengkhawatirkan dan mendorong kebutuhan akan penelitian dalam bidang peramalan atau prediksi kualitas udara. Salah satu cara yang sangat efektif untuk menghadapi masalah ini adalah dengan menggunakan teknologi *machine learning*.

Penelitian sebelumnya sudah menggunakan berbagai algoritma *machine learning* untuk memprediksi Indeks Standar Pencemar Udara (ISPU) atau kadar polutan tertentu. Mayoritas penelitian ini berfokus pada tugas klasifikasi, yaitu mengelompokkan kualitas udara ke dalam kategori tertentu seperti Baik, Sedang, atau Tidak Sehat. Misalnya, penelitian (Jayadi et al., 2023) membandingkan algoritma *K-Nearest Neighbor* (K-NN) dan *Support Vector Machine* (SVM) untuk klasifikasi kualitas udara di Jakarta, dan hasilnya menunjukkan akurasi 98% untuk SVM. Sementara itu (Amalia et al., 2022) menggunakan K-NN pada data Jakarta dan mencapai akurasi sebesar 96%. Penelitian lain oleh (Luthfi & Fauzi, 2024) membandingkan algoritma *Random Forest, Support Vector Machines*, dan LGBM untuk klasifikasi kualitas udara di Jakarta. Di sisi lain, (Kannajmi & Saputra, 2025) menemukan bahwa *Random Forest* dan LGBM memberikan hasil terbaik dalam tugas klasifikasi tersebut.

Namun, beberapa penelitian juga menggunakan pendekatan regresi untuk memperkirakan nilai dari polutan atau Indeks Standar Pencemar Udara (ISPU). Pendekatan ini dianggap lebih tepat karena mampu memberikan angka prediksi yang lebih detail, sehingga bisa dibandingkan dengan data nyata. Dalam jurnal karya (Gupta et al., 2023), mereka secara langsung menggunakan tiga metode regresi, yaitu *Support Vector Regression* (SVR), *Random Forest Regression* (RFR), dan *CatBoost Regression* (CR), untuk memperkirakan Indeks Kualitas Udara di beberapa kota di India. Hasil dari penelitian tersebut dievaluasi menggunakan beberapa metrik seperti

RMSE, MAE, dan R^2. Dari hasil penelitian tersebut, ditemukan bahwa prediksi memiliki kesamaan yang cukup baik dengan data nyata. Di sisi lain, jurnal karya (Syahreza et al., 2024) juga melakukan pembandingan antara *Random Forest*, SVR, dan *XGBoost* untuk memprediksi kondisi cuaca, dan menemukan bahwa *XGBoost* merupakan model terbaik dengan nilai R^2 sebesar 0,8183, yang menunjukkan tingkat kesamaan (akurasi) model sangat tinggi, sekitar 81,83% dibandingkan dengan data nyata.

Berdasarkan tinjauan tersebut, terdapat celah penelitian yang signifikan. Meskipun algoritma *Random Forest* dan *XGBoost* terbukti unggul dalam tugas regresi, penelitian yang secara khusus membandingkan kedua algoritma tersebut untuk memprediksi nilai ISPU di Jakarta dengan menyertakan variabel meteorologi sebagai fitur tambahan masih terbatas. Selain itu, penggunaan data agregat bulanan dalam penelitian ini diharapkan dapat menangkap tren kualitas udara jangka menengah secara lebih stabil. Oleh karena itu, penelitian ini bertujuan mengisi celah tersebut untuk memberikan model prediksi yang lebih presisi dan aplikatif bagi instansi terkait.

1. Using *machine learning* for air quality prediction and sustainable urban planning

Tabel 2. 1 *State Of The Art*

Penulis	M.A. Mujtaba, dkk.
Tahun	2025
Metode	Penelitian ini menggunakan pendekatan *time-series* dan *deep learning*, dengan model spesifik seperti SARIMA, SARIMAX, *Non-linear Autoregressive* (NAR), dan *Long Short-Term Memory* (LSTM). Model ARIMA juga digunakan.
Fitur/Variabel	Data yang digunakan mencakup delapan polutan (*Carbon monoxide* (CO), *Nitrogen oxide* (NO), *Nitrogen dioxide* (NO_2), *Ozone* (O_3), *Sulphur dioxide* (SO_2), *Particulate matter* 1 (PM_1), PM2.5, dan PM10) serta empat faktor meteorologi (seperti suhu dan komponen angin)

Dataset	Data konsentrasi polutan dikumpulkan dari *Copernicus Atmosphere Monitoring Service* (CAMS). Datasetnya terdiri dari 7.305 observasi untuk setiap parameter, mencakup periode dari Januari 2003 hingga Desember 2022.
Akurasi/Kinerja Terbaik	Akurasi model dievaluasi dengan *Root Mean Squared Error* (RMSE). Model NAR dinilai sebagai yang terbaik dengan RMSE minimum 23,52. Jurnal ini juga memprediksi adanya peningkatan Indeks Kualitas Udara (AQI) sebesar 13% pada tahun 2030 dibandingkan tahun 2022
Keterkaitan Penelitian	Keterkaitan: Sama-sama melakukan prediksi kualitas udara menggunakan machine learning. Perbandingan: Penelitian ini menggunakan model *deep learning* pada skala global dengan rentang waktu yang panjang, sedangkan penelitian saya menggunakan algoritma *ensemble learning* yang lebih efisien untuk data skala menengah. Kebaruan penelitian ini terletak pada penggunaan variabel meteorologi yang lebih lengkap, dengan fokus khusus pada karakteristik kualitas udara di DKI Jakarta.

2. Monitoring air quality index with EWMA and individual charts using *XGBoost* and SVR residuals

Penulis	Zulfani Alfasanah, dkk.
Tahun	2025
Metode	Penelitian ini menggunakan model hibrida yang mengintegrasikan algoritma *XGBoost* dan SVR (*Support Vector Regression*) untuk memprediksi Indeks Standar Pencemar Udara (ISPU).
Fitur/Variabel	Konsentrasi polutan udara, termasuk PM2.5, NO_2, SO_2, dan CO.
Dataset	Data kualitas udara harian dari platform AQICN di Jakarta, dari 1 Januari 2023 hingga 31 Mei 2024

Akurasi/Kinerja Terbaik	Jurnal ini lebih berfokus pada performa pemantauan dan kontrol kualitas, bukan pada akurasi prediksi model utama. Namun, disebutkan bahwa model *XGBoost* memiliki performa yang optimal untuk mendeteksi anomali.
Keterkaitan Penelitian	Keterkaitan: Sama-sama menggunakan algoritma *XGBoost* dan data kualitas udara di Jakarta. Perbandingan: Penelitian ini menggunakan model hibrida *XGBoost-SVR* untuk deteksi anomali pada data harian tanpa variabel cuaca. Sementara itu, penelitian saya melakukan perbandingan murni *XGBoost* dan *Random Forest Regressor* dengan mengintegrasikan 12 variabel (polutan + meteorologi) menggunakan data bulanan untuk mememprediksi nilai ISPU maksimum di Jakarta.

3. Explainable *forecasting* of air quality index using a *hybrid random forest* and ARIMA model

Penulis	Anuradha Yenkikara, dkk.
Tahun	2025
Metode	Model hibrida yang menggabungkan *Random Forest* (regresi) dengan ARIMA untuk koreksi residu.
Fitur/Variabel	$PM_{2.5}$, NO_2, dan SO_2
Dataset	Data AQI multi-tahun dari India.
Akurasi/Kinerja Terbaik	MSE=508.46, R2=0.94. Ini berarti model mampu menjelaskan 94% variasi data riil, menunjukkan kesamaan yang sangat tinggi.
Keterkaitan Penelitian	Keterkaitan: Sangat relevan karena sama-sama menggunakan algoritma *Random Forest* untuk memprediksi AQI. Perbandingan: Penelitian ini menggunakan model hibrida RF-ARIMA dengan fitur terbatas (3 polutan) di India. Sementara itu, penelitian saya berfokus pada perbandingan murni *Random Forest* dan *XGBoost Regressor* di Jakarta dengan menggunakan

	12 variabel lengkap (polutan dan meteorologi) untuk mendapatkan prediksi ISPU yang lebih komprehensif.

4. Prediction of Air Quality Index Using *Machine Learning* Techniques: A Comparative Analysis

Penulis	N. Srinivasa Gupta, dkk.
Tahun	2023
Metode	Penelitian ini menggunakan tiga metode regresi: *Support Vector Regression* (SVR), *Random Forest Regression* (RFR), dan *CatBoost Regression* (CR). Metode SMOTE (*Synthetic Minority Oversampling Technique*) juga digunakan untuk menyeimbangkan dataset
Fitur/Variabel	Variabel yang digunakan mencakup beberapa polutan, seperti Particulate Matter (PM_2.5 dan PM_10), CO, NO, NO_2, SO_2, Amonia (NH_3), Ozon (O_3), Benzene, dan Toluene
Dataset	Dataset dikumpulkan dari stasiun cuaca dan pemantauan lingkungan di beberapa kota di India, yaitu New Delhi, Bangalore, Kolkata, dan Hyderabad. Penelitian ini tidak menyebutkan jumlah total data secara eksplisit
Akurasi/Kinerja Terbaik	Akurasi model diukur menggunakan metrik seperti *Root Mean Square Error* (RMSE), *Mean Absolute Error* (MAE), dan R-SQUARE. Akurasi yang dilaporkan untuk dataset yang seimbang (dengan SMOTE) adalah sebagai berikut: *Random Forest Regression* mencapai akurasi tertinggi di Kolkata (93.7438%) dan Hyderabad (97.6080%) . *CatBoost Regression* mencapai akurasi tertinggi di New Delhi (85.0847%) dan Bangalore (90.3071%)
Keterkaitan Penelitian	Keterkaitan: Sangat relevan karena sama-sama menggunakan pendekatan regresi untuk memprediksi indeks kualitas udara,

	dan salah satu algoritma yang digunakan adalah *Random Forest*. Perbandingan: Penelitian in dilakukan di India, sementara penelitian saya berfokus pada Jakarta dengan membandingkan *Random Forest* dan *XGBoost*, yang tidak dilakukan oleh penelitian ini.

5. Perbandingan Kinerja Model Prediksi Cuaca: *Random Forest*, *Support Vector Regression*, dan *XGBoost*

Penulis	Ahmad Syahreza, dkk.
Tahun	2024
Metode	Penelitian ini membandingkan tiga algoritma machine learning: *Random Forest, Support Vector Regression* (SVR), dan *XGBoost*
Fitur/Variabel	Variabel yang digunakan meliputi suhu minimum, suhu maksimum, curah hujan, arah angin, dan kelembaban rata-rata.
Dataset	Dataset berasal dari data meteorologi BMKG dan sensor IoT. Terdapat 1.650 data per variabel, yang dikumpulkan dari Januari 2020 hingga Agustus 2024
Akurasi/Kinerja Terbaik	Akurasi dievaluasi menggunakan *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan *R-squared* (R^2). Model terbaik adalah *XGBoost* dengan performa sebagai berikut: - MAE: 0,3744 - MSE: 0,2278 - R^2: 0,8183
Keterkaitan Penelitian	Keterkaitan: Jurnal ini sangat relevan karena membandingkan dua algoritma yang sama dengan penelitian saya, yaitu *Random Forest* dan *XGBoost*. Perbandingan: Penelitian ini membandingkan RF dan *XGBoost* untuk prediksi parameter cuaca menggunakan data harian. Sementara itu, penelitian saya

	menggunakan algoritma yang sama untuk memprediksi nilai ISPU maksimum dengan mengintegrasikan 12 variabel gabungan (polutan dan meteorologi).

6. Prediksi Kualitas Udara Menggunakan Algoritma *K-Nearest Neighbor*

Penulis	Adinda Amalia, dkk.
Tahun	2022
Metode	Penelitian ini menggunakan teknik data mining dengan metode klasifikasi, khususnya algoritma *K-Nearest Neighbor* (K-NN). Penentuan nilai K yang optimal dilakukan dengan pengujian K=3 sampai K=9.
Fitur/Variabel	Variabel atau atribut yang digunakan antara lain tanggal, stasiun, pm10, pm25, so2, co, o3, no2, max, critical, dan categori.
Dataset	Dataset didapatkan dari situs web Open Data Jakarta. Dataset ini memuat informasi kualitas udara DKI Jakarta dari Januari hingga Maret 2021. Jumlah total data yang digunakan adalah 450 data.
Akurasi/Kinerja Terbaik	Akurasi dievaluasi menggunakan akurasi, presisi, recall, dan f-measure. Nilai K=7 memiliki performa terbaik dengan akurasi tertinggi sebesar 96%.
Keterkaitan Penelitian	Keterkaitan: Sama-sama melakukan prediksi kualitas udara di Jakarta. Perbandingan: Penelitian ini menggunakan metode klasifikasi K-NN dengan data terbatas selama 3 bulan (450 entri harian) tanpa variabel cuaca. Sementara itu, penelitian saya menggunakan pendekatan regresi (RF dan *XGBoost*) dengan data agregat bulanan yaitu Januari 2020 – Desember 2022 serta mengintegrasikan 12 variabel lengkap (polutan dan meteorologi) untuk memprediksi nilai ISPU secara presisi.

7. Penentuan Model Algoritma Klasifikasi Terbaik Untuk Klasifikasi Kualitas Udara di Jakarta 2023

Penulis	Ahmad Rafi Kannajmi dan Dicky Saputra
Tahun	2025
Metode	Penelitian ini membandingkan beberapa algoritma klasifikasi: *Naive Bayes*, *Random Forest*, *Neural Network*, *K-Nearest Neighbors* (KNN), *Decision Tree*, dan *Support Vector Machine* (SVM).
Fitur/Variabel	Variabel yang digunakan mencakup parameter polutan utama seperti PM2.5, SO2, NO2, CO, dan O3. Dataset juga mencakup PM10, tanggal, stasiun, max, parameter_pencemar_kritis, dan kategori.
Dataset	Dataset berasal dari Stasiun Pengendalian Kualitas Udara (SPKU) di Jakarta yang datanya diperoleh dari situs web resmi Pemerintah Provinsi DKI Jakarta. Jumlah total data adalah 1.804 entri.
Akurasi/Kinerja Terbaik	Akurasi dievaluasi menggunakan metrik seperti akurasi, *recall*, *F1-score*, dan presisi. Model terbaik adalah *Random Forest* dengan akurasi 95,3%.
Keterkaitan Penelitian	Keterkaitan: Penelitian ini sama-sama berlokasi di Jakarta dan menggunakan *Random Forest*. Perbandingan: Penelitian ini menggunakan *Random Forest* untuk klasifikasi kategori kualitas udara dengan fitur yang hanya terbatas pada parameter polutan. Sementara itu, penelitian saya menggunakan *Random Forest Regressor* dan *XGBoost* untuk memprediksi nilai angka ISPU dengan mengintegrasikan 12 variabel gabungan (polutan dan meteorologi).

8. Prediksi Kualitas Udara Menggunakan Metode *CatBoost*

Penulis	Mohamad Arif Abdul Syukur, dkk.
Tahun	2025

Metode	Penelitian ini menggunakan metode *CatBoost* untuk memprediksi indeks kualitas udara. Model ini diuji menggunakan *GridSearchCV* untuk mencari kombinasi parameter terbaik.
Fitur/Variabel	Fitur yang digunakan adalah tanggal, stasiun, pm10, so2, co, o3, no2, max, kritis, dan kategori. Namun, tanggal, stasiun, dan kritis dihapus karena pengaruhnya kecil.
Dataset	Dataset ini menggunakan data Indeks Standar Pencemar Udara (ISPU) di wilayah Jakarta tahun 2020 yang diambil dari Kaggle. Jumlah total data yang digunakan adalah 1.830 entri.
Akurasi/Kinerja Terbaik	Akurasi dievaluasi menggunakan akurasi, presisi, *recall*, dan *F1 Score*. Model terbaik, dengan pembagian data latih 90% dan data pengujian 10%, menghasilkan akurasi sebesar 97%.
Keterkaitan Penelitian	Keterkaitan: Memiliki kesamaan lokasi dan topik, yaitu prediksi indeks kualitas udara di Jakarta. Perbandingan: Penelitian ini menggunakan algoritma *CatBoost* untuk klasifikasi dengan dataset terbatas pada tahun 2020 tanpa variabel cuaca. Sementara itu, penelitian saya melakukan perbandingan kinerja *Random Forest* dan *XGBoost Regressor* untuk memprediksi nilai kontinu ISPU dengan mengintegrasikan 12 variabel (polutan dan meteorologi) selama periode 2020-2022.

9. A Web-Based *Supervised Machine Learning* Model for Air Quality Index and Respiratory Care Prediction

Penulis	Mukund Pratap Singh, dkk.
Tahun	2025
Metode	Penelitian ini menggunakan beberapa model klasifikasi dan *regresi supervised machine learning*, termasuk *Random Forest*, *XGBoost*, *k-nearest neighbors* (KNN), *Decision Trees*, dan

	Support Vector Machines (SVM). Algoritma *Decision Tree* juga digunakan untuk mengidentifikasi tindakan pencegahan.
Fitur/Variabel	Variabel yang digunakan mencakup 7 polutan lingkungan: CO, PM10, NO2, PM2.5, SO2, NH3, dan O3.
Dataset	Dataset dikumpulkan dari portal web https://waqi.info/ dari Januari 2014 hingga April 2024, yang mencakup data dari empat kota di India.
Akurasi/Kinerja Terbaik	Akurasi diukur dengan metrik regresi seperti MAE, RMSE, dan *R-squared* (R^2), serta metrik klasifikasi seperti akurasi, presisi, *recall*, dan *F1 score*. Model *XGBoost* mencapai akurasi sempurna 1.0 setelah penyesuaian *hyperparameter*. Model *Random Forest* mencapai MAE 0,97 dan *R-squared* 0,998.
Keterkaitan Penelitian	Keterkaitan: Sama-sama bertujuan untuk memprediksi indeks kualitas udara menggunakan machine learning. Perbandingan: Penelitian ini menggunakan berbagai model untuk klasifikasi dan regresi berbasis web di India dengan fitur terbatas pada parameter polutan. Sementara itu, penelitian saya berfokus pada perbandingan mendalam *Random Forest* dan *XGBoost Regressor* di Jakarta dengan keunggulan berupa integrasi 6 variabel meteorologi yang tidak digunakan dalam penelitian ini.

10. Global air quality index prediction using integrated spatial observation data and geographics *machine learning*

Penulis	Tania Septi Anggraini, dkk.
Tahun	2025
Metode	Penelitian ini menggunakan beberapa model pembelajaran mesin geografis (*geographics machine learning*) dan regresi (*regression*), termasuk *Random Forest Regression*, *Support Vector Regression*, dan *Neural Network*.

Fitur/Variabel	Variabel yang digunakan mencakup data satelit seperti konsentrasi aerosol, SO2, NO2, CO, HCHO, O3, dan UVAerosolIndex, serta variabel meteorologi seperti suhu, kelembaban, dan curah hujan.
Dataset	Dataset dikumpulkan dari pengamatan satelit Sentinel-5P dan Aqua MODIS yang mencakup data dari seluruh dunia. Jurnal ini tidak secara spesifik menyebutkan jumlah total data yang digunakan.
Akurasi/Kinerja Terbaik	Akurasi dievaluasi menggunakan *Mean Absolute Error* (MAE), *Root Mean Squared Error* (RMSE), dan *R-squared* (R^2). Model *Random Forest* menunjukkan kinerja terbaik dengan nilai *R-squared* (R^2) di atas 0,97.
Keterkaitan Penelitian	Keterkaitan: Sama-sama bertujuan untuk memprediksi indeks kualitas udara. Perbandingan: Penelitian ini menggunakan data satelit (Sentinel-5P) dan model geografis untuk prediksi skala global. Sementara itu, penelitian saya menggunakan data stasiun pemantauan resmi dari Dinas Lingkungan Hidup DKI Jakarta serta membandingkan performa *Random Forest* dan *XGBoost RegressorI.*

11. Perbandingan Klasifikasi *Random Forest*, *Support Vector Machines*, dan LGBM Pada Klasifikasi Kualitas Udara di Jakarta

Penulis	Anisa Ma'u Luthfi dan Fatkhurokhman Fauzi
Tahun	2024
Metode	Penelitian ini membandingkan tiga algoritma klasifikasi: *Random Forest*, *Support Vector Machines* (SVM), dan *Light Gradient Boosting Machine* (LGBM).
Fitur/Variabel	Data yang digunakan mencakup beberapa polutan seperti PM10, PM2.5, SO2, CO, O3, dan NO2. Penelitian juga menyebutkan tanggal, stasiun, max,

	parameter_pencemar_kritis, dan kategori.
Dataset	Dataset diambil dari situs web resmi Indeks Standar Pencemar Udara (ISPU) dan mencakup data dari Jakarta.
Akurasi/Kinerja Terbaik	Akurasi diukur dengan metrik seperti akurasi, presisi, *recall*, dan *f1-score*. Model LGBM menunjukkan kinerja terbaik dengan akurasi 99,06%.
Keterkaitan Penelitian	Keterkaitan: Topik dan lokasi penelitian ini sangat relevan dengan penelitian saya karena sama-sama membandingkan algoritma *Random Forest* untuk kualitas udara di Jakarta. Perbandingan: Penelitian ini menggunakan pendekatan klasifikasi untuk menentukan kategori kualitas udara dengan fitur yang hanya terbatas pada parameter polutan. Sementara itu, penelitian saya menggunakan pendekatan regresi *Random Forest* dan *XGBoost Regressor* untuk memprediksi nilai angka ISPU secara spesifik serta mengintegrasikan 6 variabel meteorologi sebagai fitur tambahan.

12. Perbandingan KNN dan SVM untuk Klasifikasi Kualitas Udara di Jakarta

Penulis	Bryan Valentino Jayadi, dkk.
Tahun	2023
Metode	*K-Nearest Neighbor* (K-NN) dan *Support Vector Machine* (SVM)
Fitur/Variabel	Kategori kualitas udara (berfokus pada klasifikasi)
Dataset	Data ISPU dari situs Open Data Jakarta (Januari 2010 - Desember 2021). Total 4.383 data.
Akurasi/Kinerja Terbaik	SVM mencapai akurasi 98%. K-NN mencapai akurasi 96%. Karena ini adalah tugas klasifikasi, kesamaan diukur berdasarkan akurasi prediksi kategori yang tepat, bukan kesamaan nilai ISPU.

Keterkaitan Penelitian	Keterkaitan: Penelitian ini sangat relevan karena sama-sama melakukan perbandingan algoritma untuk klasifikasi kualitas udara di Jakarta. Perbandingan: Penelitian ini menggunakan metode klasifikasi (K-NN dan SVM) untuk menentukan kategori kualitas udara. Sementara itu, penelitian saya menggunakan pendekatan regresi (RF dan *XGBoost Regressor*) untuk memprediksi nilai angka ISPU secara presisi.

Berdasarkan tinjauan literatur, mayoritas penelitian prediksi kualitas udara di Jakarta menggunakan pendekatan klasifikasi yang mengelompokkan ISPU ke dalam kategori tertentu. Meskipun menunjukkan akurasi tinggi, pendekatan ini tidak dapat memprediksi nilai ISPU secara numerik yang lebih detail. Sementara itu, penelitian dengan pendekatan regresi untuk memprediksi nilai ISPU telah dilakukan, namun sebagian besar berlokasi di India dengan karakteristik polusi berbeda dari Jakarta. Beberapa studi membandingkan *Random Forest* dan *XGBoost*, tetapi fokusnya pada topik lain seperti prediksi cuaca.

Kebaharuan penelitian ini terletak pada tiga aspek utama. Pertama, menggunakan pendekatan regresi untuk memprediksi nilai ISPU maksimum secara numerik di Jakarta. Kedua, mengintegrasikan 12 variabel komprehensif yang terdiri dari enam parameter polutan (PM2.5, PM10, SO2, CO, O3, NO2) dan enam variabel meteorologi (GRAD W/m2, TempAir degrees C, HumAir %, FF m/s, DD degrees, RAIN mm) sebagai input model. Ketiga, membandingkan secara spesifik performa *Random Forest Regressor* dan *XGBoost Regressor* untuk prediksi ISPU di DKI Jakarta dengan melibatkan variabel meteorologi.

Penelitian ini bertujuan mengisi kesenjangan literatur dengan memberikan bukti empiris mengenai algoritma regresi paling efektif untuk memprediksi nilai ISPU di Jakarta, serta berkontribusi dalam pengembangan sistem prediksi kualitas udara yang lebih akurat dan aplikatif untuk wilayah DKI Jakarta.

2.2 Landasan Teori

Landasan teori dalam penelitian ini menyajikan konsep-konsep fundamental yang menjadi dasar analisis perbandingan algoritma *Random Forest* dan *XGBoost* untuk memodelkan ISPU di Jakarta. Pembahasan mencakup konsep polusi udara dan parameter polutannya, sistem pengukuran ISPU, *machine learning* dan *ensemble learning*, penjelasan detail kedua algoritma yang dibandingkan, serta metrik evaluasi model regresi. Teori-teori ini disusun berdasarkan literatur dari buku, jurnal ilmiah, dan artikel terpercaya yang relevan dengan topik penelitian.

2.2.1 Polusi Udara

Menurut *World Health Organization* (WHO, 2021), polusi udara merujuk pada pencemaran udara dalam ruangan dan luar ruangan (*ambient/outdoor air pollution*) yang disebabkan oleh berbagai agen kimia, fisik, dan biologis yang mengubah sifat dasar atmosfer. Polusi udara merupakan campuran kompleks dari partikel padat, tetesan cairan, dan gas yang dapat membahayakan kesehatan manusia, hewan, dan tumbuhan. WHO memperkirakan bahwa polusi udara menyebabkan sekitar tujuh juta kematian dini setiap tahun secara global, dengan lebih dari separuh kematian tersebut terjadi di negara-negara berkembang (WHO, 2021).

Di Indonesia, sumber polusi udara dapat diklasifikasikan menjadi dua kategori utama, yaitu antropogenik dan alami. Sumber antropogenik yang paling menonjol, terutama di daerah perkotaan, meliputi emisi kendaraan bermotor, aktivitas industri, dan pembakaran biomassa (Syukur & Chamidy, 2025). Sumber-sumber ini menghasilkan berbagai polutan seperti partikulat ($PM_{2.5}$ dan PM_{10}), nitrogen dioksida (NO_2), sulfur dioksida (SO_2), karbon monoksida (CO), dan senyawa organik volatil yang berkontribusi signifikan terhadap penurunan kualitas udara di wilayah urban.

Dampak polusi udara terhadap kesehatan manusia sangat mengkhawatirkan dan mencakup berbagai sistem organ. Paparan terhadap polutan, terutama $PM_{2.5}$, telah dikaitkan dengan peningkatan risiko gangguan pernapasan seperti asma dan bronkitis, penyakit kardiovaskular termasuk serangan jantung dan stroke, serta penurunan fungsi paru-paru (Mujtaba et al., 2025). Kelompok yang paling rentan

terhadap dampak polusi udara adalah anak-anak, lansia, ibu hamil, dan individu dengan kondisi kesehatan yang sudah ada sebelumnya.

Parameter polutan yang diukur dan digunakan sebagai dasar perhitungan Indeks Standar Pencemar Udara (ISPU) di Indonesia terdiri dari $PM_{2.5}$, PM_{10}, SO_2, CO, O_3, dan NO_2 sebagaimana diatur dalam Peraturan Menteri Lingkungan Hidup dan Kehutanan Nomor P.14/MENLHK/SETJEN/KUM.1/7/2020 (KLHK, 2020).

2.2.2 Indeks Standar Pencemar Udara (ISPU)

Sistem penilaian kualitas udara di Indonesia diatur oleh Kementerian Lingkungan Hidup dan Kehutanan (KLHK) melalui Peraturan Menteri LHK RI Nomor P.14/MENLHK/SETJEN/KUM.1/7/2020 tentang Indeks Standar Pencemar Udara (KLHK, 2020). Menurut peraturan ini, ISPU adalah angka yang tidak memiliki satuan dan digunakan untuk menunjukkan tingkat kualitas udara ambien di lokasi tertentu, yang didasarkan pada dampak terhadap kesehatan manusia, nilai estetika, dan makhluk hidup lainnya.

ISPU diklasifikasikan ke dalam lima kategori berdasarkan rentang nilai: Baik (1-50), Sedang (51-100), Tidak Sehat (101-200), Sangat Tidak Sehat (201-300), dan Berbahaya (≥301) (KLHK, 2020). Namun, dalam penelitian ini, kategori ISPU tersebut tidak digunakan sebagai kelas prediksi. Penelitian difokuskan pada pemodelan regresi untuk memprediksi nilai numerik ISPU secara langsung.

ISPU dihitung berdasarkan konsentrasi parameter polutan yang diukur menggunakan persamaan interpolasi linear. Nilai ISPU yang dilaporkan adalah nilai tertinggi dari seluruh parameter yang diukur, dan parameter yang menghasilkan nilai tertinggi tersebut disebut sebagai parameter pencemar kritis (KLHK, 2020).

2.2.3 *Machine Learning*

Machine Learning (ML) merupakan cabang dari kecerdasan buatan (*artificial intelligence*) yang berfokus pada pengembangan sistem yang dapat belajar dari data, menemukan pola, dan membuat prediksi tanpa diprogram secara eksplisit untuk setiap tugas (Santoso, 2022). Prinsip dasar *machine learning* adalah membangun model matematis yang dapat mempelajari hubungan antara variabel input (X) dan

output (Y) melalui proses pelatihan dengan data, sehingga model dapat memprediksi output untuk data baru yang belum pernah dilihat sebelumnya (Chyan et al., 2024).

Machine learning dapat diklasifikasikan menjadi tiga kategori utama berdasarkan cara pembelajaran dan jenis data yang digunakan (Chyan et al., 2024):

1. *Supervised Learning*: Metode pembelajaran yang menggunakan data berlabel (*labeled data*). Metode ini dibagi menjadi klasifikasi untuk memprediksi kategori diskrit dan regresi untuk memprediksi nilai kontinu. Penelitian ini menggunakan pendekatan regresi untuk memprediksi nilai ISPU numerik.
2. *Unsupervised Learning*: Menggunakan data tidak berlabel untuk menemukan struktur dan pola tersembunyi dalam data, umumnya untuk *clustering* atau reduksi dimensi.
3. *Reinforcement* Learning: Metode pembelajaran di mana agen belajar melalui interaksi dengan lingkungan untuk memperoleh hasil optimal secara bertahap.

Machine learning, khususnya metode *ensemble learning* seperti *Random Forest* dan *XGBoost*, telah terbukti efektif dalam prediksi kualitas udara karena kemampuannya menangani data multivariat dengan hubungan non-linear yang kompleks dan memberikan akurasi prediksi yang tinggi (Singh et al., 2025).

2.2.4 Regresi

Regresi adalah salah satu teknik fundamental dalam *supervised machine learning* yang bertujuan memodelkan hubungan antara satu variabel dependen (variabel target) dan satu atau lebih variabel independen (variabel prediktor) (Kangalli Uyar et al., 2025). Secara umum, regresi digunakan untuk memprediksi nilai kontinu, yang membedakannya dari klasifikasi yang digunakan untuk memperkirakan kategori diskrit. Dalam penelitian ini, upaya memprediksi Indeks Standar Pencemar Udara (ISPU) disebut sebagai masalah regresi karena ISPU merupakan variabel terikat yang berupa nilai-nilai kontinu dalam suatu rentang tertentu.

Tujuan utama dari analisis regresi adalah mengestimasi hubungan antara variabel independen dan variabel dependen (Chyan et al., 2024). Dalam konteks *machine learning*, fokus regresi adalah meminimalkan kesalahan prediksi antara nilai aktual dan nilai hasil prediksi model (Yosrita & Idham, 2025). Secara matematis, tujuan regresi adalah menemukan fungsi (f) yang paling mendekati hubungan antara set fitur polutan (x) dan nilai ISPU (y):

$$\hat{y} \approx f(x) \tag{2.1}$$

Di mana ŷ adalah nilai prediksi ISPU, dan tujuannya adalah meminimalkan kesalahan: |ŷ_i - y_i|.

Regresi sering digunakan dalam studi prediksi lingkungan karena kemampuannya memperkirakan nilai kompleks seperti konsentrasi polutan ($PM_{2.5}$, CO) atau indeks kualitas udara (ISPU) berdasarkan berbagai faktor seperti polutan dan kondisi meteorologi (Alfasanah et al., 2025).

2.2.5 *Ensemble Learning*

Ensemble learning adalah metode dalam *machine learning* yang bertujuan meningkatkan akurasi, stabilitas, dan ketangguhan model prediksi dengan menggabungkan hasil dari beberapa model dasar (*base learners*) menjadi satu model yang lebih kuat (*strong learner*). Prinsip dasar metode ini adalah bahwa keputusan kolektif dari sekelompok model dapat menghasilkan prediksi yang lebih akurat dibandingkan model tunggal terbaik (Chyan et al., 2024).

Ensemble learning diklasifikasikan menjadi dua kategori utama:

1. *Bagging* (*Bootstrap Aggregating*)
   Bagging melatih setiap *base learner* secara paralel menggunakan subset data yang diambil secara acak dengan pengembalian (*bootstrap sampling*). Tujuan utamanya adalah mengurangi varians model dan mencegah *overfitting*. Contoh metode ini adalah *Random Forest*, di mana sejumlah *decision tree* dibangun berdasarkan subset data dan fitur yang berbeda,

kemudian hasil prediksinya digabungkan melalui *averaging* (Yu et al., 2025).

2. *Boosting*

   Boosting melatih setiap *base learner* secara sekuensial, di mana setiap model baru berfokus memperbaiki kesalahan (*residual error*) dari model sebelumnya. Pendekatan ini mengurangi bias dengan memberikan bobot lebih besar pada data yang sebelumnya salah diprediksi. *Extreme Gradient Boosting* (*XGBoost*) merupakan implementasi lanjutan dari teknik ini yang dikenal karena efisiensi dan akurasi tinggi dalam berbagai tugas regresi (Syahreza et al., 2024).

Keunggulan *ensemble learning* meliputi: (1) kemampuan menyeimbangkan bias dan varians untuk akurasi lebih tinggi, (2) stabilitas terhadap perubahan data dan generalisasi yang lebih baik, dan (3) efektivitas dalam menangkap hubungan kompleks dan non-linear pada data kualitas udara (Alfasanah et al., 2025). Metode *ensemble* seperti *Random Forest* dan *XGBoost* telah banyak diterapkan dalam prediksi kualitas udara karena kemampuannya menangani data yang kompleks, mengandung *noise*, dan bersifat multivariat.

2.2.6 *Random Forest Regressor*

Random Forest Regressor adalah jenis algoritma *ensemble learning* yang khusus digunakan untuk menyelesaikan masalah regresi, yaitu memprediksi nilai-nilai yang bersifat kontinu, seperti Indeks Standar Pencemar Udara (ISPU). Algoritma ini dikembangkan dengan metode *Bagging* (*Bootstrap Aggregating*), yang bekerja dengan menggabungkan hasil prediksi dari banyak pohon keputusan (*decision tree*) yang dilatih secara mandiri agar hasil prediksi menjadi lebih tepat dan konsisten (Chyan et al., 2024). RFR menggabungkan hasil dari banyak *Decision Tree* yang dilatih pada subset data yang berbeda untuk menurunkan varians dan meningkatkan generalisasi model. Algoritma ini dikembangkan dari teknik dasar *data mining*, yaitu *Decision Tree*, di mana input dimasukkan pada bagian atas (*root*) pohon dan diproses melalui serangkaian percabangan hingga mencapai bagian bawah (*leaf*) untuk menentukan nilai prediksi (Jatnika et al., 2025). Secara umum, semakin

banyak pohon yang digunakan dalam *forest*, semakin tinggi akurasi yang dapat dicapai oleh model (Jatnika et al., 2025). *Random Forest Regressor* termasuk dalam metode *Supervised Learning* yang dapat digunakan baik untuk tugas klasifikasi maupun regresi (Chyan et al., 2024).

Berbeda dengan *Random Forest* untuk klasifikasi yang menggunakan *voting*, *Random Forest Regressor* menggunakan rata-rata (*averaging*) dari seluruh prediksi pohon untuk menentukan nilai akhir (Santoso, 2022). *Random Forest Regressor* bekerja berdasarkan metode *Bagging* dengan menciptakan keragaman antar model melalui tiga mekanisme utama:

1. *Bootstrap Aggregating* (*Bagging*)
   Setiap pohon keputusan dilatih menggunakan sebagian data pelatihan yang dipilih secara acak dengan pengembalian dari dataset asli (Chyan et al., 2024) . Metode pengambilan sampel *bootstrap* ini memastikan setiap pohon dilatih dengan data yang sedikit berbeda, sehingga mengurangi hubungan antar pohon dan meningkatkan kemampuan model dalam menangani data baru.
2. *Feature Randomness* (Pengacakan Fitur)
   Ketika setiap pohon dibuat, algoritma hanya memperhatikan sebagian dari fitur-fitur yang dipilih secara acak dari semua fitur yang ada, untuk menentukan pembagian terbaik pada setiap node (Syahreza et al., 2024). Cara memilih fitur secara acak ini membuat setiap pohon tidak sama dan membantu mengurangi kemungkinan *overfitting*.
3. *Aggregation* (Penggabungan Prediksi)
   Dalam tugas regresi, hasil akhir didapatkan dengan menghitung rata-rata dari semua prediksi yang diberikan oleh setiap pohon individu. Secara matematis, prediksi akhir dari *Random Forest Regressor* dapat diformulasikan sebagai berikut (Fawaz et al., 2025):

$$\hat{y}RF = \frac{1}{B}\sum_{b=1}^{B} h_b(x) \tag{2.2}$$

   Keterangan:

- $\hat{y}RF(x)$: nilai prediksi akhir *Random Forest* untuk input x
- B : jumlah total pohon dalam forest (*n_estimators*)
- $h_b(x)$: hasil prediksi dari pohon ke-b

Prediksi akhir didapatkan dengan cara menghitung rata-rata aritmatika dari B pohon, sehingga mengurangi varians dan menghasilkan prediksi yang lebih stabil dan akurat. Dalam proses pembangunan setiap pohon (*splitting*), *Random Forest Regressor* mencari titik potong (*threshold*) terbaik dengan meminimalkan nilai Total *Loss* yang diukur menggunakan kriteria *Sum of Squared Errors* (SSE). Berdasarkan teori yang dipaparkan oleh (Montesinos López et al., 2022), pemilihan *threshold* dilakukan dengan meminimalkan rumus berikut:

$$SSE = \sum_{i \in R_1} \left(y_i - \widehat{y_{R_1}}\right)^2 + \sum_{i \in R_2} \left(y_i - \widehat{y_{R_2}}\right)^2 \quad (2.3)$$

Keterangan:

- R_1, R_2: Node hasil pembagian (kiri dan kanan)
- y_i: Nilai aktual target (ISPU)
- $\hat{y}R_1$, $\hat{y}R_2$: Rata-rata nilai target pada masing-masing node

Kinerja *Regressor Random Forest* sangat bergantung pada pengaturan beberapa hyperparameter utama:

- *n_estimators*: Jumlah total pohon keputusan yang dibangun dalam *forest.* Semakin banyak pohon yang dibuat, biasanya semakin baik hasilnya, tetapi akan memakan waktu komputasi yang lebih lama (Chyan et al., 2024).
- *max_depth*: Kedalaman maksimum yang diperbolehkan untuk setiap pohon. Parameter ini mengatur tingkat kesulitan model dan membantu mencegah masalah *overfitting* (Yenkikar et al., 2025).

- *min_samples_split*: Jumlah sampel minimal yang dibutuhkan untuk membagi *node* internal. Parameter ini menentukan seberapa rinci pohon dapat membedakan data (Yenkikar et al., 2025).
- *max_features*: Jumlah fitur terbanyak yang dipilih secara acak oleh setiap pohon saat mencari bagian terbaik untuk membagi data. Parameter ini membantu membuat setiap pohon memiliki perbedaan yang lebih besar (Yu et al., 2025).

Random Forest Regressor memiliki beberapa kelebihan yang membuatnya sering dipakai dalam tugas prediksi. Pertama, algoritma ini cukup kuat dalam menghindari kelebihan sesuaian (*overfitting*) dan mampu beradaptasi dengan berbagai jenis data karena menggabungkan beberapa pohon keputusan (*bagging*) dan memakai variasi fitur (*feature randomization*) (Yenkikar et al., 2025). Kedua, *Random Forest Regressor* dikenal memiliki tingkat keakuratan yang tinggi dan mampu menghadapi hubungan non-linear yang kompleks antar variabel menjadikannya cocok untuk analisis lingkungan seperti memprediksi kualitas udara (Yu et al., 2025). Ketiga, algoritma ini bisa digunakan untuk data dengan banyak variabel dan memberikan gambaran tentang variabel mana yang paling berpengaruh dalam model.

Meskipun memiliki banyak kelebihan, *Random Forest Regressor* juga memiliki beberapa kekurangan. Pertama, model ini memiliki interpretabilitas yang rendah (*low interpretability*) karena terdiri dari ratusan hingga ribuan pohon keputusan yang digabungkan, sehingga struktur internalnya terasa rumit dan sulit dijelaskan dengan mudah. Model ini sering disebut sebagai *black box* yang tidak begitu transparan (Yenkikar et al., 2025). Kedua, algoritma ini memerlukan sumber daya komputasi yang besar dan memakan waktu lebih lama untuk dilatih, terutama ketika jumlah pohon dan ukuran data semakin besar (Syahreza et al., 2024).

2.2.7 *Extreme Gradient Boosting* (*XGBoost*) *Regressor*

Extreme Gradient Boosting (*XGBoost*) merupakan implementasi lanjutan, cepat, dan terdistribusi dari algoritma *Gradient Boosting Decision Tree* (GBDT), yang termasuk salah satu teknik *ensemble learning* paling unggul dalam bidang

machine learning. *XGBoost* dikembangkan oleh Tianqi Chen dan Carlos Guestrin, dan kini telah menjadi pilihan utama dalam berbagai kompetisi *data science* karena mampu memberikan hasil prediksi yang baik, menggunakan sumber daya komputer secara efisien, serta bisa menangani berbagai jenis data (Sabri et al., 2025).

XGBoost termasuk dalam kategori algoritma *boosting*, yaitu metode pembelajaran yang bekerja secara sekuensial dan aditif. Konsep dasar *boosting* adalah membangun model secara bertahap, di mana setiap model baru biasanya berupa pohon keputusan (*decision tree*) ditambahkan untuk memperbaiki kesalahan (*residual error*) yang dihasilkan (Yosrita & Idham, 2025). Dengan kata lain, *XGBoost* dapat dipandang sebagai versi yang telah disempurnakan dari algoritma *Gradient Boosting* tradisional, di mana setiap pohon baru dilatih untuk memprediksi residu dari prediksi kumulatif sebelumnya, sehingga secara progresif mampu mengurangi nilai *loss* dan menghasilkan model dengan akurasi yang lebih tinggi serta generalisasi yang lebih baik.

XGBoost membangun model secara aditif dan iteratif, di mana setiap pohon keputusan baru berfungsi memperbaiki kesalahan dari model sebelumnya. Proses ini berlanjut hingga jumlah maksimum pohon tercapai atau hingga fungsi objektif mencapai nilai minimum (XGBoost Developers, 2026).Secara matematis, fungsi prediksi model *XGBoost* dapat ditulis sebagai:

$$\widehat{y_i} = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in F \tag{2.4}$$

Keterangan:

- $\widehat{y_i}$: nilai prediksi untuk data ke-i
- f_k: pohon keputusan ke-k
- F: himpunan semua fungsi pohon keputusan
- K: jumlah total pohon dalam mode

Tujuan utama *XGBoost* adalah meminimalkan fungsi objektif (*objective function*) yang menggabungkan *loss function* dan *regularization term*:

$$\text{Obj}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2.5}$$

Keterangan:

- $l(y_i, \hat{y}_i)$: *loss function* yang mengukur selisih antara nilai aktual (y_i) dan prediksi ($\hat{y}_i$)
- $\Omega(f_k)$: fungsi regularisasi yang mengontrol kompleksitas pohon

Fungsi regularisasi dituliskan sebagai:

$$\Omega(f_k) = \gamma T_k + \frac{1}{2}\lambda \sum_{j=1}^{T_k} w_{kj}^2 \tag{2.6}$$

Keterangan:

- T_k: jumlah daun pada pohon ke-k
- w_{kj}: bobot pada daun ke-j dari pohon ke-k
- w_k: bobot pada setiap daun
- γ: penalti terhadap jumlah daun
- λ: parameter regularisasi L2

Proses pembelajaran aditif *XGBoost* dilakukan dengan memperbarui prediksi pada setiap iterasi:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{2.7}$$

Keterangan:

- $\hat{y}_i^{(t)}$: nilai prediksi terbaru untuk data ke-*i* pada iterasi ke-*t*
- $\hat{y}_i^{(t-1)}$: nilai prediksi sebelumnya (hasil iterasi sebelumnya, t−1)
- $f_t(x_i)$: fungsi pohon keputusan (*decision tree*) yang dibangun pada iterasi ke-t, digunakan untuk memperbaiki kesalahan prediksi (*residual error*) dari model sebelumnya.

Untuk optimasi yang lebih efisien, *XGBoost* menggunakan ekspansi Taylor orde kedua pada *loss function*, memanfaatkan gradien pertama dan kedua (*Hessian*).

$$\text{Obj}(t) \approx \sum_{i=1}^{n} \left[l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \tag{2.8}$$

di mana gradien pertama (g_i) dan gradien kedua atau *Hessian* (h_i) didefinisikan sebagai:

$$g_i = \frac{\partial\, l\left(y_i, \hat{y}_i^{(t-1)}\right)}{\partial\, \hat{y}_i^{(t-1)}} \tag{2.9}$$

$$h_i = \frac{\partial^2\, l\left(y_i, \hat{y}_i^{(t-1)}\right)}{\partial\, (\hat{y}_i)^{(t-1)2}} \tag{2.10}$$

Keterangan:

- g_i: gradien pertama dari fungsi *loss* terhadap nilai prediksi sebelumnya, yang menunjukkan arah dan besarnya kesalahan prediksi.
- h_i: gradien kedua (*Hessian*) dari fungsi *loss*, yang menunjukkan tingkat kelengkungan fungsi *loss* dan membantu dalam konvergensi yang lebih cepat

Setelah menghilangkan konstanta yang tidak relevan dengan optimasi, fungsi objektif pada iterasi ke-*t* menjadi:

$$\text{Obj}(t) \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \tag{2.11}$$

Pendekatan ini mempercepat konvergensi dan mengontrol *bias-variance trade-off* dengan lebih baik dibandingkan *Gradient Boosting* tradisional. Salah satu keunggulan penting dari definisi ini adalah nilai fungsi objektif hanya bergantung pada $g_i$dan h_i, yang memungkinkan *XGBoost* mendukung fungsi loss kustom dengan cara yang sama.

Untuk mempermudah perhitungan, struktur pohon dapat didefinisikan sebagai:

$$f_t(x) = w_{q(x)}, \quad w \in R^T, \quad q: R^d \rightarrow \{1,2,\dots,T\}$$

di mana w adalah vektor skor pada daun, q adalah fungsi yang memetakan setiap titik data ke daun yang sesuai, dan T adalah jumlah daun. Dengan mendefinisikan $I_j = \{i \mid q(x_i) = j\}$ sebagai himpunan indeks data yang ditetapkan

ke daun ke-j, dan $G_j = \sum_{i \in I_j} g_i$ serta $H_j = \sum_{i \in I_j} h_i$, fungsi objektif dapat ditulis ulang sebagai:

$$\text{Obj}(t) = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2 \right] + \gamma T \tag{2.12}$$

Untuk struktur pohon $q(x)$ yang diberikan, bobot optimal pada setiap daun dan nilai objektif optimal dapat dihitung sebagai:

$$w_j^* = -\frac{G_j}{H_j + \lambda} \tag{2.13}$$

$$\text{obj}^* = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \tag{2.14}$$

Persamaan (2.13) mengukur seberapa baik struktur pohon $q(x)$, mirip dengan ukuran *impurity* dalam pohon keputusan, namun dengan mempertimbangkan kompleksitas model melalui regularisasi.

Untuk menemukan struktur pohon terbaik, *XGBoost* menggunakan algoritma *greedy* yang mencoba membagi satu daun menjadi dua daun. *Gain* yang diperoleh dari pemisahan dihitung sebagai:

$$\text{Gain} = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \tag{2.15}$$

Keterangan:

- G_L, H_L: total gradien dan *Hessian* pada daun kiri setelah pemisahan
- G_R, H_R : total gradien dan *Hessian* pada daun kanan setelah pemisahan
- γ: parameter regularisasi untuk penalti penambahan daun baru

Jika gain kurang dari γ, maka pemisahan tidak dilakukan. Ini merupakan mekanisme *pruning* alami yang mencegah *overfitting*.

Perbedaan utama *XGBoost* dibandingkan *Gradient Boosting* tradisional terletak pada tiga aspek:

1. Regularisasi Eksplisit

XGBoost menambahkan penalti L2 (*Ridge*) untuk mengendalikan kompleksitas pohon melalui parameter λ dan γ, sedangkan GBDT tradisional tidak memiliki mekanisme regularisasi eksplisit dalam formulasi dasarnya.

2. Optimisasi Komputasi

   XGBoost menggunakan pendekatan berbasis *second-order gradient* (gradien dan *Hessian*), sedangkan GBDT hanya menggunakan gradien pertama. Hal ini membuat proses konvergensi lebih cepat dan stabil.

3. Efisiensi Eksekusi

   XGBoost mengoptimalkan penggunaan memori dan memungkinkan paralelisasi pembentukan pohon secara efektif, menjadikannya lebih efisien pada *dataset* besar (Uyar et al., 2025).

Kinerja *XGBoost* sangat bergantung pada pengaturan beberapa *hyperparameter* utama:

- *n_estimators*: Jumlah total pohon dalam model yang mempengaruhi kompleksitas dan kemampuan pembelajaran.
- *max_depth*: Kedalaman maksimum setiap pohon yang memengaruhi kompleksitas model dan risiko *overfitting*.
- *learning_rate* (η): Tingkat pembelajaran yang mengontrol kontribusi setiap pohon. Nilai kecil meningkatkan stabilitas tetapi memerlukan lebih banyak iterasi.
- *subsample*: Rasio data pelatihan yang digunakan untuk setiap pohon, membantu mencegah *overfitting*.
- *colsample_bytree*: Rasio fitur yang digunakan secara acak pada setiap pohon untuk meningkatkan keragaman model.
- *lambda* (λ): Parameter regularisasi L2 yang membantu mengontrol *overfitting* dengan memberikan penalti pada bobot daun yang besar.
- *gamma* (γ): Parameter regularisasi yang memberikan penalti pada penambahan daun baru (Sabri et al., 2025).

XGBoost Regressor memiliki beberapa keunggulan signifikan dalam tugas prediksi. Pertama, algoritma ini menawarkan akurasi prediktif yang tinggi dan kemampuan luar biasa dalam menangani hubungan non-linear antar variabel, bahkan mampu mengungguli algoritma *machine learning* lainnya (Uyar et al., 2025). Kedua, *robustness* model ditingkatkan melalui penggunaan regularisasi L1 dan L2 secara eksplisit, yang merupakan mekanisme efektif untuk menghindari *overfitting* (Uyar et al., 2025).Ketiga, *XGBoost* efisien secara komputasi karena mendukung pemrosesan paralel dan memiliki fitur bawaan untuk menangani *missing value* secara otomatis. Keempat, meskipun secara fundamental adalah model *black box*, *XGBoost* mampu menyediakan informasi *feature importance* yang detail, yang berkontribusi pada interpretasi model (Uyar et al., 2025).

XGBoost juga memiliki beberapa keterbatasan yang perlu dipertimbangkan. Pertama, kompleksitas *hyperparameter tuning* yang memakan waktu karena banyaknya parameter yang harus disesuaikan untuk mencapai kinerja optimal (Uyar et al., 2025).Kedua, sebagai model *ensemble* yang kompleks, algoritma ini dapat membutuhkan sumber daya memori dan CPU yang tinggi ketika diterapkan pada *dataset* yang sangat besar. Ketiga, meskipun didukung oleh alat interpretasi modern, model berbasis pohon ini secara inheren lebih sulit diinterpretasikan daripada model linear sederhana (Yu et al., 2025).

Sebagai algoritma regresi yang kuat, *XGBoost Regressor* telah banyak digunakan dalam bidang prediksi lingkungan. Dalam konteks kualitas udara, algoritma ini mampu memprediksi nilai Indeks Standar Pencemar Udara (ISPU) dan konsentrasi polutan seperti $PM_{2.5}$ dengan tingkat akurasi tinggi (Alfasanah et al., 2025).Keunggulan *XGBoost* dalam menangani data non-linear, heterogen, dan berskala besar menjadikannya pilihan yang ideal untuk memodelkan hubungan kompleks antara konsentrasi polutan dan nilai ISPU dalam penelitian prediksi kualitas udara.

2.2.8 Metrik Evaluasi Model Regresi

Dalam penelitian prediksi regresi, metrik evaluasi berperan penting untuk mengukur seberapa dekat nilai prediksi ($\hat{y}$) dengan nilai target aktual (y). Penggunaan

beberapa metrik secara bersamaan memberikan gambaran yang lebih komprehensif mengenai akurasi, bias, dan kekuatan eksplanatori model (Syahreza et al., 2024).Tiga metrik utama yang digunakan untuk mengevaluasi kinerja model regresi dalam penelitian ini adalah *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), dan Koefisien Determinasi (R^2). Metrik-metrik ini sangat penting untuk menentukan model mana (*Random Forest* atau *XGBoost*) yang paling unggul dalam memodelkan Indeks Standar Pencemar Udara (ISPU).

1. *Root Mean Squared Error* (RMSE)

   Root Mean Squared Error (RMSE) adalah metrik yang mengukur rata-rata besar kesalahan antara nilai yang diprediksi oleh model dan nilai target aktual, dengan memberikan penalti yang lebih besar pada kesalahan yang besar (*outliers*). Metrik ini dihitung dengan mengambil rata-rata kuadrat dari seluruh kesalahan. Nilai RMSE dihitung menggunakan rumus berikut (Uyar et al., 2025):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.16}$$

   Keterangan:

   - n: jumlah data observasi
   - y_i: nilai aktual ISPU ke-i
   - ŷ_i : nilai prediksi ISPU ke-i

   Nilai RMSE berada dalam satuan yang sama dengan variabel target (ISPU), sehingga mudah diinterpretasikan. Karena adanya proses kuadrat, kesalahan prediksi yang besar akan memperbesar nilai RMSE secara signifikan, menjadikan metrik ini sensitif terhadap keberadaan data ekstrem. Model dianggap lebih baik jika menghasilkan nilai RMSE yang lebih kecil, yang menunjukkan prediksi model lebih mendekati nilai aktual.

2. *Mean Absolute Error* (MAE)

   Mean Absolute Error (MAE) mengukur rata-rata magnitudo kesalahan antara nilai prediksi dan nilai aktual, dihitung sebagai nilai absolut. Berbeda dengan RMSE, MAE tidak memberikan penalti yang lebih besar pada kesalahan yang ekstrem. Hal ini menghasilkan gambaran yang lebih linear dan mudah diinterpretasikan mengenai rata-rata kesalahan prediksi model (Uyar et al., 2025a).Nilai MAE dihitung dengan rumus:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.17}$$

   Keterangan:

   - n: jumlah data observasi
   - y_i : nilai aktual ISPU ke-i
   - ŷ_i : nilai prediksi ISPU ke-i
   - |...| : nilai absolut

   Sama seperti RMSE, nilai MAE juga berada dalam satuan yang sama dengan variabel target. Metrik ini sangat berguna untuk memahami besarnya kesalahan rata-rata yang dapat diharapkan dari model. Secara umum, MAE lebih stabil terhadap *outliers* dibandingkan RMSE. Nilai MAE yang lebih kecil menunjukkan kinerja model yang lebih baik dan prediksi yang lebih dekat ke nilai aktual.

3. Koefisien Determinasi (R^2)

   Koefisien Determinasi (R^2) atau *R-squared* adalah metrik statistik yang merepresentasikan proporsi variasi dalam variabel dependen (ISPU) yang dapat dijelaskan oleh variabel independen (polutan) dalam model regresi. Dengan kata lain, R^2 mengukur seberapa baik model secara keseluruhan *fit* dengan data. Nilai R^2 berkisar antara 0 hingga 1 (atau 0% hingga 100%). Nilai yang mendekati 1 menunjukkan bahwa model dapat menjelaskan sebagian besar variabilitas data output, mengindikasikan *goodness of fit*

model yang tinggi dan bahwa fitur-fitur yang digunakan adalah prediktor yang efektif (Uyar et al., 2025).R^2 dihitung menggunakan rumus:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (2.18)$$

Keterangan:

- y_i : nilai aktual ISPU ke-i
- $\hat{y}_i$: nilai prediksi ISPU ke-i
- $\bar{y}$: nilai rata-rata aktual ISPU
- n : jumlah data observasi

Metrik R^2 merupakan indikator penting untuk mengevaluasi kekuatan eksplanatori model. Model dengan nilai R^2 yang lebih tinggi menunjukkan kemampuan yang lebih baik dalam menjelaskan variabilitas data dan mereplikasi hubungan antara variabel input dan output. Dalam konteks penelitian ini, model dengan kombinasi RMSE dan MAE terendah serta R^2 tertinggi akan dipilih sebagai model terbaik untuk memprediksi nilai ISPU di Jakarta.

2.2.9 CRISP-DM (*Cross-Industry Standard Process for Data Mining*)

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) adalah kerangka kerja metodologi terstruktur yang paling banyak digunakan dalam proyek *data mining* dan *machine learning*. Kerangka ini bersifat siklis dan fleksibel, dirancang untuk memandu penelitian melalui serangkaian tahapan yang logis dan sistematis (Afandi et al., 2024). CRISP-DM terdiri dari enam fase utama yang bersifat iteratif, di mana setiap fase dapat kembali ke fase sebelumnya jika diperlukan penyesuaian (Rifai et al., 2019):

1. *Business Understanding* (Pemahaman Bisnis): Fase ini adalah titik awal dalam proses CRISP-DM yang berfokus pada penetapan tujuan proyek dari perspektif bisnis atau masalah yang ingin dipecahkan (Afandi et al., 2024; Rifai et al., 2019).

2. *Data Understanding* (Pemahaman Data): Setelah tujuan ditetapkan, fase ini melibatkan pengumpulan data awal, eksplorasi mendalam, dan verifikasi kualitas data termasuk deteksi *missing values* dan *outliers* (Siswipraptini et al., 2023).
3. *Data Preparation* (Persiapan Data): Pembersihan data, seleksi fitur, transformasi, dan pembagian data menjadi *training set* dan *testing* (Rifai et al., 2019).
4. *Modeling* (Pemodelan): Pada fase ini, teknik *machine learning* diterapkan untuk membangun model prediksi (Siswipraptini et al., 2023).
5. *Evaluation* (Evaluasi): Penilaian kualitas dan akurasi model menggunakan metrik evaluasi untuk memastikan model mencapai tujuan yang ditetapkan (Afandi et al., 2024).
6. *Deployment* (Penyebaran)
   Integrasi model yang tervalidasi untuk menghasilkan nilai praktis, termasuk dokumentasi dan visualisasi hasil (Siswipraptini et al., 2023).

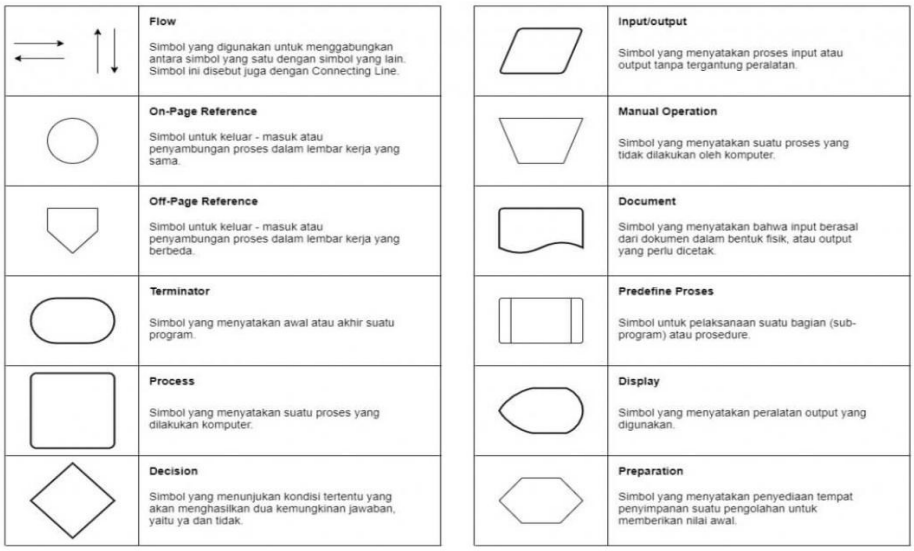
Metodologi CRISP-DM memberikan struktur yang jelas dan sistematis untuk penelitian ini, memastikan setiap tahapan dilakukan secara komprehensif dan dapat direplikasi.

2.2.10 *Python*

Python adalah bahasa pemrograman tingkat tinggi yang bersifat *interpreter*, *object-oriented*, dan *multiplatform* yang dikembangkan oleh Guido van Rossum pada tahun 1991 (Santoso, 2022). *Python* telah menjadi bahasa pemrograman utama dalam bidang *data science*, *artificial intelligence*, dan *machine learning* karena sintaksnya yang sederhana, mudah dipahami, dan memiliki ekosistem pustaka yang luas (Chyan et al., 2024).

Dalam penelitian ini, Python dipilih karena menyediakan pustaka lengkap untuk *machine learning* seperti *Scikit-learn* untuk algoritma *ensemble*, XGBoost untuk *gradient boosting*, *Pandas* dan *NumPy* untuk manipulasi data, serta *Matplotlib* dan *Seaborn* untuk visualisasi (XGBoost Developers, 2026).*Python* memungkinkan pengelolaan seluruh siklus *machine learning* mulai dari persiapan data, pemodelan, evaluasi, hingga visualisasi hasil secara efisien.

2.2.11 Flowchart



Gambar 2. 1 Simbol-simbol Flowchart

Sumber : (Rasyid Ridho, 2024)

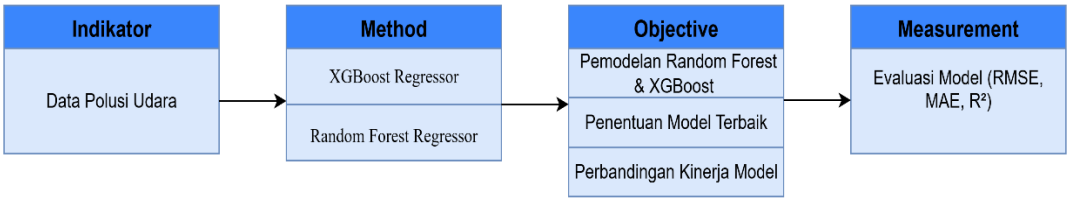
Flowchart adalah alat bantu visual yang digunakan untuk merepresentasikan alur kerja atau proses secara sistematis dalam bentuk diagram (Zhang et al., 2023). *Flowchart* terdiri dari simbol-simbol standar yang saling terhubung dengan garis atau panah untuk menunjukkan urutan langkah-langkah dalam suatu proses.

Dalam penelitian *machine learning*, *flowchart* digunakan untuk menggambarkan langkah-langkah penelitian secara logis dan teratur, mulai dari pemahaman masalah, persiapan data, pemodelan, evaluasi, hingga penerapan hasil (Uyar et al., 2025). Simbol-simbol standar yang digunakan meliputi oval (awal/akhir), persegi panjang (proses), belah ketupat (keputusan), dan jajaran genjang (input/output), sebagaimana ditunjukkan pada Gambar 2.1.

2.3 Kerangka Pemikiran

Kerangka pemikiran adalah struktur konseptual yang digunakan untuk merencanakan, mengatur, dan mengembangkan pemahaman dalam suatu penelitian. Dalam konteks penelitian ini, kerangka pemikiran menggambarkan alur berpikir sistematis dari tahap pengumpulan dan pengolahan data hingga analisis hasil, dengan

tujuan membandingkan kinerja dua algoritma *ensemble learning*, *Random Forest Regressor* dan *Extreme Gradient Boosting* (*XGBoost*) *Regressor* dalam memodelkan Indeks Standar Pencemar Udara (ISPU) di wilayah DKI Jakarta menggunakan pendekatan regresi.



Gambar 2. 2 Kerangka Pemikiran

1. Indikator (Data Input)

   Indikator merupakan parameter yang digunakan sebagai variabel independen (input) yang akan dianalisis dalam proses pemodelan. Dalam penelitian ini, indikator utama berupa data polutan udara digunakan untuk memprediksi nilai Indeks Standar Pencemar Udara (ISPU) sebagai variabel dependen (target). Variabel independen yang digunakan meliputi $PM_{2.5}$, PM_{10}, SO_2, CO, O_3, dan NO_2, serta enam parameter meteorologi sebagai variabel tambahan, yang meliputi radiasi matahari (GRAD), suhu udara (TempAir), kelembapan udara (HumAir), kecepatan angin (FF), arah angin (DD), dan curah hujan (RAIN). Data tersebut diperoleh dari hasil pengukuran kualitas udara di lima Stasiun Pemantauan Kualitas Udara (SPKU) di wilayah DKI Jakarta.

2. Metode

   Metode merupakan langkah atau pendekatan yang sistematis digunakan dalam penelitian untuk memecahkan suatu permasalahan serta mencapai tujuan yang telah ditetapkan. Metode yang digunakan dalam penelitian ini adalah pendekatan *ensemble learning* berbasis regresi, yang bertujuan meningkatkan akurasi dan stabilitas prediksi dengan menggabungkan hasil dari beberapa model dasar. Dua algoritma utama yang dibandingkan adalah:

a. *Random Forest Regressor* (RFR)

   Algoritma berbasis *bagging* yang menggabungkan hasil prediksi dari banyak *decision tree* untuk mengurangi varians dan meningkatkan akurasi prediksi.

b. Extreme Gradient Boosting (*XGBoost*) *Regressor*

   Algoritma berbasis *boosting* yang membangun model secara berurutan, di mana setiap pohon baru berfungsi memperbaiki kesalahan dari model sebelumnya. *XGBoost* juga menerapkan regularisasi untuk mencegah *overfitting* dan mempercepat proses optimasi.

3. Tujuan (*Objective*)

   Tujuan merupakan arah utama yang ingin dicapai dalam penelitian ini sebagai hasil dari penerapan metode yang telah dirancang. Penelitian ini bertujuan untuk memperoleh model prediksi *Indeks Standar Pencemar Udara (ISPU)* yang akurat dengan menggunakan pendekatan regresi berbasis *machine learning*. Melalui pendekatan ini, penelitian berfokus pada perbandingan kinerja dua algoritma *ensemble learning* yang populer, yaitu *Random Forest Regressor* dan *Extreme Gradient Boosting (XGBoost) Regressor*, dalam memodelkan hubungan antara konsentrasi polutan udaradan nilai ISPU di wilayah DKI Jakarta. Selain menghasilkan model prediksi, penelitian ini juga bertujuan untuk menganalisis seberapa baik masing-masing algoritma mampu merepresentasikan pola *non-linear* pada data kualitas udara. Evaluasi dilakukan menggunakan tiga metrik utama, yaitu *Root Mean Squared Error (RMSE)*, *Mean Absolute Error (MAE)*, dan *Koefisien Determinasi (R^2)*, untuk menilai tingkat akurasi dan stabilitas model. Dari hasil analisis perbandingan tersebut, diharapkan dapat diketahui algoritma mana yang memberikan performa terbaik dalam memprediksi nilai ISPU, serta memberikan kontribusi terhadap pengembangan model prediksi kualitas udara yang lebih andal dan efisien di masa depan.

4. Pengukuran (*Measurement*)

   Setelah model dibangun, evaluasi kinerja model regresi dilakukan menggunakan metrik pengukuran kinerja yang telah ditetapkan:

a. *Root Mean Squared Error* (RMSE)

   RMSE digunakan untuk mengukur besar kesalahan kuadrat rata-rata antara nilai aktual dan nilai yang diprediksi. Nilai RMSE yang kecil menunjukkan bahwa model memiliki tingkat kesalahan yang rendah dan prediksi yang lebih akurat.

b. *Mean Absolute Error* (MAE)

   MAE mengukur rata-rata kesalahan absolut antara nilai aktual dan prediksi tanpa memperhatikan arah kesalahan. Metrik ini memberikan gambaran yang lebih stabil terhadap rata-rata kesalahan model dan mudah diinterpretasikan.

c. Koefisien Determinasi (R^2)

   R^2 menunjukkan seberapa besar variasi data aktual yang dapat dijelaskan oleh model. Nilai R^2 yang mendekati 1 menunjukkan bahwa model mampu menjelaskan sebagian besar variabilitas data dan memiliki kecocokan yang baik terhadap data aktual.

Kerangka ini secara keseluruhan menggambarkan alur logis penggunaan data polutan udara, data meteorologi serta penerapan dua metode regresi *ensemble*, hingga penilaian kinerja model dengan metrik yang tepat untuk mencapai tujuan penelitian.