Terbit *online* pada laman : http://teknosi.fti.unand.ac.id/

Jurnal Nasional Teknologi dan Sistem Informasi

| ISSN (Print) 2460-3465 | ISSN (Online) 2476-8812 |

Studi Kasus

Evaluasi Model Prediksi Curah Hujan Berbasis *Machine Learning* di Kota Bandung

Indri Hapsari [a, *], *Shifa Pandya Wisesa* [b]

[a] *Balai Besar Meteorologi Klimatologi dan Geofisika Wilayah II, Jalan H. Abdul Ghani No. 05, Kota Tangerang Selatan, 15412, Indonesia*
[b] *Universitas Amikom Yogyakarta, Jl. Ring Road Utara, Kab. Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 15 Juli 2025
Revisi Akhir: 02 September 2025
Diterbitkan *Online*: 07 September 2025

KATA KUNCI

Model prediksi
Logistic Regression
Random Forest
Extreme Gradient Boosting

KORESPONDENSI

E-mail: indri.hapsari@bmkg.go.id*

A B S T R A C T

Model prediksi curah hujan berbasis *machine learning* bertujuan untuk memberikan prediksi curah hujan harian yang akurat. Penelitian ini membandingkan kinerja beberapa algoritma *machine learning* yang digunakan untuk membangun model prediksi kejadian hujan di Kota Bandung, yaitu *Logistic Regression, Random Forest,* dan *Extreme Gradient Boosting*. Evaluasi model prediksi dilakukan dengan menganalisis nilai *accuracy, precision, recall, F1 score*, dan AUC. Hasilnya menunjukkan bahwa ketiga model tersebut dapat memprediksi kejadian hujan harian di Kota Bandung secara efektif. Namun, model *Random Forest* secara umum menunjukkan performa terbaik dengan akurasi prediksi mencapai 85% sehingga model ini direkomendasikan untuk digunakan dalam memprediksi kejadian hujan guna mendukung pengambilan keputusan dan perencanaan kegiatan di Kota Bandung.

1. PENDAHULUAN

Algoritma *machine learning* berperan penting dalam analisis prediktif dengan menggunakan data historis dan pola data saat ini untuk memprediksi kejadian di masa depan [1]. Dalam konteks prediksi cuaca, metode machine learning dapat digunakan untuk mempelajari pola dan hubungan kompleks antara variabel cuaca yang beragam, seperti suhu, kelembaban, waktu, dan wilayah [2]. Istilah "prediksi curah hujan" mengacu pada proses penggunaan berbagai metode untuk memprediksi jumlah, waktu, dan lokasi presipitasi (hujan, salju, hujan es, dll.) di area tertentu selama periode waktu tertentu [3]. Untuk meningkatkan akurasi prediksi curah hujan, penting untuk mengintegrasikan berbagai sumber data dan memanfaatkan teknik pemodelan tingkat lanjut [4].

Pada penelitian ini, model prediksi yang dibangun berbasis algoritma *machine learning,* di antaranya *Logistic Regression* (LR), *Random Forest* (RF), dan *Extreme Gradient Boosting* (XGB). Berbagai penelitian terdahulu telah menggunakan ketiga algoritma ini dalam melakukan prediksi cuaca dan terbukti keandalan masing-masing model prediksi untuk wilayah yang berbeda. Model prediksi berbasis RF telah digunakan [1] dalam melakukan prediksi cuaca di Jakarta dan digunakan [8] dalam memperkirakan kemungkinan hujan di Kecamatan Purwodadi dengan hasil prediksi masing-masing adalah sangat baik dan cukup akurat. Sementara, curah hujan bulanan di Kabupaten Banyuwangi diprediksi cukup baik menggunakan model prediksi XGB [9]. Metode LR dan RF yang digunakan [10] untuk melakukan prediksi kejadian hujan di India menunjukkan hasil akurasi LR sedikit lebih tinggi dibandingkan RF. Perbandingan kinerja model LR, RF, XGB dan beberapa model prediksi lainnya pada prediksi curah hujan harian di Australia menunjukkan bahwa model RF memiliki performa terbaik dengan tingkat akurasi sebesar 87% [11]. XGB justru menunjukkan kinerja terbaik dibandingkan RF ketika memprediksi curah hujan harian di Kota Bahir Dar, Ethiopia [12].

https://doi.org/10.25077/TEKNOSI.v11i2.2025.136-143

Penelitian ini melakukan prediksi kejadian hujan di Kota Bandung. Karakteristik cuaca Kota Bandung dipengaruhi oleh topografinya yang berupa dataran tinggi dan sering terjadi hujan dengan intensitas yang bervariasi, sehingga prediksi cuaca yang akurat diharapkan dapat bermanfaat untuk pariwisata, operasional transportasi, dan aktivitas ekonomi lainnya. Beberapa penelitian terdahulu mengenai prediksi curah hujan di Kota Bandung menggunakan metode berbasis algoritma *machine learning,* di antaranya, prediksi pola hujan menggunakan data hujan tahun 2017-2021 dengan metode *Long Short Term Memory* (LSTM) menunjukkan hasil yang cukup baik dengan nilai *Train Score* RMSE sebesar 12,24 dan *Test Score* RMSE sebesar 8,86 [5], prediksi cuaca selama 1 tahun ke depan menggunakan data 4 tahun terakhir dengan metode *Naive Bayes* menunjukkan ketepatan prediksi dengan hasil akurasi sebesar 65% [6], dan prediksi hujan menggunakan data tahun 2015-2020 dengan metode *K-Nearest Neighbor* (KNN) menunjukan bahwa hasil terbaik adalah ketika nilai K sebesar 5 dengan hasil akurasi sebesar 86,199% [7].

2. METODE

2.1. Data Penelitian

Data yang digunakan pada penelitian ini adalah data klimatologi harian dari Badan Meteorologi Klimatologi dan Geofisika (BMKG), yaitu hasil observasi Stasiun Geofisika Bandung periode tahun 2019-2024 yang diperoleh dari https://dataonline.bmkg.go.id/. Data klimatologi harian tersebut meliputi data suhu udara rata-rata (TR), suhu udara maksimum (TX), suhu udara minimum (TM), curah hujan (RR), lamanya penyinaran matahari (SS), kelembaban udara rata-rata (RH), kecepatan angin rata-rata (WW), arah angin terbanyak (DD), kecepatan angin maksimum (WWX), dan arah angin maksimum (DDX).

Contoh data klimatologi harian Stasiun Geofisika Bandung dapat dilihat pada Tabel 1.

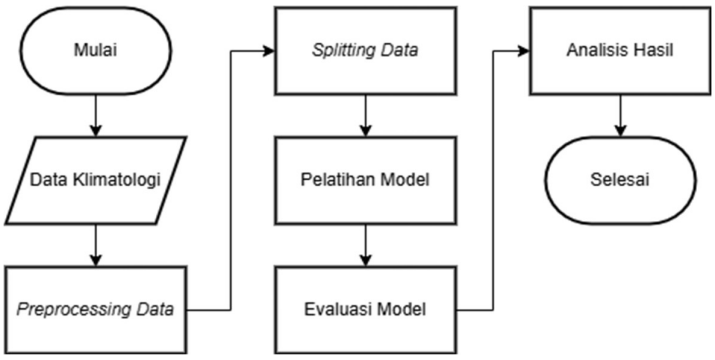
Tabel 1. Data Klimatologi Stasiun Geofisika Bandung

Tanggal	TR	TM	TX	RR	SS	RH	WW	DD	WWX	DDX
01/01/2019	23,0	21,8	28,0	0,0	14,2	79	2	270	3	280
02/01/2019	24,0	20,8	31,0	13,5	15,8	74	4	270	8	250
03/01/2019	24,1	20,7	30,6	0,8	60,0	76	4	270	7	260
04/01/2019	24,2	20,0	30,8	0,0	45,8	69	3	270	5	210
05/01/2019	24,8	20,0	31,7	0,0	70,8	70	3	0	5	300

2.2. Tahapan Penelitian

Penelitian ini mengikuti tahapan sesuai diagram alur penelitian pada Gambar 1. Seluruh tahapan penelitian menggunakan bahasa pemrograman Python yang dilakukan dalam dalam aplikasi berbasis web Jupyter Notebook. Penjelasan dari tiap tahapan penelitian adalah sebagai berikut:

1. Data klimatologi harian dikumpulkan melalui website BMKG. Selanjutnya, data disimpan dalam dalam format CSV untuk diolah lebih lanjut.
2. *Preprocessing data,* meliputi *data cleaning* (pembersihan data dan pengisian data kosong) dan menentukan variabel data yang memiliki korelasi kuat dengan curah hujan. Pembersihan data kosong dilakukan antara lain dengan mengganti data hujan yang bernilai 8888 (hujan tak terukur) menjadi 0 dan mengganti seluruh data yang bernilai -9999 (tidak ada data) menjadi data kosong. Masing-masing data kosong ini kemudian akan diisi menggunakan nilai rata-rata bulanannya. Selanjutnya, dilakukan perhitungan nilai korelasi dari tiap variabel data klimatologi dengan curah hujan untuk menentukan variabel apa saja yang akan digunakan untuk membangun model prediksi curah hujan. Pada tahap ini pula dilakukan penambahan data kejadian hujan esok hari pada data klimatologi harian.
3. *Splitting data*, dilakukan dengan membagi data menjadi data latih sebesar 80% dan data uji sebesar 20% yang dipilih secara random menggunakan *resampling* [13]. Pada tahap ini juga diterapkan *oversampling* menggunakan metode SMOTE *(Synthetic Minority Oversampling Technique)* pada data latih untuk mengatasi ketidakseimbangan jumlah data pada tiap kelas [14]. Kelebihan dari metode SMOTE



Gambar 1. Diagram Alur Penelitian

secara umum adalah tidak menyebabkan adanya informasi yang hilang, menghindari terjadinya *overfitting*, membangun wilayah keputusan yang lebih besar, serta mampu meningkatkan akurasi prediksi kelas minoritas[15].

4. Pelatihan model berbasis algoritma *Logistic Regression*, *Random Forest*, dan *Extreme Gradient Boosting* menggunakan bahasa pemrograman Python. Model dikembangkan melalui beberapa iterasi untuk mencapai performa yang optimal.
5. Evaluasi kinerja model dilakukan untuk mengetahui tingkat akurasi dan efektivitas model, yaitu menggunakan metrik evaluasi dan visualisasi grafik ROC (*Receiver Operating Characteristic)*.
6. Analisis hasil dilakukan dengan membandingkan nilai evaluasi tiap model menentukan performa model prediksi terbaik.

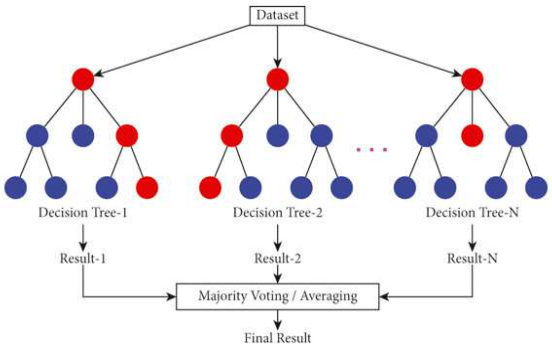
2.3. *Algoritma Machine Learning*

Model prediksi yang dibangun pada penelitian ini menggunakan tiga jenis algoritma *machine learning*, yaitu *Logistic Regression*, *Random Forest*, dan *Extreme Gradient Boosting*. Ketiga algoritma ini digunakan untuk memodelkan kemungkinan terjadinya hujan berdasarkan data cuaca historis.

Logistic Regression **(LR)** adalah salah satu algoritma yang dapat digunakan untuk menghasilkan output bernilai diskrit. LR biasa digunakan untuk mengatasi masalah dalam klasifikasi. Keuntungan dari regresi logistik adalah implementasi metode yang sederhana dan hasil akurasinya cenderung baik [16]. Dalam klimatologi, Persamaan umumnya adalah sebagai berikut [16]:

$$log\left[\frac{y}{1-y}\right] = y = b_0 + b_1x_1 + \ldots + b_nx_n \quad (1)$$

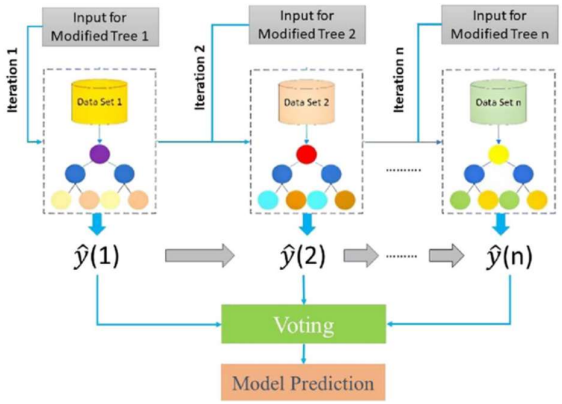
Dimana, n adalah banyaknya variabel bebas;
y adalah nilai prediksi dari variabel bebas x;
$x_1,\ldots,x_n$ adalah variabel bebas;
b_0 adalah konstanta;
$b_1,\ldots,b_n$ adalah koefisien regresi.



Gambar 2. Arsitektur *Random Forest*
(Sumber: Khan *et.al*, 2021)

Random Forest **(RF)** adalah algoritma yang membangun kumpulan pohon keputusan *(decision trees)* untuk membuat prediksi akurat dan stabil. RF sangat cocok untuk dataset yang kompleks karena mampu menangani hubungan non-linear antar variabel serta meminimalkan *overfitting* (kondisi ketika model machine learning terlalu cocok dengan data latih, sehingga tidak dapat memprediksi data baru dengan akurat). Hasil prediksi yang dihasilkan adalah berdasarkan voting mayoritas untuk klasifikasi atau rata-rata untuk regresi [8]. Ilustrasi arsitektur Random Forest [17] seperti yang ditampilkan pada Gambar 2.

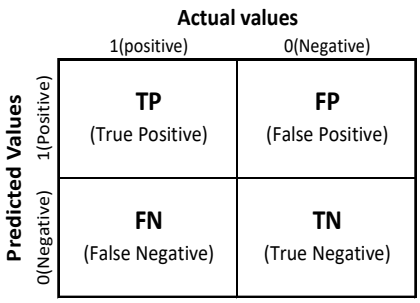
Extreme Gradient Boosting **(XGB)** adalah pengembangan dari algoritma *gradient boosting* yang menggunakan perkiraan lebih akurat untuk menemukan model *decision tree* terbaik. Algoritma XGB meningkatkan akurasi model secara keseluruhan dengan mencegah masalah *overfitting* dalam proses pelatihannya. XGB banyak digunakan untuk berbagai masalah regresi dan klasifikasi karena kecepatan dan keakuratan prediksinya [12]. Ilustrasi mekanisme XGB [18] dapat dilihat pada Gambar 3.



Gambar 3. Mekanisme *Extreme Gradient Boosting*
(Sumber: Faska *et.al*, 2023)

2.4. *Evaluasi Kinerja Model*

Evaluasi performa model yang dilakukan untuk melihat tingkat akurasi serta mengetahui seberapa baik kinerja model pada penelitian ini adalah menggunakan metrik evaluasi dan visualisasi grafik ROC *(Receiver Operating Characteristic)*. Penentuan nilai-nilai dalam metrik evaluasi dan grafik ROC adalah melalui perhitungan menggunakan hasil dari model prediksi berbasis *Logistic Regression*, *Random Forest*, dan *Extreme Gradient Boosting* yang ditampilkan dalam bentuk *Confusion Matrix* (Gambar 4).



Gambar 4. *Confusion Matrix*

Dalam *Confusion Matrix* terdapat TP *(True Positive)*, TN *(True Negative)*, FN *(False Negative),* dan FP *(False Positive)* yang masing-masing menggambarkan banyaknya kondisi saat prediksi maupun nilai aktualnya positif, prediksi maupun nilai aktualnya

https://doi.org/10.25077/TEKNOSI.v11i2.2025.136-143

negatif, nilai prediksi negatif tetapi nilai aktualnya positif, dan nilai prediksi positif tapi nilai aktualnya negatif.

Nilai-nilai dalam metrik evaluasi diperoleh melalui perhitungan menggunakan nilai-nilai dalam *Confusion Matrix*, yaitu meliputi [1]:

1. *Accuracy* (akurasi).
   Nilai *accuracy* menunjukkan seberapa baik model membuat prediksi yang benar dari total prediksi yang dilakukan. Persamaan untuk menghitung nilai *accuracy* adalah sebagai berikut:
   $$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

2. *Precision* (presisi).
   Nilai *precision* menunjukkan seberapa baik model membuat prediksi yang benar untuk kelas positif dari total prediksi positif yang dilakukan. Perhitungan nilai *precision* adalah melalui persamaan sebagai berikut:
   $$Precision = \frac{TP}{TP+FP} \quad (3)$$

3. *Recall* (sensitivitas).
   Nilai *recall* menggambarkan seberapa baik suatu model dalam mengidentifikasi kelas positif dengan benar, cara menghitungnya adalah melalui persamaan berikut:
   $$Recall = \frac{TP}{TP+FN} = TPR \quad (4)$$

4. *F1 score*.
   Nilai *F1 score* merupakan rata-rata harmonis dari *precision* dan *recall* yang memberikan indikasi keseluruhan kinerja model dalam memprediksi kelas berbeda. Persamaan yang digunakan untuk menghitung *F1 score* adalah sebagai berikut:
   $$F1\ score = \frac{2\times(Recall\times Precision)}{Recall+Precision} \quad (5)$$

Selain menggunakan metrik evaluasi, pada tahap evaluasi kinerja model digunakan pula grafik ROC sebagai visualisasi dari performa tiap model prediksi. Kurva ROC mengekspresikan *Confusion Matrix* [19], yaitu memberikan gambaran tentang hubungan antara TPR (*True Positive Rate*) dan FPR (*False Positive Rate*) saat ambang batas klasifikasi bervariasi. TPR (dikenal juga sebagai *recall*) menunjukkan berapa banyak data positif yang diprediksi dengan benar sebagai positif, sementara FPR menunjukkan seberapa sering model salah memprediksi kelas negatif sebagai positif. Formula untuk menghitung TPR dapat dilihat pada persamaan (4), sementara persamaan untuk menghitung FPR adalah sebagai berikut:

$$FPR = \frac{FP}{FP+TN} \quad (6)$$

Kinerja model dinilai semakin baik bila kurva ROC cenderung menjulur ke arah sudut kiri atas. Model prediksi terbaik dapat diketahui dengan membandingkan nilai AUC *(Area Under the Curve)* dari masing-masing kurva ROC. Nilai AUC menyatakan persentase luasan area dibawah kurva ROC yang menunjukkan seberapa baik model bisa membedakan antara kelas positif dan negatif. Nilai AUC makin mendekati 1 menunjukkan kinerja model yang lebih baik, yaitu model memiliki tingkat akurasi yang tinggi dari hasil prediksinya [20].

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data

Data klimatologi harian yang berhasil dikumpulkan adalah data hasil observasi Stasiun Geofisika Bandung periode tahun 2019-2024 dalam format CSV yang meliputi data suhu udara rata-rata (TR), suhu udara maksimum (TX), suhu udara minimum (TM), curah hujan (RR), lamanya penyinaran matahari (SS), kelembaban udara rata-rata (RH), kecepatan angin rata-rata (WW), arah angin terbanyak (DD), kecepatan angin maksimum (WWX), dan arah angin maksimum (DDX). Informasi data tersebut seperti ditampilkan pada Gambar 5.

3.2. Preprocessing Data

3.2.1. Pengisian Data Kosong

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2130 entries, 0 to 2129
Data columns (total 11 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Tanggal  2130 non-null   datetime64[ns]
 1   TR       2116 non-null   float64
 2   TM       2087 non-null   float64
 3   TX       2115 non-null   float64
 4   RR       1908 non-null   float64
 5   SS       2130 non-null   float64
 6   RH       2115 non-null   float64
 7   WW       2123 non-null   float64
 8   DD       2123 non-null   float64
 9   WWX      2123 non-null   float64
 10  DDX      2123 non-null   float64
dtypes: datetime64[ns](1), float64(10)
```

Gambar 5. Informasi Data

Hasil pendeteksian data kosong menunjukkan bahwa terdapat data kosong pada variabel suhu udara rata-rata, suhu udara minimum, suhu udara maksimum, dan kelembaban udara rata-rata, yaitu masing-masing sebanyak 4, 27, 7, dan 5 data. Data-data kosong dari masing-masing variabel ini diisi menggunakan rata-rata bulanannya. *Souce code* pengisian data kosong ditunjukkan pada Gambar 6.

3.2.2. Nilai Korelasi Variabel Iklim

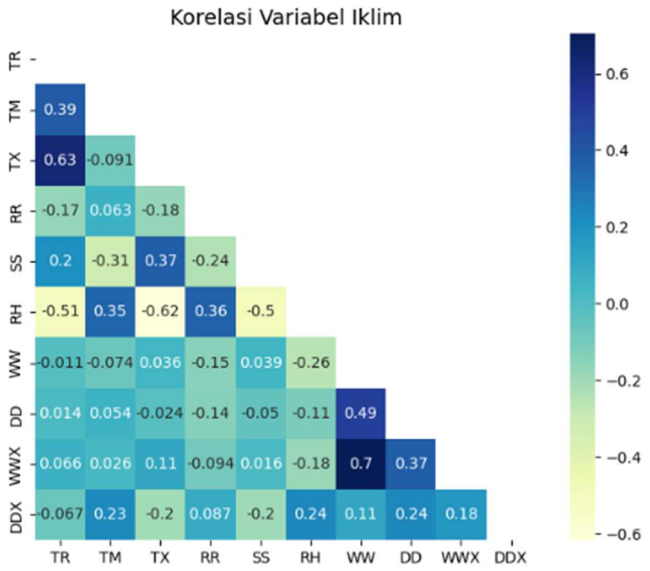
Hasil perhitungan nilai korelasi curah hujan dengan variabel iklim lainnya ditampilkan pada Gambar 7. Variabel iklim yang

```
# mengisi value yang kosong dengan rata rata bulan nya
df["Bulan"] = df["Tanggal"].dt.month
# Isi nilai null berdasarkan mean pada bulan yang sama
kolom_target = ["TM", "TX", "TR", "RH"]

for kolom in kolom_target:
    # Menghitung mean berdasarkan bulan
    mean_per_bulan = df.groupby("Bulan")[kolom].transform("mean")
    # Mengisi nilai null dengan mean bulan terkait
    df[kolom] = df[kolom].fillna(mean_per_bulan)
```

Gambar 6. *Source Code* Pengisian Data Kosong

diputuskan untuk digunakan dalam prediksi curah hujan adalah yang memiliki nilai korelasi di atas 0,1, yaitu antara lain suhu udara rata-rata (TR), suhu udara maksimum (TX), lamanya penyinaran matahari (SS), kelembaban udara rata-rata (RH), kecepatan angin rata-rata (WW), dan arah angin terbanyak (DD). Variabel yang berkorelasi paling tinggi terhadap curah hujan adalah RH dengan nilai korelasi sebesar 0,36. Sementara, suhu udara minimum (TM), arah angin maksimum (WWX), dan kecepatan angin maksimum (DDX) tidak digunakan dalam membangun model prediksi curah hujan.



Gambar 7. Korelasi Variabel Iklim

```
# membuat kolom besok terjadi hujan atau tidak
df['Besok_hujan'] = df.shift(-1)['Hujan']
df = df.iloc[:-1,:].copy()
df.round(1).head()
```

	Tanggal	TR	TM	TX	RR	SS	RH	WW	DD	WWX	DDX	Bulan	Hujan	Besok_hujan
0	2019-01-01	23.0	21.8	28.0	0.0	14.2	79.0	2.0	270.0	3.0	280.0	1	No	Yes
1	2019-01-02	24.0	20.8	31.0	13.5	15.8	74.0	4.0	270.0	8.0	250.0	1	Yes	Yes
2	2019-01-03	24.1	20.7	30.6	0.8	60.0	76.0	4.0	270.0	7.0	260.0	1	Yes	No
3	2019-01-04	24.2	20.0	30.8	0.0	45.8	69.0	3.0	270.0	5.0	210.0	1	No	No
4	2019-01-05	24.8	20.0	31.7	0.0	70.8	70.0	3.0	0.0	5.0	300.0	1	No	No

Gambar 8. *Source Code* Penambahan Data

3.2.3. Penambahan Data

Sebelum dilanjutkan ke tahap *splitting data,* dilakukan penambahan data kejadian hujan esok hari berdasarkan data hujan yang dimiliki. *Source code* dan hasil penambahan kolom data kejadian hujan esok hari adalah seperti pada Gambar 8.

3.3. Splitting Data

Langkah pertama dalam *splitting data* adalah membagi data menjadi variabel input (X) dan variabel target (y). Variabel inputnya berupa data suhu udara rata-rata (TR), suhu udara maksimum (TX), lamanya penyinaran matahari (SS), kelembaban udara rata-rata (RH), kecepatan angin rata-rata (WW), dan arah angin terbanyak (DD), sementara variabel targetnya berupa data kejadian hujan esok hari (Besok_hujan). Selanjutnya, dilakukan pembagian data menjadi data latih sebesar 80% dan data uji sebesar 20% menggunakan *resampling,* kemudian diterapkan metode SMOTE pada data latih. *Source code splitting data* dan penerapan metode SMOTE ditunjukkan pada gambar 9.

3.4. Pelatihan Model Prediksi

Pelatihan model berbasis algoritma *Logistic Regression*, *Random Forest*, dan *Extreme Gradient Boosting* ditunjukkan dalam *source code* dalam Gambar 10. Hasil prediksi kejadian curah hujan harian di Kota Bandung menggunakan model prediksi berbasis algoritma *Logistic Regression, Random Forest*, dan *Extreme Gradient Boosting,* selanjutnya digunakan untuk mendapatkan nilai TP, TN, FP, dan FN masing-masing model

```
# split data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
from imblearn.over_sampling import SMOTE
smote = SMOTE(random_state=2)  # Inisialisasi SMOTE

# Terapkan SMOTE hanya pada data training
X_resampled, y_resampled = smote.fit_resample(X_train, y_train)
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size=0.2, random_state=42)
```

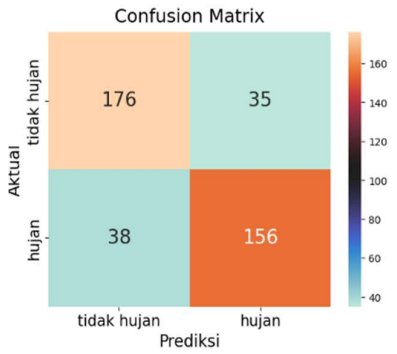
Gambar 9. *Source Code Splitting Data* dan SMOTE

https://doi.org/10.25077/TEKNOSI.v11i2.2025.136-143

```
from sklearn.metrics import roc_curve, roc_auc_score, f1_score
def data_training_(X_train, X_test, y_train, y_test):
    models = []
    models.append(('LR', LogisticRegression(random_state=0)))
    models.append(('RF', RandomForestClassifier(n_estimators=100, random_state=42)))
    models.append(('XGB', XGBClassifier(
        learning_rate=0.23, max_delta_step=5,
        objective='reg:logistic', n_estimators=92,
        max_depth=5, eval_metric="logloss", gamma=3, base_score=0.5)))
    res_cols = ["Model", "Accuracy", "Precision", "Recall", "F1-score"]
    df_result_ = pd.DataFrame(columns=res_cols)
    index = 0
    for name, model in models:
        model.fit(X_train, y_train)
        y_pred = model.predict(X_test)
        score = accuracy_score(y_test, y_pred)
        precision = precision_score(y_test, y_pred)
        recall = recall_score(y_test, y_pred)
        f1_ = f1_score(y_test, y_pred)
        idx_res_values = [name, score, precision, recall, f1_]
        df_result_.loc[index, res_cols] = idx_res_values
        index += 1
    df_result_ = df_result_.sort_values("Accuracy", ascending=False).reset_index(drop=True)
    return df_result_
```

Gambar 10. *Source Code* Model Prediksi

prediksi yang hasilnya dapat digambarkan dalam bentuk *confusion matrix*. Contoh *confusion matrix* dapat dilihat pada Gambar 11 yang merupakan *confusion matrix* dari hasil prediksi menggunakan algoritma *Logistic Regression*. Nilai dalam *confusion matrix* masing-masing model prediksi selanjutnya digunakan sebagai input dalam penentuan nilai evaluasi, yaitu untuk menentukan nilai *accuracy, precision*, *recall*, dan *F1 score* tiap model prediksi.



Gambar 11. *Confusion Matrix* Model *Logistic Regression*

3.5. Evaluasi Model Prediksi

Hasil perhitungan nilai *accuracy, precision*, *recall*, dan *F1 score* tiap model prediksi dapat dilihat pada Tabel 2 yang menunjukkan hasil evalusi model prediksi *Logistic Regression* (LR), *Random Forest* (RF), dan *Extreme Gradient Boosting* (XGB). Berdasarkan nilai hasil evaluasi pada Tabel 2, secara umum ketiga model prediksi dapat dikatakan memiliki performa yang baik dalam memprediksi data, dapat dilihat dari nilai *accuracy, precision*, *recall*, dan *F1 score* ketiga model yang lebih besar dari 80%, kecuali untuk nilai *recall* model prediksi XGB yang bernilai 78%.

Pada Tabel 2 dapat dilihat pula bahwa model prediksi berbasis algoritma Random Forest secara umum memiliki nilai hasil evaluasi yang lebih tinggi dibandingkan model prediksi lainnya untuk nilai *accuracy, precision,* dan *F1 score*, yaitu masing-masing sebesar 85%, 89%, dan 85%. Sementara nilai *recall* tertinggi dimiliki oleh model prediksi *Logistic Regression* sebesar 83%, hanya berbeda tipis dengan model prediksi *Random Forest* sebesar 82%. Hal ini menunjukkan bahwa model prediksi *Random Forest* memiliki kinerja terbaik di antara ketiga model dalam membuat prediksi yang benar dari total prediksi yang dilakukan, memprediksi data yang benar untuk kelas positif, dan memprediksi data pada kelas yang berbeda. Sementara, model prediksi *Logistic Regression* memiliki kinerja terbaik dalam mengidentifikasi kelas positif dengan benar dibandingkan model prediksi lainnya.

Tabel 2. Hasil Evaluasi Model Prediksi

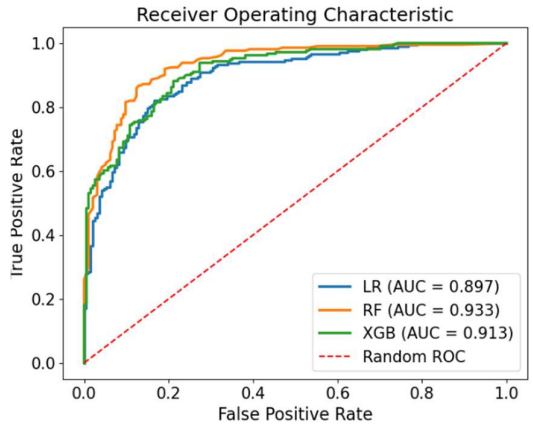
Hasil Evaluasi	Model		
	LR (%)	RF (%)	XGB (%)
Accuracy	82	85	81
Precision	82	89	84
Recall	83	82	78
F1 Score	83	85	81

Performa model prediksi *Logistic Regression* terlihat lebih baik dibandingkan model prediksi *Extreme Gradient Boosting* untuk hasil evaluasi model pada Tabel 2 berupa nilai *accuracy, recall*, dan *F1 score.* Secara umum, model prediksi *Extreme Gradient Boosting* memiliki performa terendah dibandingkan model prediksi lainnya berdasarkan hasil evaluasi pada Tabel 2, kecuali untuk nilai *precision* yang masih lebih tinggi dibandingkan model prediksi *Logistic Regression*.

Selain menggunakan perbandingan nilai *accuracy, precision*, *recall*, dan *F1 score*, tingkat kinerja model prediksi *Logistic Regression* (LR), *Random Forest* (RF), dan *Extreme Gradient Boosting* (XGB) dapat pula dibandingkan menggunakan grafik ROC *(Receiver Operating Characteristic)* yang ditampilkan pada Gambar 12. Pada grafik tersebut terdapat kurva ROC dan nilai AUC *(Area Under the Curve)* masing-masing model prediksi

untuk menjelaskan tingkat kinerja model prediksi dalam membedakan antara kelas positif dan negatif pada berbagai ambang batas.

Kurva ROC dari ketiga model prediksi pada Gambar 5 terlihat cenderung menjulur ke arah sudut kiri atas dan nilai AUC masing-masing model prediksi adalah mendekati 1. Hal ini mengindikasikan bahwa ketiga model prediksi mampu membedakan antar kelas dengan baik. Berdasarkan nilai AUC, dapat dikatakan bahwa model prediksi *Random Forest* dan *Extreme Gradient Boosting* memiliki kinerja yang sangat baik karena memiliki nilai AUC >0,9, yaitu masing-masing sebesar 0,933 dan 0,913, sementara model prediksi *Logistic Regression* dinilai memiliki kinerja yang baik dengan nilai AUC sebesar 0,897. Performa terbaik dalam membedakan antar kelas di antara ketiga model berdasarkan grafik ROC tersebut adalah model prediksi *Random Forest* yang memiliki nilai AUC tertinggi dibandingkan model prediksi lainnya, sementara kinerja terendah dimiliki oleh model prediksi *Logistic Regression* yang memiliki nilai AUC terendah.



Gambar 12. *Grafik ROC*

4. KESIMPULAN

Implementasi algoritma *Logistic Regression, Random Forest,* dan *Extreme Gradient Boosting* pada model prediksi kejadian hujan di Kota Bandung mengindikasikan bahwa ketiga model prediksi mampu melakukan prediksi dengan baik sehingga ketiga model dapat dipertimbangkan untuk digunakan sebagai model prediksi. Performa terbaik di antara ketiga model berdasarkan nilai *accuracy, precision*, *recall, F1 score*, dan AUC, secara umum ditunjukkan oleh model prediksi *Random Forest* dengan nilai masing-masing sebesar 85%, 89%, 82%, 85%, dan 0,933 dengan catatan nilai *recall* yang berbeda tipis dari metode *Logistic Regression* sebesar 83%.

DAFTAR PUSTAKA

[1] Z. A. Dwiyanti and C. Prianto, "Prediksi Cuaca Kota Jakarta Menggunakan Metode Random Forest," *J. Tekno Insentif*, vol. 17, no. 2, pp. 127–137, 2023, doi: 10.36787/jti.v17i2.1136.

[2] A. Sharma and V. Vijayakumar, "on Cloud Systems EAI Endorsed Transactions Predictive Analytics in Weather Forecasting using Machine Learning Algorithms," pp. 1–4, 2019, doi: 10.4108/eai.7-12-2018.159405.

[3] S. D. Latif et al., "Assessing Rainfall Prediction Models: Exploring the Advantages of Machine Learning and Remote Sensing Approaches," *Alexandria Eng. J.*, vol. 82, no. May, pp. 16–25, 2023, doi: 10.1016/j.aej.2023.09.060.

[4] O. A. Wani et al., "Predicting Rainfall Using Machine Learning, Deep Learning, and Time Series Models Across an Altitudinal Gradient an The North-Western Himalayas," *Sci. Rep.*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-77687-x.

[5] R. Farikhul Firdaus and I. V. Paputungan, "Prediksi Curah Hujan di Kota Bandung Menggunakan Metode Long Short Term Memory," *J. Penelit. Inov.*, vol. 2, no. 3, pp. 453–460, 2022, doi: 10.54082/jupin.99.

[6] A. N. Sallamah, A. Sunandar, A. M. Rizka, and C. Rozikin, "Prakiraan Cuaca Kota Bandung di Tahun Berikutnya dengan Menggunakan Metode Naïve Bayes," *DoubleClick: Journal of Computer and Information Technology*, vol. 8, no. 1, pp. 27–32, 2024. http://e-journal.unipma.ac.id/index.php/doubleclick

[7] D. M. Nanda, T. H. Pudjiantoro, and P. N. Sabrina, "Metode K-Nearest Neighbor (KNN) dalam Memprediksi Curah Hujan di Kota Bandung," *Semin. Nas. Tek. Elektro, Sist. Informasi, dan Tek. Inform.*, pp. 387–393, 2022, doi: 10.31284/p.snestik.2022.2750

[8] R. Torhino and P. N. Andono, "Penerapan Algoritma Random Forest dalam Prediksi Curah Hujan untuk Mendukung Analisis Cuaca," *Building of Informatics Technology and Science (BITS),* vol. 6, no. 3, PP. 1688-1699, 2024, doi: 10.47065/bits.v6i3.6404.

[9] A. Fauziah, H. Hermanto, and M. A. Sukmarini, "*Extreme Gradient Boosting* pada Peramalan Pola Curah Hujan Bulanan Kabupaten Banyuwangi*," J. Kridatama Sains Dan Teknol.*, vol. 6, no. 02, pp. 430–440, 2024, doi: 10.53863/kst.v6i02.1154.

[10] S. M. Gowtham, Y. S. Ganesh, and M. M. Ali, "Efficient Rainfall Prediction and Analysis using Machine Learning Techniques," *Turkish J. Comput. Math. Educ.* vol. 12, no. 6, pp. 3467–3474, 2021.

[11] S. R. Vinta and R. Peeriga, "Rainfall Prediction using XGB Model with the Australian Dataset," *EAI Endorsed Trans. Energy Web*, vol. 11, no. April, pp. 1–4, 2024, doi: 10.4108/ew.5386.

[12] C. M. Liyew and H. A. Melese, "Machine Learning Techniques to Predict Daily Rainfall Amount," *J. Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00545-4.

[13] H. N. Irmanda, Ermatita, M. K. Awang, and M. Adrezo, "Enhancing Weather Prediction Models through the Application of Random Forest Method and Chi-Square Feature Selection," *Int. J. Informatics Vis.*, vol. 8, no. 3–2, pp. 1506–1514, 2024, doi: 10.62527/joiv.8.3.2356.

[14] R. Ridwan, E. H. Hermaliani, and M. Ernawati, "Penerapan: Penerapan Metode SMOTE untuk Mengatasi Imbalanced Data pada Klasifikasi Ujaran Kebencian," *Comput. Sci.*, vol. 4, no. 1, pp. 80–88, 2024, doi: 10.31294/coscience.v4i1.

https://doi.org/10.25077/TEKNOSI.v11i2.2025.136-143

[15] N. P. Y. T. Wijayanti, E. N. Kencana, and I. W. Sumarjaya, "Smote: Potensi dan Kekurangannya pada Survei," *E-Jurnal Mat.*, vol. 10, no. 4, p. 235, 2021, doi: 10.24843/mtk.2021.v10.i04.p348.

[16] R. Praveena, T. R. G. Babu, M. Birunda, G. Sudha, P. Sukumar, and J. Gnanasoundharam, "Prediction of Rainfall Analysis Using Logistic Regression and Support Vector Machine*," J. Phys. Conf. Ser.,* vol. 2466, no. 1, 2023, doi: 10.1088/1742-6596/2466/1/012032.

[17] M. Y. Khan, A. Qayoom, M. S. Nizami, M. S. Siddiqui, S. Wasi, and S. M. K. Raazi, "Automated Prediction of Good Dictionary EXamples (GDEX): A Comprehensive Experiment with Distant Supervision, Machine Learning, and Word Embedding-Based Deep Learning Techniques," vol. 2021, 2021, doi: 10.1155/2021/2553199.

[18] Z. Faska, L. Khrissi, K. Haddouch, and N. El Akkad, "A Robust and Consistent Stack Generalized Ensemble - Learning Framework for Image Segmentation," *J. Eng. Appl. Sci.*, vol. July, 2023, doi: 10.1186/s44147-023-00226-4.

[19] D. T. Wilujeng, M. Fatekurohman, and I. M. Tirta, "Analisis Risiko Kredit Perbankan Menggunakan Algoritma K-Nearest Neighbor dan Nearest Weighted K-Nearest Neighbor," Indones. J. Appl. Stat., vol. 5, no. 2, p. 142, 2023, doi: 10.13057/ijas.v5i2.58426.

[20] N. A. P. Indaryono, R. Saedudin, and F. Hamami, "Analisa Perbandingan Algoritma Random Forest dan Naïve Bayes untuk Klasifikasi Curah Hujan Berdasarkan Iklim di Indonesia", *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.,* vol. 9, no. 1, pp. 158–167, 2024, doi: 10.29100/jipi.v9i1.4421.

BIODATA PENULIS

Indri Hapsari

Penulis adalah pegawai di Balai besar Meteorologi Klimatologi Wilayah II Tangerang Selatan. Lahir di Bekasi pada tanggal 25 September 1982. Penulis memiliki latar belakang pendidikan S1-Matematika (Universitas Gadjah Mada) dan S2-Klimatologi Terapan (Institut Pertanian Bogor).

Shifa Pandya Wisesa

Penulis lahir di Purworejo pada tanggal 2 Mei 2004. Saat ini sedang menempuh pendidikan S1 di Universitas Amikom Yogyakarta.