



## Analisis Kinerja Algoritma *K-Nearest Neighbors* (KNN) dan *Random Forest* untuk Klasifikasi Kondisi Cuaca

Agim Sahrija Asha Yuda<sup>1</sup>, Muhammad Desfriyan Arif Rosady<sup>2</sup>, Nabil Ibrahim Faisal<sup>3</sup>, Edi Ismanto<sup>4</sup>

Email: <sup>1</sup>230401320@student.umri.ac.id, <sup>2</sup>230401262@student.umri.ac.id, <sup>3</sup>230401297@student.umri.ac.id,

<sup>4</sup>edi.ismanto@umri.ac.id

<sup>1,2,3,4</sup>Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau

Diterima: 27 Juli 2025 | Direvisi: 06 Agustus 2025 | Disetujui: 30 Agustus 2025

©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,  
Universitas Muhammadiyah Riau, Indonesia

### Abstrak

Perkembangan teknologi informasi telah mendorong pemanfaatan algoritma pembelajaran mesin dalam berbagai bidang, termasuk dalam analisis dan prediksi kondisi cuaca. Penelitian ini bertujuan untuk menganalisis dan membandingkan kinerja dua algoritma machine learning, yaitu *K-Nearest Neighbors* (KNN) dan *Random Forest*, dalam klasifikasi kondisi cuaca berdasarkan data meteorologis historis. Dataset yang digunakan mencakup fitur-fitur seperti curah hujan, suhu maksimum, suhu minimum, dan kecepatan angin, dengan kategori target berupa jenis cuaca seperti hujan, cerah, kabut, gerimis, dan salju. Proses meliputi pra-pemrosesan data, penskalaan fitur, pembagian data latih dan uji, serta pelatihan model menggunakan pustaka *scikit-learn*. Evaluasi kinerja dilakukan menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil menunjukkan bahwa model *Random Forest* memiliki akurasi lebih tinggi (82%) dibandingkan KNN (78%), dengan performa yang lebih stabil pada kelas mayoritas. Namun, kedua model mengalami penurunan kinerja signifikan pada kelas minoritas akibat ketidakseimbangan data. Studi ini merekomendasikan optimasi lebih lanjut seperti penyeimbangan kelas dan pemilihan parameter model untuk meningkatkan akurasi klasifikasi cuaca secara keseluruhan.

**Kata kunci:** *K-Nearest Neighbors*, *Random Forest*, *Klasifikasi Cuaca*, *Pembelajaran Mesin*, *Evaluasi Kinerja*.

### *Performance Analysis of K-Nearest Neighbors (KNN) and Random Forest Algorithms for Classification of Weather Conditions*

#### Abstract

The development of information technology has encouraged the use of machine learning algorithms in various fields, including in the analysis and prediction of weather conditions. This study aims to analyze and compare the performance of two machine learning algorithms, namely *K-Nearest Neighbors* (KNN) and *Random Forest*, in the classification of weather conditions based on historical meteorological data. The dataset used includes features such as rainfall, maximum temperature, minimum temperature, and wind speed, with target categories in the form of weather types such as rain, sunny, fog, drizzle, and snow. The process includes data pre-processing, feature scaling, training and test data sharing, and model training using the *scikit-learn* library. Performance evaluations are conducted using accuracy, precision, recall, and F1-score metrics. The results showed that the *Random Forest* model had higher accuracy (82%) than KNN (78%), with more stable performance in the majority class. However, both models experienced significant performance declines in minority classes due to data imbalances. The study recommends further optimizations such as class balancing and model parameter selection to improve the overall accuracy of weather classification.

**Keywords:** *K-Nearest Neighbors*, *Random Forest*, *Weather Classification*, *Machine Learning*, *Performance Evaluation*.

## 1. PENDAHULUAN

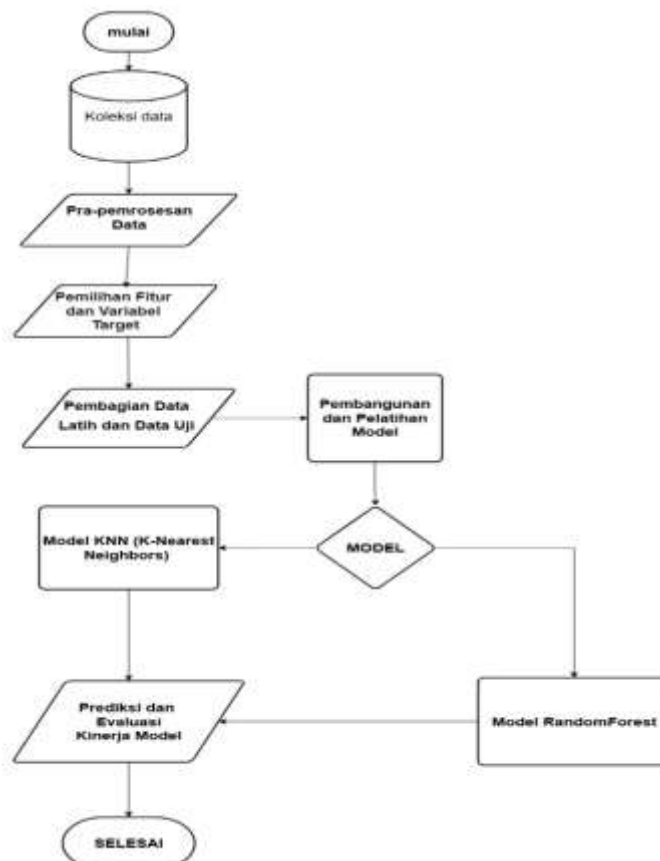
Prediksi cuaca merupakan fenomena penting yang analisisnya semakin berkembang melalui penerapan berbagai algoritma data mining[1]. Pemanfaatan teknik data mining telah menjadi pendekatan umum untuk memprediksi keadaan cuaca berdasarkan data historis[2]. Dalam konteks ini, beragam algoritma machine learning terus dievaluasi kinerjanya guna menemukan metode yang paling efektif untuk tugas peramalan cuaca[3].

Berbagai penelitian telah menerapkan algoritma klasifikasi untuk prediksi cuaca. Salah satu algoritma yang populer adalah *K-Nearest Neighbors* (KNN), yang telah terbukti dapat digunakan untuk memprediksi data cuaca[4]. Fleksibilitas KNN juga telah ditunjukkan dalam berbagai domain klasifikasi lain, mulai dari analisis sentimen di media sosial, klasifikasi kasus hukum, hingga penentuan lokasi layanan kependudukan[5][6][7]. Di sisi lain, algoritma *Random Forest* juga menunjukkan kinerja yang kuat dan telah berhasil diimplementasikan sebagai model machine learning untuk peramalan cuaca[8]. Analisis menggunakan *Random Forest* juga telah divalidasi sebagai metode yang efektif dalam konteks peramalan cuaca[9]. Studi komparatif antar algoritma merupakan pendekatan yang umum, seperti perbandingan antara KNN dan *Naïve Bayes*, untuk menentukan metode mana yang paling unggul pada suatu kasus[10]. Meskipun algoritma KNN dan *Random Forest* telah terbukti berhasil dalam tugas prediksi cuaca secara terpisah, pemilihan algoritma yang paling optimal sering kali sangat bergantung pada karakteristik dataset yang digunakan. Kinerja sebuah model klasifikasi seperti KNN juga dapat sangat dipengaruhi oleh faktor-faktor seperti proses seleksi fitur, yang menegaskan pentingnya analisis mendalam terhadap implementasi algoritma pada data spesifik.

Oleh karena itu, diperlukan sebuah studi komparatif yang secara langsung mengevaluasi dan membandingkan kinerja kedua algoritma ini pada dataset cuaca yang sama untuk memberikan bukti empiris mengenai keunggulan serta kelemahan masing-masing metode. Berdasarkan latar belakang dan permasalahan yang ada, penelitian ini bertujuan untuk melakukan analisis kinerja komparatif antara algoritma *K-Nearest Neighbors* (KNN) dan *Random Forest* untuk tugas klasifikasi kondisi cuaca. Secara spesifik, penelitian ini bertujuan untuk menjawab pertanyaan: "Manakah di antara algoritma *K-Nearest Neighbors* dan *Random Forest* yang memberikan tingkat akurasi lebih tinggi dalam mengklasifikasikan kondisi cuaca berdasarkan dataset yang digunakan?"

## 2. METODE PENELITIAN

Metodologi penelitian ini mengadopsi pendekatan kuantitatif dengan berfokus pada implementasi dan evaluasi algoritma *K-Nearest Neighbors* (KNN) untuk klasifikasi kondisi cuaca. Tahapan-tahapan penelitian dirinci sebagai berikut:



Gambar 1. Struktur keseluruhan metode yang digunakan

## 2.1. Dataset

Dataset yang digunakan dalam penelitian ini adalah data cuaca historis yang bersumber dari file `seattle_weather.csv`. Dataset ini mengandung beberapa atribut meteorologis yang akan digunakan sebagai fitur, yaitu *precipitation* (jumlah curah hujan), *temp\_max* (suhu maksimum harian), *temp\_min* (suhu minimum harian), dan *wind* (kecepatan angin). Kolom target yang mendeskripsikan kondisi cuaca adalah *weather*, yang merujuk pada kategori kondisi cuaca seperti *'drizzle'*, *'fog'*, *'rain'*, *'snow'*, atau *'sun'*. Pemilihan dataset ini didasarkan pada ketersediaan data yang relevan untuk tujuan klasifikasi kondisi cuaca.

## 2.2. Pra-pemrosesan Data

Tahap pra-pemrosesan data esensial untuk memastikan kualitas dan format data yang sesuai untuk pelatihan model pembelajaran mesin. Dalam tahap ini, dataset dibaca menggunakan pustaka `pandas` di Python. Langkah-langkah pra-pemrosesan yang dilakukan adalah sebagai berikut:

1. Penanganan Nilai Hilang: Pemeriksaan nilai hilang dilakukan pada semua kolom fitur dan kolom target. Dari hasil pemeriksaan, tidak ditemukan adanya nilai hilang di dataset yang digunakan, sehingga tidak diperlukan imputasi atau penghapusan baris berdasarkan kondisi ini.
2. Penskalaan Fitur Numerik: Mengingat sensitivitas algoritma KNN terhadap skala fitur, penskalaan fitur numerik (*precipitation*, *temp\_max*, *temp\_min*, *wind*) adalah langkah krusial. Teknik `StandardScaler` digunakan untuk menormalisasi fitur-fitur ini, mengubahnya sehingga memiliki rata-rata nol dan variansi satu. Penskalaan ini diintegrasikan ke dalam pipeline model untuk memastikan konsistensi dalam tahap pelatihan dan pengujian.

## 2.3. Pemilihan Fitur dan Variabel Target

Untuk membangun model klasifikasi, diperlukan identifikasi fitur-fitur independen (variabel prediktor) dan variabel target (variabel yang akan diklasifikasikan). Dalam penelitian ini, fitur-fitur yang dipilih dari dataset adalah:

1. *precipitation*: Merepresentasikan jumlah curah hujan.
2. *temp\_max*: Merepresentasikan suhu maksimum harian.
3. *temp\_min*: Merepresentasikan suhu minimum harian.
4. *wind*: Merepresentasikan kecepatan angin.

Variabel target yang akan diklasifikasikan oleh model adalah *weather*, yang merujuk pada kategori kondisi cuaca seperti *'sun'*, *'rain'*, *'fog'*, *'drizzle'*, atau *'snow'*.

---

### Kode Program

```
# Memilih kolom-kolom yang akan digunakan sebagai fitur (variabel independen)
X = df[['precipitation', 'temp_max', 'temp_min', 'wind']]
# Memilih kolom 'weather' sebagai variabel target (variabel dependen)
y = df['weather']
```

---

## 2.4. Pembagian Data Latih dan Data Uji

Agar model yang dibangun dapat dievaluasi secara objektif terhadap data yang belum pernah dilihat sebelumnya, dataset dibagi menjadi dua subset: data latih (training data) dan data uji (testing data). Data latih digunakan untuk melatih model, sementara data uji digunakan untuk mengukur kinerja model setelah pelatihan. Dalam penelitian ini, 80% dari dataset dialokasikan untuk data latih dan 20% sisanya untuk data uji. Parameter `random_state=42` digunakan untuk memastikan pembagian data yang konsisten dan dapat direproduksi pada setiap eksekusi.

---

### Kode Program

```
# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

---

## 2.5. Pembangunan dan Pelatihan Model *RandomForest* dan *K-Nearest Neighbors*

Pembangunan dan pelatihan model *random forest* dimulai dengan pemilihan fitur prediktor dan target. Data kemudian dibagi menjadi dua data latih yakni 80:20 menggunakan fungsi `train_test_split`. Model *Random Forest* dibentuk menggunakan *RandomForestClassifier*, yang secara otomatis membangun sejumlah pohon keputusan (*decision trees*) dari subset data latih yang berbeda-beda, lalu menggabungkan hasil voting dari masing-masing pohon untuk membuat prediksi akhir.

Lalu, model klasifikasi *K-Nearest Neighbors* dibangun menggunakan kelas *KNeighborsClassifier* dari pustaka scikit-learn. Untuk penelitian ini, jumlah tetangga terdekat ( $k$ ) diatur menjadi 5 ( $n\_neighbors=5$ ). Pemilihan nilai  $k$  ini merupakan parameter penting yang dapat memengaruhi kinerja model. Sebelum pelatihan model, fitur numerik diproses menggunakan *StandardScaler* dalam sebuah Pipeline yang juga mencakup klasifier K-NN. Ini memastikan bahwa penskalaan fitur diterapkan secara konsisten pada data latih dan data uji. Setelah inisialisasi dan persiapan pipeline, model dilatih dengan menggunakan data latih ( $X\_train$  dan  $y\_train$ ), memungkinkan algoritma untuk mempelajari pola dari fitur-fitur untuk memprediksi kondisi cuaca.

---

Kode Program

```
# 4. Buat dan latih model KNN
model = KNeighborsClassifier(n_neighbors=5)
model.fit(X_train, y_train)
```

---

### 2.6 Prediksi dan Evaluasi Kinerja Model

Setelah model KNN dan *RandomForest* berhasil dilatih, langkah selanjutnya adalah melakukan prediksi pada data uji ( $X\_test$ ) untuk menguji kemampuan generalisasi model. Hasil prediksi ( $y\_pred$ ) kemudian dibandingkan dengan nilai aktual ( $y\_test$ ) untuk mengevaluasi kinerja model. Dua metrik evaluasi utama digunakan (Metrik Evaluasi, Tahun):

- Akurasi (*Accuracy Score*): Mengukur proporsi instans yang diklasifikasikan dengan benar dari total instans di data uji
- Laporan Klasifikasi (*Classification Report*): Memberikan gambaran kinerja yang lebih rinci untuk setiap kelas, termasuk *precision*, *recall*, dan *f1-score*, serta jumlah support untuk setiap kelas

---

Kode Program

```
# 6. Evaluasi akurasi
akurasi = accuracy_score(y_test, y_pred)
print("Akurasi:", akurasi)

# Tambahan: laporan klasifikasi
print(classification_report(y_test, y_pred))
```

---

## 3. HASIL DAN PEMBAHASAN

### 3.1. Hasil pelatihan model menggunakan metode K-NN

Berdasarkan tahapan metodologi yang telah dijelaskan, model *K-Nearest Neighbors* berhasil diterapkan untuk klasifikasi kondisi cuaca. Hasil evaluasi kinerja model adalah sebagai berikut:

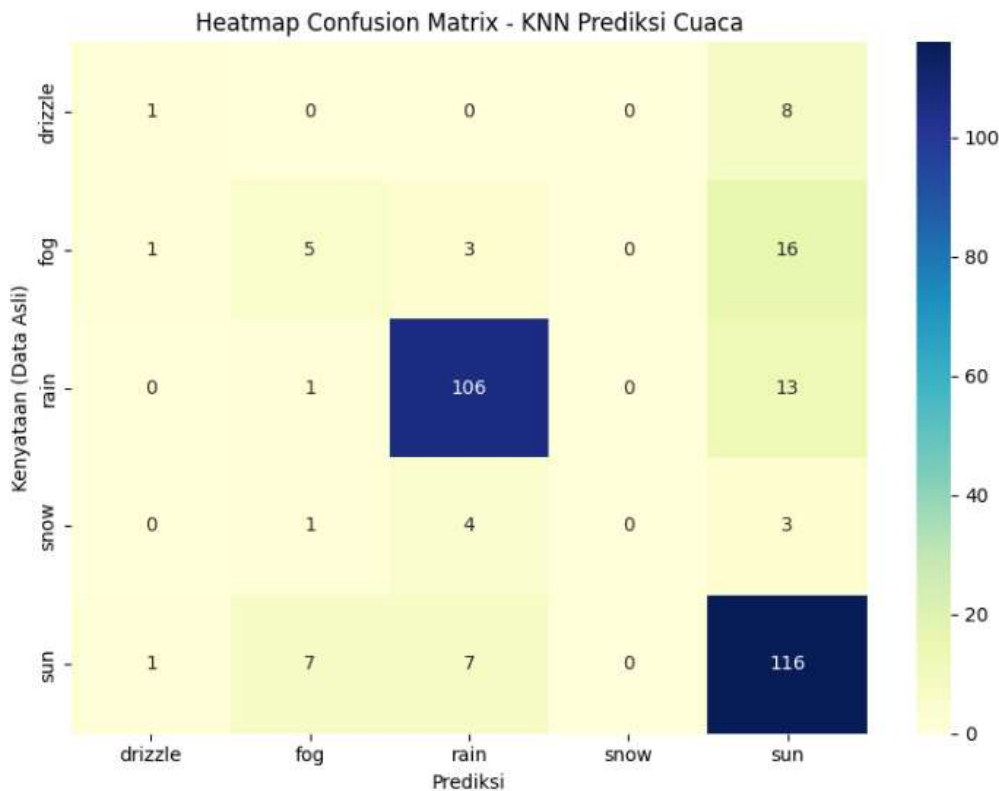
Akurasi Model K-NN: 0.7781

Tabel 1. Hasil evaluasi kinerja model K-NN

Label	Precision	Recall	F1-Score	Support
drizzle	0.33	0.11	0.17	9
fog	0.36	0.20	0.26	25
rain	0.88	0.88	0.88	120
snow	0.00	0.00	0.00	8
sun	0.74	0.89	0.81	131
accuracy			<b>0.78</b>	293
macro avg	0.46	0.42	0.42	293
weighted avg	0.73	0.78	0.75	293

Model KNN pada data uji mencapai akurasi sebesar 0.78 seperti yang ditampilkan pada Tabel 1, yang berarti sekitar 78% data kondisi cuaca berhasil diklasifikasikan dengan benar. Dari hasil klasifikasi, terlihat bahwa model sangat baik dalam mengenali kondisi cuaca seperti hujan (*rain*) dan cerah (*sun*), dengan precision dan recall masing-masing sebesar 0.88 dan 0.88 untuk *rain*, serta 0.74 dan 0.89 untuk *sun*. Namun, performa model menurun drastis pada kelas-kelas dengan jumlah data yang lebih sedikit seperti *drizzle*, *fog*, dan *snow*. Kelas *drizzle* memiliki precision 0.33 dan recall 0.11, sementara *fog* hanya mencapai precision 0.36 dan recall 0.20. Kelas *snow* menunjukkan performa terburuk dengan nilai *precision*, *recall*, dan *F1-score* sebesar 0.00, yang menandakan model gagal mendeteksi kondisi salju sama sekali. Ketimpangan distribusi data terlihat jelas

memengaruhi kinerja model, di mana kelas-kelas dengan support yang rendah cenderung tidak dikenali dengan baik. Hal ini juga tercermin dari nilai *macro average* yang cukup rendah yaitu *precision* 0.46 dan *recall* 0.42, dibandingkan dengan *weighted average* yang mencapai *precision* 0.73 dan *recall* 0.78 karena dipengaruhi oleh dominasi kelas *rain* dan *sun*. Keseluruhan hasil menunjukkan bahwa meskipun model cukup andal untuk kelas mayoritas, diperlukan penanganan khusus terhadap ketidakseimbangan data agar semua kondisi cuaca dapat dikenali secara lebih merata.



Gambar 2. Heatmap Confusion Matrix Klasifikasi Kondisi Cuaca dengan Model K-NN

Berdasarkan visualisasi confusion matrix dalam bentuk heatmap pada Gambar 2, model *K-Nearest Neighbors* (KNN) menunjukkan performa yang bervariasi dalam mengklasifikasikan kondisi cuaca. Model ini mampu mengklasifikasikan data dengan cukup baik pada kelas *rain* dan *sun*, yang masing-masing memiliki 106 dan 116 data yang diprediksi dengan benar. Hal ini menunjukkan bahwa model KNN cenderung akurat dalam mengenali dua kelas mayoritas tersebut. Namun, kelemahan model mulai terlihat pada kelas minoritas seperti *drizzle*, *fog*, dan *snow*. Pada kelas *drizzle*, hanya 1 data yang diklasifikasikan dengan benar, sedangkan 8 data lainnya salah diprediksi sebagai *sun*. Kelas *fog* juga menunjukkan hasil yang kurang optimal, di mana hanya 5 data yang diklasifikasikan dengan benar dari total 25 data, dan sebagian besar lainnya (16 data) kembali diklasifikasikan sebagai *sun*. Hal yang sama terjadi pada kelas *snow*, di mana model hanya mampu mengklasifikasikan 3 data dengan benar, sementara sisanya salah diprediksi sebagai *rain* atau *sun*. Kesalahan prediksi yang berulang pada kelas *sun* menunjukkan adanya kecenderungan model KNN untuk bias terhadap kelas yang lebih dominan dalam distribusi data. Hal ini menunjukkan bahwa model KNN rentan terhadap masalah *class imbalance*, karena algoritma ini sangat bergantung pada distribusi tetangga terdekat yang sering kali didominasi oleh kelas mayoritas. Meskipun demikian, model KNN tetap menunjukkan potensi yang cukup baik dalam klasifikasi cuaca, khususnya bila dilakukan peningkatan performa melalui normalisasi data, pemilihan parameter *k* yang optimal, atau penggunaan teknik penyeimbangan kelas seperti SMOTE.

### 3.2. Hasil pelatihan model menggunakan metode *Random Forest*

Setelah model K-NN, model *Random Forest* juga diterapkan dalam klasifikasi kondisi cuaca, berikut adalah hasil evaluasi kinerja model:

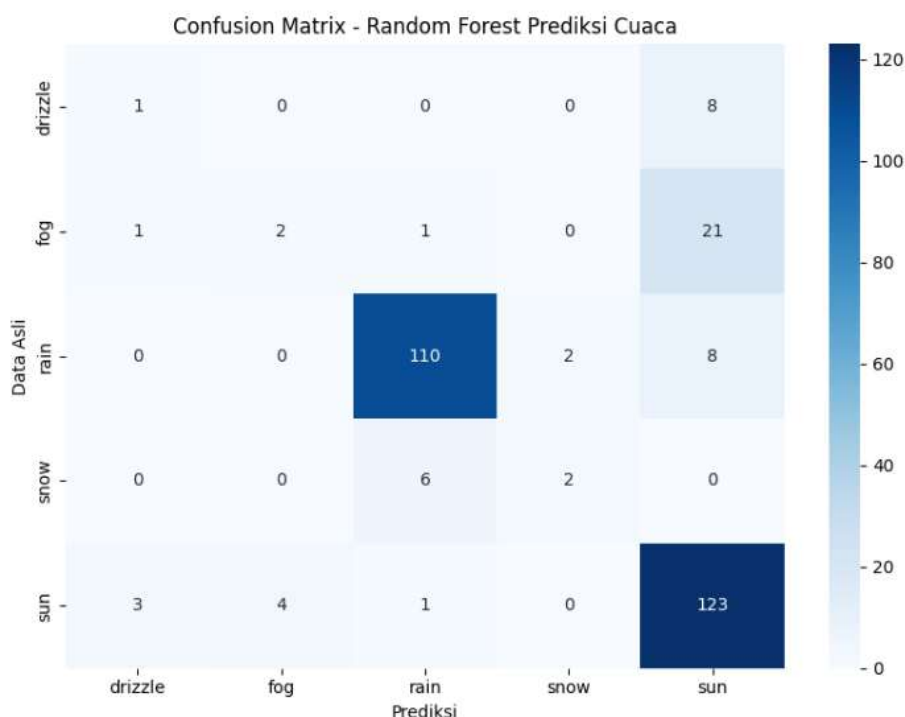
Akurasi Model Random Forest: 0.8259

Tabel 2. Hasil evaluasi kinerja model Random Forest

Label	Precision	Recall	F1-Score	Support
drizzle	0.25	0.11	0.15	9
fog	0.40	0.08	0.13	25
rain	0.93	0.93	0.93	120
snow	0.50	0.25	0.33	8
sun	0.77	0.95	0.85	131
accuracy			<b>0.82</b>	293
macro avg	0.57	0.46	0.48	293
weighted avg	0.78	0.82	0.79	293

Model *Random Forest* pada data uji mencapai akurasi sebesar 0.82 seperti yang ditampilkan pada Tabel 2, yang berarti sekitar 82% data kondisi cuaca berhasil diklasifikasikan dengan benar. Dari hasil klasifikasi, terlihat bahwa model sangat baik dalam mengenali kondisi cuaca hujan (*rain*) dan cerah (*sun*), dengan *precision* dan *recall* masing-masing sebesar 0.93 dan 0.93 untuk kelas *rain*, serta 0.77 dan 0.95 untuk kelas *sun*. Hal ini menunjukkan bahwa model mampu mengenali kedua kondisi tersebut dengan sangat baik dan konsisten. Namun, performa model menurun signifikan pada kelas-kelas dengan jumlah data yang lebih sedikit seperti *drizzle*, *fog*, dan *snow*. Kelas *drizzle* hanya memiliki *precision* sebesar 0.25 dan *recall* 0.11, sementara *fog* mencapai *precision* 0.40 namun *recall*-nya rendah di angka 0.08. Kelas *snow* juga menunjukkan performa rendah dengan *precision* 0.50 dan *recall* 0.25, yang berarti model masih kesulitan dalam mengenali kondisi salju secara akurat.

Ketimpangan distribusi data sangat memengaruhi kinerja model, di mana kelas-kelas dengan support yang rendah cenderung memiliki skor evaluasi yang lebih rendah. Hal ini tercermin dari nilai *macro average* yang relatif rendah, yaitu *precision* 0.57 dan *recall* 0.46, jika dibandingkan dengan *weighted average* yang mencapai *precision* 0.78 dan *recall* 0.82. Nilai *weighted average* yang tinggi menunjukkan bahwa model lebih dipengaruhi oleh performa pada kelas-kelas mayoritas seperti *rain* dan *sun*. Secara keseluruhan, hasil ini menunjukkan bahwa meskipun model *Random Forest* cukup andal dalam mengklasifikasikan kondisi cuaca yang dominan, perlu dilakukan penanganan terhadap ketidakseimbangan data agar model dapat mengenali semua kondisi cuaca secara lebih merata dan adil.



Gambar 3. Heatmap Confusion Matrix Klasifikasi Kondisi Cuaca dengan model Random Forest

Model *Random Forest* menunjukkan performa klasifikasi yang lebih baik dibandingkan KNN dalam memprediksi kondisi cuaca, sebagaimana terlihat dari confusion matrix yang divisualisasikan dalam bentuk heatmap pada Gambar 3. Model ini berhasil mengklasifikasikan data dengan akurasi tinggi terutama pada kelas *rain* dan *sun*, masing-masing dengan jumlah prediksi benar sebanyak 110 dan 123 data. Tidak hanya unggul dalam dua kelas mayoritas tersebut, model *Random Forest* juga

memberikan hasil yang lebih stabil pada kelas minoritas. Sebagai contoh, pada kelas *fog*, sebanyak 21 dari 25 data berhasil diklasifikasikan dengan benar, dan pada kelas *snow*, model mampu mengidentifikasi 6 data dengan benar dari total 10 data. Kesalahan klasifikasi pada model ini juga relatif lebih kecil dibandingkan KNN, meskipun masih terdapat beberapa data yang salah diprediksi ke kelas *sun*, namun jumlahnya tidak signifikan. Keunggulan *Random Forest* terletak pada kemampuannya dalam mengurangi overfitting dan sensitivitas terhadap distribusi data tidak seimbang, karena pendekatan *ensemble learning* yang digunakan menghasilkan prediksi akhir berdasarkan voting dari banyak pohon keputusan. Hal ini membuat *Random Forest* lebih tahan terhadap class imbalance dan lebih akurat dalam mengenali pola data yang kompleks. Dengan hasil ini, dapat disimpulkan bahwa *Random Forest* merupakan model yang lebih andal dan efektif untuk sistem klasifikasi cuaca dalam studi ini.

#### 4. KESIMPULAN

Penelitian ini menunjukkan bahwa algoritma *Random Forest* memiliki performa yang lebih unggul dibandingkan *K-Nearest Neighbors* (KNN) dalam tugas klasifikasi kondisi cuaca berdasarkan data meteorologis historis. Model *Random Forest* mencapai akurasi sebesar 82%, sedangkan KNN mencapai 78%. Kedua model menunjukkan performa tinggi pada kelas mayoritas seperti "rain" dan "sun", namun masih memiliki kelemahan dalam mengenali kelas minoritas seperti "drizzle", "fog", dan "snow". Ketidakseimbangan distribusi data menjadi faktor utama yang mempengaruhi rendahnya presisi dan recall pada kelas-kelas tersebut. Oleh karena itu, upaya penyeimbangan data serta optimasi parameter model sangat dianjurkan untuk penelitian selanjutnya guna meningkatkan kinerja klasifikasi secara menyeluruh. Hasil penelitian ini memperkuat pemahaman tentang pentingnya pemilihan algoritma yang tepat serta penanganan data yang sesuai dalam sistem klasifikasi cuaca berbasis pembelajaran mesin.

#### DAFTAR PUSTAKA

- [1] M. Farouk, S. Asklany, M. Abd, M. Farouk MAlI, S. A. Asklany, and M. Abd El-wahab, "Data Mining Algorithms for Weather Forecast Phenomena : Comparative Study," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 19, no. 9, p. 76, 2019, [Online]. Available: <https://www.researchgate.net/publication/337797654>
- [2] F. Q. Kareem, A. M. Abdulazeez, and D. A. Hasan, "Predicting Weather Forecasting State Based on Data Mining Classification Algorithms," *Asian J. Res. Comput. Sci.*, vol. 9, no. 3, pp. 13–24, 2021, doi: 10.9734/ajrcos/2021/v9i330222.
- [3] M. Abdurraheem, J. B. Awotunde, A. E. Adeniyi, I. D. Oladipo, and S. O. Adekola, "Weather prediction performance evaluation on selected machine learning algorithms," *IAES Int. J. Artif. Intell.*, vol. 11, no. 4, pp. 1535–1544, 2022, doi: 10.11591/ijai.v11.i4.pp1535-1544.
- [4] A. D. Putri, M. Al Haris, F. Fauzi, and S. Amri, "K-Nearest Neighbor ( KNN ) Method for Weather Data Prediction," vol. 3, no. June, pp. 56–64, 2025.
- [5] R. Gunawan, R. Septiadi, F. Apri Wenando, H. Mukhtar, and Syahril, "K-Nearest Neighbor (KNN) untuk Menganalisis Sentimen terhadap Kebijakan Merdeka Belajar Kampus Merdeka pada Komentar Twitter," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 2, pp. 152–158, 2022, doi: 10.37859/coscitech.v3i2.3841.
- [6] M. Alidin *et al.*, "Jurnal Computer Science and Information Technology ( CoSciTech ) Application of K-Nearest Neighbors to the Classification of Legal Cases in the Federal Courts of," vol. 6, no. 1, pp. 85–93, 2025.
- [7] I. Riadi, A. Yudhana, and M. Rosyidi Djou, "Comparative Analysis of Naïve Bayes and K-NN in Determining Location of Mobile Population Services," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 4, no. 3, pp. 733–742, 2024, doi: 10.37859/coscitech.v4i3.6543.
- [8] R. Meenal, P. A. Michael, D. Pamela, and E. Rajasekaran, "Weather prediction using random forest machine learning model," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, pp. 1208–1215, 2021, doi: 10.11591/ijeecs.v22.i2.pp1208-1215.
- [9] A. Mathew and J. Mathew, "Weather Forecasting Using the Random Forest Algorithm Analysis," vol. 4, no. 1, pp. 3–5, 2022, doi: 10.5281/zenodo.6361990.
- [10] A. Oktian Permana and Sudin Saepudin, "Perbandingan algoritma k-nearest neighbor dan naïve bayes pada aplikasi shopee," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 4, no. 1, pp. 25–32, 2023, doi: 10.37859/coscitech.v4i1.4474.