

## IMPLEMENTASI ALGORITMA K-NEAREST NEIGHBORS UNTUK KLASIFIKASI BUNGA IRIS BERBASIS WEB MENGGUNAKAN FLASK

PUTRA ALIANSYAH  
Universitas Bale Bandung, Bandung

**ABSTRAK:** Klasifikasi bunga Iris merupakan salah satu masalah klasik dalam pembelajaran mesin yang bertujuan untuk mengidentifikasi spesies bunga berdasarkan karakteristik morfologi. Penelitian ini mengimplementasikan algoritma K-Nearest Neighbors (KNN) untuk klasifikasi tiga spesies bunga Iris yaitu Setosa, Versicolor, dan Virginica dalam bentuk aplikasi web berbasis Flask. Metode yang digunakan meliputi preprocessing data dengan standarisasi menggunakan StandardScaler, pembagian data training dan testing dengan rasio 80:20, dan implementasi algoritma KNN dengan  $k=5$  dan metrik jarak Euclidean. Hasil penelitian menunjukkan bahwa model KNN mampu mencapai akurasi tinggi dalam mengklasifikasikan spesies bunga Iris berdasarkan empat fitur utama yaitu panjang sepal, lebar sepal, panjang petal, dan lebar petal. Aplikasi web yang dikembangkan menyediakan antarmuka yang user-friendly untuk melakukan prediksi secara real-time dengan menampilkan tingkat kepercayaan dan probabilitas untuk setiap kelas. Sistem juga menampilkan jarak ke tetangga terdekat dan informasi model yang digunakan.

**Kata kunci:** K-Nearest Neighbors, klasifikasi bunga Iris, Flask, machine learning, aplikasi web

**ABSTRACT:** Iris flower classification is one of the classic problems in machine learning that aims to identify flower species based on morphological characteristics. This study implements the K-Nearest Neighbors (KNN) algorithm for classifying three Iris flower species namely Setosa, Versicolor, and Virginica in the form of a Flask-based web application. The method used includes data preprocessing with standardization using StandardScaler, training and testing data split with an 80:20 ratio, and implementation of the KNN algorithm with  $k=5$  and Euclidean distance metric. The results show that the KNN model is able to achieve high accuracy in classifying Iris flower species based on four main features namely sepal length, sepal width, petal length, and petal width. The developed web application provides a user-friendly interface for real-time prediction by displaying confidence levels and probabilities for each class. The system also displays distances to nearest neighbors and information about the model used..

**Keywords:** K-Nearest Neighbors, Iris flower classification, Flask, machine learning, web application

## PENDAHULUAN

Klasifikasi bunga Iris merupakan salah satu dataset benchmark yang paling terkenal dalam bidang pembelajaran mesin dan pengenalan pola. Dataset ini pertama kali diperkenalkan oleh ahli statistik dan biologi Ronald Fisher pada tahun 1936 dan sejak itu menjadi standar untuk menguji algoritma klasifikasi. Dataset Iris terdiri dari 150 sampel bunga yang terbagi menjadi tiga spesies: Iris Setosa, Iris Versicolor, dan Iris Virginica, dengan masing-masing spesies memiliki 50 sampel.

Setiap sampel dalam dataset memiliki empat atribut numerik yang merepresentasikan karakteristik morfologi bunga: panjang sepal (sepal length), lebar sepal (sepal width), panjang petal (petal length), dan lebar petal (petal width). Keempat atribut ini diukur dalam satuan sentimeter dan menjadi dasar untuk membedakan ketiga spesies bunga Iris.

Algoritma K-Nearest Neighbors (KNN) merupakan salah satu algoritma pembelajaran mesin yang sederhana namun efektif untuk masalah klasifikasi. KNN menggunakan strategi pembelajaran sederhana: ketika diberikan observasi baru yang tidak diketahui, algoritma akan mencari dalam database referensi mana yang memiliki fitur terdekat dan menetapkan kelas yang predominan. Algoritma ini termasuk dalam kategori lazy learning karena tidak membangun model eksplisit selama fase training, melainkan menyimpan seluruh data training dan melakukan komputasi saat prediksi dilakukan.

Perkembangan teknologi web memungkinkan implementasi algoritma pembelajaran mesin dalam bentuk aplikasi web yang dapat diakses secara real-time. Flask, sebagai framework web Python yang ringan dan fleksibel, menyediakan platform yang ideal untuk mengembangkan aplikasi pembelajaran mesin berbasis web. Kombinasi Flask dengan library scikit-learn memungkinkan pengembangan sistem klasifikasi yang dapat digunakan oleh pengguna awam tanpa perlu pengetahuan teknis mendalam tentang pembelajaran mesin.

Penelitian ini bertujuan untuk mengimplementasikan algoritma KNN dalam bentuk aplikasi web menggunakan Flask untuk klasifikasi bunga Iris. Aplikasi yang dikembangkan diharapkan dapat memberikan prediksi yang akurat dengan antarmuka yang mudah digunakan, serta memberikan informasi tambahan seperti tingkat kepercayaan, probabilitas kelas, dan jarak ke

tetangga terdekat untuk meningkatkan transparansi dan interpretabilitas model.

## METODA

Penelitian ini menggunakan pendekatan eksperimental dengan implementasi algoritma K-Nearest Neighbors untuk klasifikasi bunga Iris. Metodologi penelitian meliputi beberapa tahap utama: pengumpulan dan preprocessing data, pembangunan model, pengembangan aplikasi web, dan evaluasi sistem.

### Dataset dan Preprocessing

Dataset yang digunakan adalah dataset Iris yang tersedia dalam library scikit-learn. Dataset iris berisi tiga kelas bunga: Versicolor, Setosa, Virginica, dan setiap kelas memiliki 4 fitur: 'Sepal length', 'Sepal width', 'Petal length', 'Petal width'. Dataset terdiri dari 150 sampel dengan distribusi yang seimbang (50 sampel per kelas).

Tahap preprocessing meliputi standarisasi fitur menggunakan StandardScaler untuk memastikan semua fitur memiliki skala yang sama. Hal ini penting untuk algoritma KNN karena menggunakan perhitungan jarak Euclidean yang sensitif terhadap skala fitur. Data kemudian dibagi menjadi set training dan testing dengan rasio 80:20 menggunakan stratified sampling untuk mempertahankan proporsi kelas.

#### • Implementasi Algoritma KNN

Algoritma KNN diimplementasikan menggunakan library scikit-learn dengan parameter sebagai berikut:

- Jumlah tetangga (k) = 5
- Metrik jarak = Euclidean
- Algoritma pencarian = auto (otomatis memilih algoritma terbaik)

Pemilihan nilai k=5 didasarkan pada praktik umum menggunakan nilai k yang tidak terlalu kecil (untuk menghindari overfitting) namun tidak terlalu besar (untuk mempertahankan lokalitas). Model dilatih menggunakan data training yang telah dinormalisasi dan disimpan dalam format pickle untuk penggunaan berulang.

### Pengembangan Aplikasi Web

Aplikasi web dikembangkan menggunakan framework Flask dengan arsitektur Model-View-Controller (MVC). Komponen utama aplikasi meliputi:

1. **Backend (Model):** Implementasi algoritma KNN, preprocessing data, dan fungsi prediksi

2. **Frontend (View):** Antarmuka pengguna berbasis HTML, CSS, dan JavaScript
3. **Controller:** Route handler Flask untuk mengelola request dan response

Aplikasi menyediakan beberapa endpoint:

- / : Halaman utama dengan form input
- /predict : API endpoint untuk prediksi
- /model-info : Informasi tentang model
- /about : Halaman informasi aplikasi

### Evaluasi dan Validasi

Model dievaluasi menggunakan metrik akurasi dan classification report pada data testing. Sistem juga menyediakan informasi tambahan untuk setiap prediksi:

- Confidence score berdasarkan probabilitas maksimum
- Probabilitas untuk setiap kelas
- Jarak ke 5 tetangga terdekat
- Informasi detail model yang digunakan

## HASIL DAN PEMBAHASAN

### Performa Model KNN

Implementasi algoritma KNN pada dataset Iris menunjukkan hasil yang sangat baik. Model yang dilatih dengan parameter  $k=5$  dan metrik jarak Euclidean mampu mencapai tingkat akurasi yang tinggi pada data testing. Algoritma KNN adalah salah satu algoritma klasifikasi paling sederhana. Bahkan dengan kesederhanaan seperti itu, ia dapat memberikan hasil yang sangat kompetitif.

Distribusi data menunjukkan karakteristik yang berbeda untuk setiap spesies. Iris Setosa memiliki karakteristik yang paling mudah dibedakan dengan ukuran petal yang relatif kecil. Iris Versicolor memiliki ukuran sedang dengan fitur yang berada di antara Setosa dan Virginica. Sedangkan Iris Virginica memiliki ukuran petal dan sepal yang paling besar di antara ketiga spesies.

### Antarmuka Aplikasi Web

Aplikasi web yang dikembangkan menyediakan antarmuka yang intuitif dan user-friendly. Halaman utama menampilkan form input untuk empat fitur bunga Iris dengan validasi input untuk memastikan nilai yang dimasukkan berada dalam rentang yang wajar (0-20 cm).

Form input dilengkapi dengan tombol "Fill Example Values" yang memudahkan pengguna untuk mencoba aplikasi dengan data contoh. Setelah pengguna memasukkan nilai dan menekan tombol "Predict Flower Species",

aplikasi akan menampilkan hasil prediksi dalam bentuk yang informatif.

### Hasil Prediksi dan Interpretabilitas

Sistem prediksi tidak hanya memberikan hasil klasifikasi, tetapi juga menyediakan informasi yang meningkatkan interpretabilitas model:

1. **Predicted Species:** Nama spesies yang diprediksi
2. **Confidence Score:** Tingkat kepercayaan prediksi dalam persentase
3. **Class Probabilities:** Probabilitas untuk setiap kelas (Setosa, Versicolor, Virginica)
4. **Distance to Nearest Neighbors:** Jarak Euclidean ke 5 tetangga terdekat

Informasi ini membantu pengguna memahami seberapa yakin model terhadap prediksinya dan memberikan transparansi dalam proses pengambilan keputusan.

### Fitur Informasi Model

Aplikasi juga menyediakan halaman informasi model yang menampilkan detail teknis:

- **Algorithm:** K-Nearest Neighbors (KNN)
- **K Value:** 5 neighbors
- **Distance Metric:** Euclidean
- **Features:** 4 measurements (sepal length, sepal width, petal length, petal width)
- **Classes:** 3 species (Setosa, Versicolor, Virginica)
- **Training Data:** Iris Dataset

### Analisis Keunggulan dan Keterbatasan

#### Keunggulan sistem:

1. **Simplicity:** Algoritma KNN mudah dipahami dan diimplementasikan
2. **No Training Period:** Tidak memerlukan waktu training yang lama
3. **Versatility:** Dapat digunakan untuk klasifikasi dan regresi
4. **Interpretability:** Hasil mudah dijelaskan kepada pengguna
5. **Web Accessibility:** Dapat diakses dari berbagai perangkat melalui browser

#### Keterbatasan sistem:

1. **Computational Cost:** Memerlukan komputasi untuk setiap prediksi
2. **Memory Usage:** Menyimpan seluruh data training
3. **Sensitivity to Scale:** Memerlukan normalisasi data

4. **Curse of Dimensionality:** Performa menurun pada data berdimensi tinggi

#### **Validasi Input dan Error Handling**

Sistem dilengkapi dengan mekanisme validasi input yang komprehensif:

- Validasi numerik untuk memastikan input berupa angka
- Validasi rentang untuk memastikan nilai berada dalam batas wajar
- Error handling untuk menangani kesalahan prediksi
- Pesan error yang informatif untuk membantu pengguna

#### **PUSTAKA ACUAN**

##### **Jurnal/Prosiding/Disertasi/Tesis/Skripsi**

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2), 179-188.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification*. Second Edition. New York: John Wiley & Sons.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*. New York: Springer.

##### **Buku**

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.

Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. Second Edition. O'Reilly Media.

##### **Lain-lain**

Scikit-learn Development Team. (2024). *Scikit-learn: Machine Learning in Python*.

Flask Development Team. (2024). *Flask Documentation*.

<https://flask.palletsprojects.com/> (Diunduh tanggal 15 Juni 2025).

UCI Machine Learning Repository. (2024). *Iris Data Set*.

<https://archive.ics.uci.edu/ml/datasets/iris> (Diunduh tanggal 15 Juni 2025).