

## Contoh Kasus Menggunakan Algoritma AHP

### **Contoh Kasus : Pemilihan Koordinator Mahasiswa Kelas DSS**

Kelas DSS sedang memilih Koordinator Mahasiswa yang terdiri dari 3 orang kandidat (Suhardiman, Ronaldo dan Messi). Ada dua kriteria yang menjadi pertimbangan pemilihan yaitu Indeks Prestasi Kumulatif (IPK) dan Kepemimpinan. Bantu kelas DSS untuk memilih Koordinator Mahasiswa menggunakan algoritma AHP berdasarkan informasi di bawah ini:

- IPK 3x lebih penting dari Kepemimpinan
- Kepemimpinan 1/3x dari IPK
- Untuk IPK:
  - Suhardiman lebih baik dari Ronaldo pada IPK sebesar 2x
  - Suhardiman lebih baik dari Messi pada IPK sebesar 3x
  - Ronaldo lebih baik dari Messi pada IPK sebesar 1.5x
- Untuk Kepemimpinan:
  - Suhardiman lebih baik dari Ronaldo pada Kepemimpinan sebesar 3x
  - Suhardiman lebih baik dari Messi pada Kepemimpinan sebesar 2x
  - Ronaldo lebih baik dari Messi pada Kepemimpinan sebesar 1.5x

### JAWABAN

#### **Langkah 1: Inisialisasi**

```
import numpy as np

# Fungsi untuk menghitung bobot relatif
def calculate_weights(matrix):
    n = len(matrix)
    eigenvalues, eigenvectors = np.linalg.eig(matrix)
    max_eigenvalue_index = np.argmax(eigenvalues)
    max_eigenvector = eigenvectors[:, max_eigenvalue_index]
    weights = max_eigenvector / np.sum(max_eigenvector)
    return weights
```

#### **Langkah 2: Matriks Perbandingan Berpasangan untuk Kriteria**

```
# Matriks perbandingan berdasarkan informasi perbandingan antar-kriteria
criteria_matrix = np.array([
    [1, 3],
    [1/3, 1]
])
```

### Langkah 3: Matriks Perbandingan Berpasangan untuk Alternatif

```
# Matriks perbandingan untuk IPK
ipk_matrix = np.array([
    [1, 2, 3],
    [1/2, 1, 1.5],
    [1/3, 2/3, 1]
])

# Matriks perbandingan untuk Kepemimpinan
leadership_matrix = np.array([
    [1, 3, 2],
    [1/3, 1, 1.5],
    [1/2, 2/3, 1]
])
```

### Langkah 4: Menghitung Bobot

```
# Menghitung bobot relatif untuk kedua kriteria
criteria_weights = calculate_weights(criteria_matrix)

# Menghitung bobot relatif untuk IPK dan Kepemimpinan
ipk_weights = calculate_weights(ipk_matrix)
leadership_weights = calculate_weights(leadership_matrix)

# Hasil akhir untuk setiap alternatif
result_Suhardiman = np.dot(criteria_weights, [ipk_weights[0], leadership_weights[0]])
result_Ronaldo = np.dot(criteria_weights, [ipk_weights[1], leadership_weights[1]])
result_Messi = np.dot(criteria_weights, [ipk_weights[2], leadership_weights[2]])
```

### Langkah 5: Mendapatkan Hasil

```
# Menampilkan hasil
print("Hasil Perannkingan Kandidat Koordinator Mahasiswa Kelas DSS:")
print(f"Suhardiman dengan hasil {result_A:.3f}")
print(f"Ronaldo B dengan hasil {result_B:.3f}")
print(f"Messi C dengan hasil {result_C:.3f}")
```

SILAKAN IMPLEMENTASIKAN KODE PYTHON DI ATAS DI KOMPUTER ANDA!

SELAMAT MENJALANKAN 😊