KOMPUTA : Jurnal Ilmiah Komputer dan Informatika
Vol. xx, No. x, mmmm yyyy, ISSN : 2089-9033, e- ISSN : 2715-7849

# Ramgkuman Materi Perkuliahan  Machine learning Kelompok 21

Muhammad Shaadam haidar Yuwono[1], Muhammad labib[2]

## 1.Deskripsi singkat

Pertemuan ini membahas teknik pengelompokan data (*clustering*) tingkat lanjut dengan membandingkan metode konvensional berbasis jarak seperti K-Means dengan metode berbasis distribusi statistik yaitu **Gaussian Mixture Model (GMM)**. Fokus utamanya adalah memahami bagaimana mesin mengidentifikasi kelompok tersembunyi berdasarkan pola probabilitas dan distribusi data.

**Poin-poin Utama:**

1. Transisi Paradigma Klasterisasi:

Berbeda dengan K-Means yang hanya menghitung jarak fisik ke titik tengah (centroid), GMM bekerja dengan asumsi bahwa setiap kelompok data mengikuti pola Distribusi Normal (Gaussian). Hal ini membuat GMM lebih fleksibel dalam menangani kelompok data yang bentuknya tidak bulat sempurna atau lonjong.

2. Identifikasi Kelompok melalui Grafik Frekuensi:

Dosen menjelaskan bahwa kelompok yang berbeda dapat dideteksi melalui "puncak" atau kenaikan data yang tiba-tiba pada grafik frekuensi. Setiap puncak merepresentasikan satu distribusi normal yang unik dalam sebuah populasi data yang besar.

3. **Dua Parameter Kunci ($\mu$ dan $\sigma$):**
    a. **Mu ($\mu$):** Menunjukkan rata-rata atau lokasi pusat dari sebuah kelompok.
    b. **Sigma ($\sigma$):** Menunjukkan standar deviasi atau lebar persebaran data. Memahami kedua nilai ini sangat penting untuk mendefinisikan identitas matematis dari sebuah klaster.
4. Konsep Keanggotaan Lembut (Soft Membership):

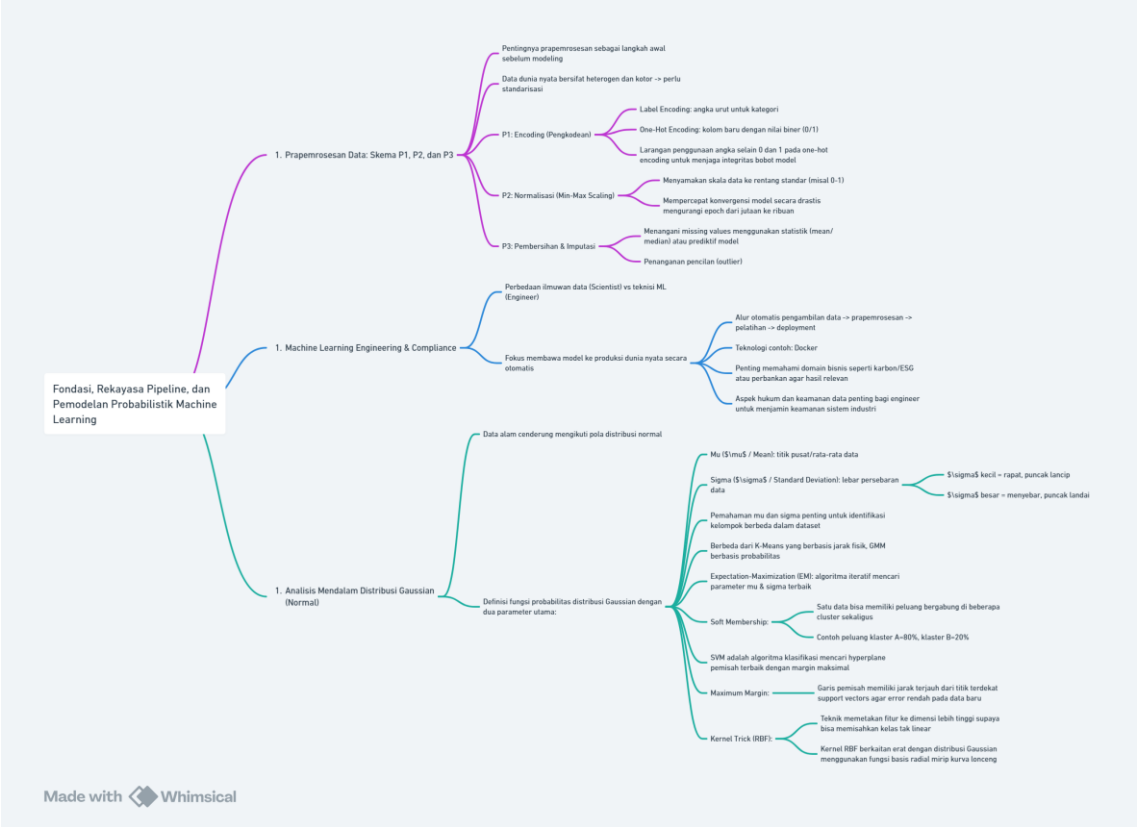
Dalam GMM, satu data tidak harus kaku dimiliki oleh satu kelompok saja. Sebaliknya, setiap data memiliki nilai probabilitas (tanggung jawab) untuk menjadi bagian dari beberapa kelompok sekaligus, yang dihitung menggunakan fungsi padat probabilitas.

5. Optimasi dengan Algoritma EM:

GMM menggunakan siklus Expectation-Maximization (EM) untuk menemukan posisi dan lebar kelompok yang paling pas. Proses ini mirip dengan logika optimasi SGD pada Neural Networks, namun digunakan untuk model generatif dan probabilistik.

Naskah dikirim: dd-mm-yyyy; direvisi dd-mm-yyyy; diterima: dd-mm-yyyy

## 2.Mindmap




## 3.penjelasan detil

### Fondasi, Rekayasa Pipeline, dan Pemodelan Probabilistik Machine Learning

#### 1. Prapemrosesan Data: Skema P1, P2, dan P3

Dosen menekankan bahwa prapemrosesan adalah langkah paling menentukan sebelum data masuk ke model. Data dunia nyata bersifat heterogen dan kotor, sehingga perlu distandarisasi.

- **P1 - Encoding (Pengkodean):** Mengubah data teks/kategorikal menjadi angka.
  - *Label Encoding:* Memberikan angka urut.
  - *One-Hot Encoding:* Membuat kolom baru berisi 0 atau 1. Dosen melarang penggunaan angka selain 0 dan 1 dalam sistem biner ini untuk menjaga integritas bobot model.
- **P2 - Normalisasi (Min-Max Scaling):** Menyamakan skala data (misal dari skala jutaan ke rentang 0-1).

    - **Penting:** Normalisasi secara drastis mempercepat konvergensi (pencapaian titik optimal) model, mengurangi jumlah *epoch* dari jutaan menjadi ribuan.
- **P3 - Pembersihan & Imputasi:** Menangani nilai kosong (*missing values*) dengan statistik (mean/median) atau model prediktif, serta penanganan pencilan (*outlier*).

**2. Machine Learning Engineering & Compliance**

Materi ini membedakan antara ilmuwan (*Scientist*) dan teknisi (*Engineer*). Fokus utamanya adalah bagaimana membawa model ke dunia nyata.

- **Pipeline Engineering:** Membangun alur otomatis dari pengambilan data -> prapemrosesan -> pelatihan -> *deployment* (penerapan) menggunakan teknologi seperti **Docker**.
- **Business Understanding:** Informatika adalah matematika terapan. Praktisi ML harus memahami domain bisnisnya (seperti proyek karbon/ESG atau perbankan) agar model yang dibuat akurat dan relevan secara fakta.
- **Compliance (Kepatuhan):** Memahami aspek hukum dan keamanan data. Seorang engineer yang paham *compliance* memiliki nilai ekonomi lebih tinggi karena menjamin keamanan sistem di industri.

**3. Analisis Mendalam: Distribusi Gaussian (Normal)**

Ini adalah poin paling penting dalam pemodelan statistik. Dosen menjelaskan bahwa data di alam cenderung mengikuti pola distribusi normal.

Definisi Detil:

Distribusi Gaussian adalah fungsi probabilitas yang menunjukkan bahwa data cenderung berkumpul di sekitar satu titik pusat (rata-rata). Identitasnya ditentukan oleh dua parameter:

- **Mu ($\mu$ / Mean):** Lokasi puncak tertinggi atau rata-rata data.
- **Sigma ($\sigma$ / Standard Deviation):** Ukuran lebar persebaran data. $\sigma$ kecil berarti data rapat (gunung lancip), $\sigma$ besar berarti data menyebar (gunung landai).

Memahami $\mu$ dan $\sigma$ sangat penting untuk mengidentifikasi adanya kelompok berbeda dalam satu dataset (misalnya melihat dua "puncak gunung" pada grafik frekuensi).

**4. Algoritma Klasterisasi: GMM dan EM**

Berbeda dengan K-Means yang berbasis jarak fisik, **Gaussian Mixture Model (GMM)** menggunakan probabilitas.

- **Expectation-Maximization (EM):** Algoritma yang digunakan GMM untuk menemukan $\mu$ dan $\sigma$ yang paling pas secara iteratif.
- **Soft Membership:** Dalam GMM, satu data tidak kaku di satu kelompok, melainkan memiliki peluang (misal 80% kelompok A, 20% kelompok B).

**5. Support Vector Machine (SVM) dan Kernel Trick**

SVM adalah algoritma klasifikasi yang mencari "pagar" atau garis pemisah terbaik.

**Definisi Detil:**

- **Maximum Margin:** SVM mencari garis pemisah (*hyperplane*) yang memiliki jarak terjauh dengan titik data terdekat (Support Vectors) agar model memiliki tingkat kesalahan yang rendah pada data baru.
- **Kernel Trick (RBF):** Teknik untuk memisahkan data yang tidak bisa dipisah garis lurus dengan cara memetakannya ke dimensi yang lebih tinggi. Kernel RBF sangat berkaitan dengan distribusi Gaussian karena menggunakan fungsi basis radial yang mirip kurva lonceng.

## 4.pseoducode

// ALGORITMA SUPPORT VECTOR MACHINE (SVM) // Berbasis pada Margin Maksimum dan Distribusi Gaussian

BEGIN SVM_CLASSIFICATION

```
// 1. PENYIAPAN DATA & LABEL
INPUT dataset_fitur (X)
INPUT label_target (Y) // Nilai biner: -1 atau 1

// 2. PRAPEMROSESAN (Langkah Krusial agar SVM Akurat)
FOR EACH kolom IN X:
    // Standarisasi agar data mengikuti Distribusi
Normal (Mu=0, Sigma=1)
    X_scaled = (X - Mean) / Standard_Deviation
END FOR
```

```
// 3. KONFIGURASI PARAMETER MODEL
SET C = 1.0           // Parameter penalti (Trade-off
antara margin & error)
SET Gamma = 1 / (2 * Sigma^2) // Menggunakan Sigma dari
Distribusi Gaussian
SET Tolerance = 0.001 // Batas berhenti iterasi

// 4. FUNGSI KERNEL GAUSSIAN (RBF)
// Menghitung kemiripan data di dimensi yang lebih
tinggi
FUNCTION Gaussian_Kernel(x1, x2, gamma):
    squared_distance = SUM((x1 - x2)^2)
    RETURN exp(-gamma * squared_distance)
END FUNCTION

// 5. PROSES OPTIMASI (Mencari Lagrange Multipliers /
Alpha)
// Mencari Hyperplane dengan Margin Terlebar
INITIALIZE alpha = [0, 0, ..., 0]
INITIALIZE bias = 0

REPEAT UNTUK SETIAP iterasi:
    FOR EACH data_i, data_j IN dataset:
        // Hitung error prediksi saat ini
        Error_i = Predict_Score(data_i) - Y_i

        // Jika data melanggar kondisi KKT (berada di
dalam margin)
        IF (Y_i * Error_i < -Tolerance) OR (Y_i *
Error_i > Tolerance):
            // Update Alpha_i dan Alpha_j menggunakan
metode SMO (Sequential Minimal Optimization)
            // Update Bias (b) berdasarkan Support
Vectors yang baru
        END IF
    END FOR
UNTIL alpha_tidak_berubah_signifikan OR
max_iterasi_tercapai

// 6. IDENTIFIKASI SUPPORT VECTORS
```

```
// Hanya menyimpan data yang memiliki Alpha > 0
SUPPORT_VECTORS = Dapatkan_Titik_Data(dimana alpha > 0)

// 7. FUNGSI PREDIKSI UNTUK DATA BARU
FUNCTION Predict(data_baru):
    hasil_sum = 0
    FOR EACH sv IN SUPPORT_VECTORS:
        hasil_sum += alpha_sv * label_sv *
Gaussian_Kernel(sv, data_baru, Gamma)
    END FOR

    score = hasil_sum + bias

    IF score >= 0 THEN
        RETURN Class_Positif (+1)
    ELSE
        RETURN Class_Negatif (-1)
    END IF
END FUNCTION
```

END SVM_CLASSIFICATION