1. Silhouette Score Rendah meski Elbow Menunjukkan K=5

Silhouette score yang rendah (0.3) menunjukkan bahwa banyak titik data berada di batas antar cluster, sehingga pemisahan kurang jelas.

Elbow method hanya mempertimbangkan penurunan inertia (jarak dalam cluster), bukan kualitas pemisahan antar cluster.

Penyebab inkonsistensi:

- Data tidak membentuk pola spherical (bulat), sehingga K-Means kesulitan.
- Variansi antar cluster yang berbeda (heterogen).

Solusi alternatif:

- Gap Statistic: Membandingkan inertia dengan data acak untuk validasi jumlah cluster.
- Bootstrap clustering stability: Mengukur seberapa konsisten hasil clustering pada sampling ulang data.

Distribusi non-spherical membuat centroid-based method seperti K-Means kurang efektif.

2. Preprocessing Campuran Numerik dan Teks untuk Clustering

Untuk fitur numerik seperti Quantity dan UnitPrice, gunakan normalisasi (misalnya StandardScaler atau MinMax).

Untuk fitur kategorikal seperti Description yang memiliki high-cardinality, One-Hot Encoding tidak efisien karena:

- Menghasilkan vektor sparse berdimensi tinggi.
- Menyebabkan curse of dimensionality dan overfitting.

Alternatif yang lebih baik:

- TF-IDF: Menghitung bobot relevansi kata dalam teks.
- Embedding berdimensi rendah seperti UMAP: Memampatkan representasi kata sambil menjaga hubungan semantik dan struktur cluster.

3. Sensitivitas Parameter Epsilon pada DBSCAN

DBSCAN sangat sensitif terhadap nilai epsilon (eps). Nilai terlalu kecil menghasilkan banyak noise; terlalu besar menyatukan cluster.

Solusi:

- Gunakan k-distance graph (plot jarak ke tetangga ke-k), lalu ambil nilai 'tekukan' (elbow) atau kuartil ke-3 sebagai nilai eps adaptif.
- MinPts dapat disesuaikan berdasarkan dimensi data atau regional density, misalnya MinPts = 2 * dimensi atau lebih tinggi pada cluster padat (contoh: pelanggan dari UK).

4. Overlap Cluster High-Value vs Bulk Buyers

Overlap menunjukkan bahwa metrik jarak tidak cukup membedakan segmen yang mirip.

Solusi:

- Constrained Clustering (misalnya COP-KMeans): Menambahkan label atau aturan 'must-link' dan 'cannot-link' untuk memandu pemisahan.
- Metric Learning (contoh Mahalanobis Distance): Memodifikasi ruang jarak agar fitur yang relevan lebih dominan.

Tantangannya: Pendekatan ini dapat mengaburkan interpretasi bisnis karena hasil clustering tidak lagi hanya berdasarkan jarak Euclidean biasa.

5. Merancang Fitur Temporal dari InvoiceDate

Fitur seperti 'day of week' atau 'hour of day' dapat menangkap pola perilaku konsumen (misalnya, belanja pagi vs malam).

Hindari data leakage dengan tidak menghitung agregasi temporal (misalnya rata-rata bulanan) dari seluruh dataset sebelum split.

Gunakan time-based cross-validation.

Lag features (misalnya pembelian 7 hari sebelumnya) berisiko menambahkan noise jika pola pembelian tidak cukup konsisten antar waktu, terutama jika digunakan untuk clustering yang tidak mempertimbangkan urutan waktu.