1. Jika algoritma K-Means menghasilkan nilai silhouette score rendah (0.3) meskipun elbow method menunjukkan K=5 sebagai optimal pada dataset ini, faktor apa yang menyebabkan inkonsistensi ini? Bagaimana strategi validasi alternatif (misal: analisis gap statistic atau validasi stabilitas cluster via bootstrapping) dapat mengatasi masalah ini, dan mengapa distribusi data non-spherical menjadi akar masalahnya?

   Jawab :

   Inkonsistensi antara elbow method (K=5 optimal) dengan silhouette score rendah terjadi karena elbow method hanya melihat total within-cluster variance tanpa mempertimbangkan pemisahan antar-cluster. Gap statistic atau validasi stabilitas via bootstrapping bisa membantu memberikan estimasi jumlah cluster yang lebih robust dengan mempertimbangkan stabilitas dan distribusi internal data. Distribusi data non-spherical menjadi akar masalah karena K-Means mengasumsikan bentuk cluster spherical dengan ukuran yang relatif seragam.

2. Dalam dataset dengan campuran fitur numerik (Quantity, UnitPrice) dan kategorikal high-cardinality (Description), metode preprocessing apa yang efektif untuk menyelaraskan skala dan merepresentasikan fitur teks sebelum clustering? Jelaskan risiko menggunakan One-Hot Encoding untuk Description, dan mengapa teknik seperti TF-IDF atau embedding berdimensi rendah (UMAP) lebih robust untuk mempertahankan struktur cluster!

   Jawab :

   Preprocessing yang efektif adalah menggunakan TF-IDF atau embedding berdimensi rendah (misal UMAP) untuk fitur teks dan normalisasi untuk fitur numerik.Risiko One-Hot Encoding pada high-cardinality data adalah menghasilkan matriks fitur sparse yang besar, menyulitkan interpretasi cluster dan meningkatkan kompleksitas. TF-IDF atau embedding berdimensi rendah lebih baik karena mampu menangkap struktur semantik dari data teks dan menghasilkan fitur yang lebih informatif dengan dimensi yang efisien.

3. Hasil clustering dengan DBSCAN sangat sensitif terhadap parameter epsilon—bagaimana menentukan nilai optimal epsilon secara adaptif untuk memisahkan cluster padat dari noise pada data transaksi yang tidak seimbang (misal: 90% pelanggan dari UK)? Jelaskan peran k-distance graph dan kuartil ke-3 dalam automasi parameter, serta mengapa MinPts harus disesuaikan berdasarkan kerapatan regional!

   Jawab :

   Menentukan epsilon optimal DBSCAN adaptif bisa menggunakan k-distance graph (jarak ke titik terdekat) lalu memilih titik infleksi (elbow point) atau menggunakan kuartil ketiga (Q3) untuk otomatisasi. K-distance graph membantu mengidentifikasi jarak yang sesuai antara titik-titik yang menunjukkan perbedaan signifikan antara cluster padat dan noise. MinPts perlu disesuaikan untuk mengakomodasi kepadatan regional agar cluster tidak terpecah atau menyatu secara tidak tepat.

4. Jika analisis post-clustering mengungkapkan overlap signifikan antara cluster "high-value customers" dan "bulk buyers" berdasarkan total pengeluaran, bagaimana teknik semi-supervised (contoh: constrained clustering) atau integrasi metric learning (Mahalanobis distance) dapat memperbaiki pemisahan cluster? Jelaskan tantangan dalam mempertahankan interpretabilitas bisnis saat menggunakan pendekatan non-Euclidean!

   Jawab :

   Menggunakan constrained clustering atau metric learning (Mahalanobis distance) dapat memperbaiki pemisahan cluster dengan memperkenalkan supervisi atau pembelajaran jarak yang lebih relevan dengan pola bisnis. Tantangannya adalah menjaga interpretabilitas bisnis karena penggunaan jarak non-Euclidean mungkin tidak intuitif, meskipun secara statistik lebih akurat.

5. Bagaimana merancang temporal features dari InvoiceDate (misal: hari dalam seminggu, jam pembelian) untuk mengidentifikasi pola pembelian periodik (seperti transaksi pagi vs. malam)? Jelaskan risiko data leakage jika menggunakan agregasi temporal (misal: rata-rata pembelian bulanan) tanpa time-based cross-validation, dan mengapa lag features (pembelian 7 hari sebelumnya) dapat memperkenalkan noise pada cluster!

   Jawab :

   Merancang temporal features dapat berupa jam pembelian, hari dalam seminggu, periode waktu (pagi, siang, malam). Risiko data leakage muncul jika menggunakan agregasi temporal seperti rata-rata bulanan tanpa validasi silang berbasis waktu, karena data masa depan bocor ke masa lalu. Lag features dapat memperkenalkan noise karena adanya transaksi acak yang tidak relevan dalam rentang lag tersebut, sehingga justru mengaburkan pola temporal yang diinginkan.