

**KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN
TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN
ALGORITMA *K-MEANS* DAN *DBSCAN***

SKRIPSI

DISUSUN OLEH

GAIZKA PASYA DERMAWAN SINUKABAN

2209020080



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

MEDAN

2026

**KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN
TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN
ALGORITMA K-MEANS DAN DBSCAN**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana
Komputer (S.Kom) dalam Program Studi Teknologi Informasi. Pada
Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas
Muhammadiyah Sumatera Utara**

GAIZKA PASYA DERMAWAN SINUKABAN

NPM. 2209020080

**PROGRAM STUDI TEKNOLOGI INFROMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

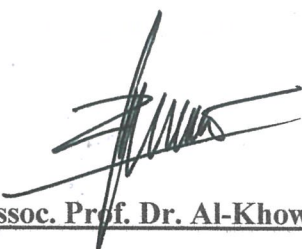
MEDAN

2026

LEMBAR PENGESAHAN

Judul Skripsi : Klaterisasi Daerah Rawan Kriminalitas di Medan Timur Berdasarkan Data Kepolisian Menggunakan Algoritma K-Means Dan DBSCAN
Nama Mahasiswa : Gaizka Pasya Dermawan Sinukaban
NPM : 2209020080
Program Studi : Teknologi Informasi

Menyetujui
Komisi Pembimbing


(Assoc. Prof. Dr. Al-Khowarizmi, M.Kom.)
NIDN. 0127099201

Ketua Program Studi

Dekan


(Fatma Sari Hutagalung, M.Kom.)
NIDN. 0117019301



(Assoc. Prof. Dr. Al-Khowarizmi, M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN ALGORITMA K-MEANS DAN DBSCAN

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 30 Maret 2026

Yang membuat pernyataan



Gaizka Pasya Dermawan Sinukaban

NPM. 2209020080

PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama : Gaizka Pasya Dermawan Sinukaban
NPM : 2209020080
Program Studi : Teknologi Informasi
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bebas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN
TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN
ALGORITMA K-MEANS DAN DBSCAN

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 30 Maret 2026

Yang membuat pernyataan



Gaizka Pasya Dermawan Sinukaban

NPM. 2209020080

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Gaizka Pasya Dermawan Sinukaban
Tempat dan Tanggal Lahir : Medan, 15 Januari 2003
Alamat Rumah : Rantauprapat, Sumatera Utara
Telepon/Faks/HP : 085359518699
E-mail : sinukabangaizka@gmail.com
Instansi Tempat Kerja : -
Alamat Kantor : -

DATA PENDIDIKAN

SD : SDN 6 Rantau Utara TAMAT: 2015
SMP : SMPN 3 Rantau Utara TAMAT: 2018
SMA : SMAN 1 Rantau Utara TAMAT: 2021

KATA PENGANTAR



Alhamdulillah, segala puji dan syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan karunianya, sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul tentang “KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN ALGORITMA K-MEANS DAN DBSCAN” untuk memenuhi persyaratan dalam jenjang strata satu dan mencapai gelar sarjana Komputer di jurusan Teknologi Informasi dan Ilmu Komputer, Universitas Muhammadiyah Sumatera Utara. Shalawat serta salam selalu tercurahkan kepada junjungan Nabi besar Muhammad SAW, keluarga dan sahabatnya yang syafaatnya kita nantikan diakhir zaman nanti. Dalam penyusunan Tugas Akhir ini, penulis telah mendapatkan banyak bantuan dan bimbingan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis tentunya berterimakasih kepada pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga berterimakasih kepada:

1. Bapak Prof. Dr. Akrim, M.Pd. selaku Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU serta Pembimbing Skripsi penulis. Terima kasih atas kesabaran, arahan, serta waktu yang telah diberikan. Setiap bimbingan yang diberikan bukan hanya membantu menyelesaikan skripsi ini, tetapi juga membentuk cara berpikir penulis menjadi lebih baik.
3. Ibu Dr. Firahmi Rizky, S.Kom.,M.Kom. selaku Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
4. Bapak Mhd. Basri, S.Si, M.Kom. selaku Wakil Dekan III Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
5. Ibu Fatma Sari Hutagalung S.Kom., M.Kom. selaku Ketua Program Studi Teknologi Informasi.

6. Bapak Okvi Nugroho, S.Kom., M.Kom selaku Sekretaris Program Studi Teknologi Informasi serta Pembimbing Akademik penulis. Terima kasih telah menjadi sosok pembimbing yang tidak hanya membimbing secara akademik, tetapi juga memberikan perhatian tulus kepada penulis sebagai manusia yang sedang belajar dan bertumbuh.
7. Kepada nenek tersayang, Ibu H. Nasution, alasan penulis untuk sukses. Terima kasih atas cinta yang tulus, doa yang tak pernah putus, dan kehadiranmu yang selalu menjadi tempat paling nyaman bagi penulis. Nenek bukan hanya sosok yang penulis hormati, tetapi juga teman paling menyenangkan yang pernah penulis miliki.
8. Kepada papa dan mama tersayang, Bapak D.Sinukaban dan Ibu D.Tarigan. Terima kasih yaitu dua kata yang sederhana, namun menyimpan seluruh rasa yang tak mampu penulis ungkapkan. Untuk wanita hebatku, terima kasih karena selalu menjadi sosok inspirasi penulis untuk menjadi penyabar dan setia. Sosok yang tak pernah lelah menguatkan, meski sering kali menyembunyikan lelah dan air mata sendiri. Dan untuk lelaki kuatku, terima kasih karena telah menjadi pelindung dan panutan terbaik dalam hidup penulis. Bahumu mungkin lelah menopang begitu banyak beban, tetapi cinta dan perjuanganmu tidak pernah sedikit pun berkurang. Terima kasih telah memilih untuk bertahan, berjuang, dan berkorban tanpa pernah meminta imbalan apa pun. Setiap langkah yang penulis tempuh hari ini adalah buah dari keringat dan doa kalian yang tak pernah berhenti mengalir. Semoga pencapaian kecil ini mampu membuat senyum indah bahagia diwajah kalian.
9. Kepada adik-adik kecilku tersayang, Queen YDS, N. Gadis DS, Quisha ADS, dan yang paling lucu dan imut M. Yumna DS. Terima kasih, mungkin penulis tidak pernah mengungkapkan secara langsung, penulis merasa kehadiran kalian adalah semangat yang sesungguhnya. Tingkah kalian yang lucu selalu menjadi alasan penulis rindu pulang dan kuat untuk terus berjuang. Rasa sayang penulis kepada kalian jauh lebih besar dari yang pernah penulis tunjukkan.

10. Kepada bunda dombat dan bunda bebek, terima kasih sudah menjadi sosok bunda sekaligus orang tua penulis di saat penulis jauh dari orangtua. Terima kasih atas setiap sikap excited kalian dalam mendukung penulis dalam hal apapun. Kalian juga adalah sosok yang selalu mengusahakan memenuhi setiap keinginan penulis, sejak kecil hingga sampai saat ini, tanpa pernah membuat penulis merasa kekurangan kasih sayang. Penulis sangat bersyukur memiliki wanita hebat seperti kalian.
11. Kepada LSH teman yang dengan tulus membimbing penulis. Keikhlasanmu dalam meluangkan waktu, berbagi ilmu, dan memberikan semangat adalah sesuatu yang sangat berharga dan tidak akan pernah penulis lupakan. Terima kasih telah hadir sebagai sosok yang mengajarkan bahwa kepedulian yang tulus jauh lebih bermakna dari sekadar kewajiban.
12. Kepada Sahabat terdekat, Tika Ariyani, Mentari merona tarigan, dan Cindy Purnama Sari. Terima kasih telah menjadi teman yang bersama penulis sejak lama, yang selalu memberi dukungan dan doa pada penulis.
13. Kepada temen seperjuangan, Nadya Aulya Putri, Tiara Amanda Lestari, Siti Nurisma Siregar, Nabila Azura Putri, Muhammad Ariq Adrian, dan teman-teman lainnya. Terima kasih telah menjadi bagian dari perjalanan yang tidak selalu mudah ini. Semua momen yang kita lalui bersama adalah kenangan yang tidak akan pernah terhapus dari ingatan penulis. Semoga kita semua meraih impian yang selama ini kita perjuangkan bersama.
14. Kepada Gilang R.R, yang selama ini menjadi tempat bersandar, penyemangat, dan pengingat bahwa penulis mampu ketika penulis sendiri mulai meragukannya. Terima kasih telah sabar terhadap tingkah dan perilaku penulis, atas setiap doa yang kamu panjatkan, atas candaan lucu yang kamu berikan, atas pengorbanan yang kamu lakukan, dan atas kepercayaan yang tidak pernah goyah.
15. Kepada diri sendiri, terima kasih telah memilih untuk tidak menyerah di hari-hari ketika segalanya terasa berat dan mustahil. Terima kasih telah berdamai dengan keterbatasan, bangkit dari setiap keraguan, dan terus melangkah meskipun langkah itu terasa goyah. Ini adalah bukti bahwa kamu lebih kuat dari yang kamu kira.

16. Semua pihak yang terlibat langsung ataupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.

KLASTERISASI DAERAH RAWAN KRIMINALITAS DI MEDAN TIMUR BERDASARKAN DATA KEPOLISIAN MENGGUNAKAN ALGORITMA *K-MEANS* DAN *DBSCAN*

ABSTRAK

Kriminalitas merupakan permasalahan sosial yang serius, khususnya di Kecamatan Medan Timur, Kota Medan. Penelitian ini bertujuan untuk mengklusterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan data kepolisian dari Polsek Medan Timur tahun 2023–2025 menggunakan algoritma *K-Means* dan *DBSCAN* (*Density-Based Spatial Clustering of Applications with Noise*), serta membandingkan performa keduanya menggunakan metrik evaluasi *Silhouette Index* dan *Davies-Bouldin Index*. Data yang digunakan mencakup jenis kriminalitas pencurian, yakni pencurian kendaraan bermotor (*curanmor*), pencurian dengan kekerasan (*curas*), dan pencurian dengan pemberatan (*curat*) yang tersebar di 20 kelurahan. Tahapan penelitian meliputi *preprocessing* data, *feature engineering*, normalisasi Min-Max, implementasi algoritma, serta evaluasi kualitas kluster. Hasil penelitian menunjukkan bahwa algoritma *K-Means* dengan $K=3$ menghasilkan tiga kluster yang mencakup seluruh 20 kelurahan dengan *Silhouette Index* sebesar 0.4128 (kategori lemah). Sementara itu, algoritma *DBSCAN* dengan parameter $\text{eps}=0.20$ dan $\text{minPts}=2$ menghasilkan dua kluster dari 9 kelurahan dengan *Silhouette Index* sebesar 0.5173 (kategori sedang). Evaluasi tambahan menggunakan *Davies-Bouldin Index* (DBI) menghasilkan nilai DBI *K-Means* sebesar 0.835 dan DBI *DBSCAN* sebesar 0.5976, di mana nilai DBI yang lebih rendah pada *DBSCAN* mengonfirmasi kualitas kluster yang lebih baik. Berdasarkan evaluasi tersebut, *DBSCAN* menghasilkan kualitas kluster yang lebih baik dibandingkan *K-Means*, meskipun *K-Means* lebih unggul dalam kelengkapan pengelompokan seluruh wilayah. Hasil pemetaan ini diharapkan dapat menjadi rekomendasi bagi pihak kepolisian dalam merancang strategi pencegahan kriminalitas yang lebih efektif dan efisien di wilayah Medan Timur.

Kata Kunci: Kriminalitas; Klusterisasi; *K-Means*; *DBSCAN*; *Silhouette Index*; *Davies-Bouldin Index*; Medan Timur

CLUSTERING OF CRIME-PRONE AREAS IN EAST MEDAN BASED ON POLICE DATA USING K-MEANS AND DBSCAN ALGORITHMS

ABSTRACT

Crime is a serious social problem, particularly in the Medan Timur sub-district, Medan City. This study aims to cluster crime-prone areas in Medan Timur based on police data from the Medan Timur Police Sector for the period 2023–2025 using the K-Means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithms, and to compare their performance using the Silhouette Index and Davies-Bouldin Index evaluation metrics. The data used covers theft-related crimes, including motorcycle theft (curanmor), robbery (curas), and aggravated theft (curat) across 20 sub-villages. The research stages include data preprocessing, feature engineering, Min-Max normalization, algorithm implementation, and cluster quality evaluation. The results show that the K-Means algorithm with $K=3$ produces three clusters covering all 20 sub-villages with a Silhouette Index of 0.4105 (weak category) and Davies-Bouldin Index of 0.835. Meanwhile, the DBSCAN algorithm with parameters $\text{eps}=0.20$ and $\text{minPts}=5$ produces two clusters from 14 sub-villages with a Silhouette Index of 0.5173 (moderate category) and Davies-Bouldin Index of 0.5976. Based on this evaluation, DBSCAN produces better cluster quality than K-Means in terms of both metrics, although K-Means is superior in terms of complete coverage of all areas. The results of this mapping are expected to serve as recommendations for the police in designing more effective and efficient crime prevention strategies in the Medan Timur area.

Keywords: *Crime; Clustering; K-Means; DBSCAN; Silhouette Index; Medan Timur*

DAFTAR ISI

LEMBAR PENGESAHAN	i
PERNYATAAN ORISINALITAS.....	ii
PERNYATAAN PERSETUJUAN PUBLIKASI.....	iii
RIWAYAT HIDUP	iv
KATA PENGANTAR.....	v
ABSTRAK	ix
ABSTRACT	x
DAFTAR ISI.....	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
BAB II LANDASAN TEORI	6
2.1. Kriminalitas.....	6
2.2. Data Mining	6
2.2.1 Tahapan Data Mining.....	8
2.2.2 Teknik - Teknik Data Minig	10
2.3. <i>Clustering</i>	12
2.4. <i>K-Means</i>	13
2.5. <i>DBSCAN (Density-Based Spatial Clustering of Applications with Noise)</i> 17	
2.6. Ringkasan penelitian terdahulu	18
2.7. Analisis GAP.....	21
BAB III METODOLOGI PENELITIAN	24
3.1 Jenis penelitian	24
3.2 Teknik pengumpulan data	24
3.3 Arsitektur Penelitian.....	26
BAB IV HASIL DAN PEMBAHASAN	33
4.1. Deskripsi Data.....	33
4.2. <i>Preprocessing Data</i>	35
4.2.1. <i>Data Cleaning</i>	36
4.2.2. Distribusi Data	42
4.2.3. <i>Featuring Engineering dan Agrerasi Data</i>	49
4.2.4. Normalisasi Data (<i>Min-Max Normalization</i>)	52
4.3. Implementasi Algoritma K-Means.....	59
4.3.1 Penentuan WSSE dan membuat (Metode elbow).....	59
4.3.2 Perhitungan Manual K-Means	63
4.4. Implementasi Algoritma DBSCAN	69
4.4.1. Penentuan parameter DBSCAN.....	69
4.4.2. Perhitungan Manual DBSCAN	73

4.5.	Perbandingan K-Means dan DBSCAN.....	82
4.5.1.	Karakteristik Kedua Algoritma.....	82
4.5.2.	Hasil Klasterisasi Algoritma	83
4.5.3.	Evaluasi Kualitas Klaster Menggunakan Silhouette Index dan Davies-Bouldin Index	86
4.5.4.	Perbandingan hasil evaluasi klaster	89
4.5.5.	Kesimpulan Perbandingan Algoritma	92
4.5.6.	Hasil Implementasi Website	93
BAB V	KESIMPULAN DAN SARAN	103
5.1.	Kesimpulan	103
5.2.	Saran.....	104
DAFTAR PUSTAKA		105

DAFTAR TABEL

Tabel 2.1 Ringkasan Terdahulu	19
Tabel 2.2 Data Medan Kriminalitas Polsek Medan Timur	25
Tabel 4.1 Data Kriminalitas Sebelum Diolah	33
Tabel 4.2 Fitur Dataset yang Digunakan.....	34
Tabel 4.3 Atribut yang Digunakan.....	34
Tabel 4. 4 <i>Missing Values</i> dan Metode Penanganan.....	38
Tabel 4.5 Tahapan <i>Data Cleaning</i>	41
Tabel 4. 6 Hasil Analisis Distribusi Data.....	48
Tabel 4.7 Fitur Yang Digunakan Setelah <i>Cleaning Data</i>	49
Tabel 4.8 Hasil Agregrasi Data per Kelurahan	50
Tabel 4.9 Matriks Korelasi Pearson Antar Fitur	51
Tabel 4.10 Nilai Min, Max, dan Range Setiap Fitur.....	52
Tabel 4.11 Normalisasi total_kriminal.....	53
Tabel 4.12 Normalisasi Curanmor	54
Tabel 4.13 Normalisasi Curas	55
Tabel 4.14 Normalisasi Curat	55
Tabel 4.15 Normalisasi Rata_waktu	56
Tabel 4.16 Normalisasi MinMax Keseluruhan Fitur	57
Tabel 4.17 Output hitung WSSE.....	60
Tabel 4.18 Hasil jarak penentuan WSSE	61
Tabel 4.19 Hasil Iterasi 1	64
Tabel 4.20 Hasil Iterasi 2	65
Tabel 4.21 Hasil Akhir K-Means	66
Tabel 4.22 Aturan Baku DBSCAN.....	69
Tabel 4.23 $k = \text{minPts}$	71
Tabel 4.24 Hasil SI dari eps dan minPts	72
Tabel 4.25 Parameter DBSCAN	73
Tabel 4.26 Jenis Titik parameter.....	73
Tabel 4.27 d(Durian, Sidorame Barat I).....	74
Tabel 4.28 d(Durian, Sidorame Timur).....	75
Tabel 4.29 d(Pahlawan, Pandau Hilir)	75
Tabel 4.30 d(Pahlawan, Sei Kera Hulu).....	76
Tabel 4.31 d(Durian, Gaharu)	76
Tabel 4.32 Status Titik Tetangga	76
Tabel 4.33 Proses Pembentukan Kluster DBSCAN	77
Tabel 4.34 Hasil Akhir Klasterisasi DBSCAN	78
Tabel 4.35 Ringkasan Hasil DBSCAN	79
Tabel 4.36 Karakteristik Algoritma	82
Tabel 4. 37 Hasil Klasterisasi K-means	83
Tabel 4.38 Hasil Klasterisasi DBSCAN	85
Tabel 4.39 Interpretasi sillhoutte index.....	87
Tabel 4.40 Nilai SI K-Means	87
Tabel 4.41 Nilai SI DBSCAN.....	89
Tabel 4.42 Perbandingan Hasil Evaluasi	89

DAFTAR GAMBAR

Gambar 2. 1 Tahapan data mining menggunakan <i>CRISP-DM</i>	8
Gambar 3.1 Arsitektur Penelitian	26
Gambar 3. 2 Flowchart K-Means	29
Gambar 3. 3 Flowchart DBSCAN	31
Gambar 4.1 <i>Code Python Load Data</i>	37
Gambar 4.2 <i>Code Python Penghapusan Kolom</i>	37
Gambar 4.3 <i>Code Python Penanganan Missing Values</i>	38
Gambar 4.4 <i>Code Python Pembersihan Whitespace</i>	39
Gambar 4.5 <i>Code Python Koreksi Typo Jenis Kriminal</i>	39
Gambar 4.6 <i>Code Python Koreksi Anomali Tahun</i>	40
Gambar 4.7 <i>Code Python Konversi Tipe data</i>	40
Gambar 4. 8 <i>Code Python Standarisasi Format Jam</i>	41
Gambar 4.9 <i>Code Python Penghapusan Data Duplikat</i>	41
Gambar 4.10 Grafik Distribusi Jenis Kriminal	43
Gambar 4.11 Grafik Distribusi per Kecamatan	43
Gambar 4.12 Grafik Distribusi per Hari	44
Gambar 4.13 Grafik Distribusi per Bulan	45
Gambar 4.14 Grafik Distribusi per Tahun	46
Gambar 4.15 Grafik Distribusi per Kategori_waktu	46
Gambar 4.16 Grafik 10 Kelurahan dengan Kasus Terbanyak	47
Gambar 4.17 Heatmap Jenis Kriminal VS Kategori Waktu	48
Gambar 4.18 Program Python distribusi grafik	49
Gambar 4.19 <i>Code python feature engineering dan agregasi</i>	50
Gambar 4. 20 <i>Code Python Analisis Korelasi</i>	51
Gambar 4.21 <i>Code Python Normalisasi Min-Max (tanpa library)</i>	53
Gambar 4.22 Perbandingan Data Sebelum Dan Sesudah Normalisasi In-Max ...	58
Gambar 4.23 Hitung WSSE	60
Gambar 4.24 Hitungan Manual WSSE	61
Gambar 4.25 Metode Elbow untuk nilai K	62
Gambar 4. 26 Perhitungan Manual K-Means	63
Gambar 4.27 Hasil klasterisasi K-Means	68
Gambar 4.28 Hasil klasterisasi berdasarkan total kriminal	68
Gambar 4.29 <i>Code Python minPts</i>	70
Gambar 4.30 Output minPts	70
Gambar 4.31 Penentuan Parameter DBSCAN	72
Gambar 4.32 Grafik Elbow nilai eps	74
Gambar 4.33 Visualisasi klasterisasi DBSCAN	81
Gambar 4.34 Hasil Visualisasi DBSCAN lanjutan	82
Gambar 4.35 Visualisasi SI K-Means dan DBSCAN	91
Gambar 4.36 Perbandingan SI	91
Gambar 4.37 Tampilan Halaman Beranda	93
Gambar 4.38 Tampilan Ringkasan Data pada Halaman Beranda	94
Gambar 4.39 Tampilan Halaman Distribusi	95
Gambar 4.40 Tampilan Halaman Distribusi Lanjutan	95
Gambar 4.41 Hasil K-Means pada Halaman Clustering	96

Gambar 4.42 Hasil K-Means pada Halaman Clustering lanjutan	96
Gambar 4.43 Hasil K-Means pada Halaman Clustering lanjutan	97
Gambar 4.44 Hasil DBSCAN pada Halaman Clustering	98
Gambar 4.45 Hasil DBSCAN pada Halaman Clustering lanjutan	98
Gambar 4.46 Hasil DBSCAN pada Halaman Clustering lanjutan	99
Gambar 4.47 Hasil visualisasi 100	
Gambar 4.48 Hasil visualisasi lanjutan	100
Gambar 4.49 Tampilan Halaman Perbandingan	101
Gambar 4.50 Tampilan Halaman Perbandingan lanjutan	102
Gambar 4.51 Tampilan Halaman Perbandingan lanjutan	102

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Kriminalitas adalah setiap dan semua perbuatan yang berdampak negatif secara ekonomi dan psikologis serta bertentangan dengan norma sosial dan agama serta hukum Indonesia (Area, 2023). Tingginya angka kriminalitas dapat mengancam kesejahteraan dan ketentraman Masyarakat yang kemungkinan akan terkena dampak negatif jika anggota masyarakat percaya bahwa keselamatan mereka terancam (Suriani, 2020). Kriminalitas menjadi permasalahan sosial yang serius, terutama di Kota Medan (Maret et al., 2024). Medan Timur merupakan salah satu kecamatan di Kota Medan yang juga memiliki potensi terjadinya tindak kriminal. Tingkat kriminal yang semakin tinggi tidak hanya rugi material, tetapi juga menciptakan rasa tidak aman di kalangan masyarakat.

Oleh karena itu, diperlukan upaya pencegahan yang efektif yaitu dengan mengidentifikasi dan memetakan daerah-daerah rawan kriminalitas secara konkret (Herlina & Ernarningsih, 2023). Pemetaan ini dapat membantu pihak kepolisian dalam meningkatkan patroli di wilayah-wilayah rawan, merancang strategi pencegahan kejahatan yang tepat sasaran, dan mengalokasikan sumber daya lebih optimal.

Proses identifikasi/pengelompokkan daerah rawan kriminalitas di Kota Medan Timur dapat dilakukan dengan teknik data mining seperti klasterisasi (*clustering*) menggunakan algoritma *K-Means* dan *DBSCAN*. Klasterisasi merupakan metode dalam *machine learning* untuk mengelompokkan data ke dalam

cluster dengan kesamaan dan karakteristik diantara objek data tersebut yang bekerja dengan teknik *unsupervised learning* (Mochurad et al., 2023).

K-Means merupakan algoritma klusterisasi yang sederhana dan efisien untuk mengelompokkan data berdasarkan jarak terdekat ke pusat kluster (*centroid*), dimana algoritma ini bekerja dengan cara membagi data dalam sejumlah kluster yang telah ditentukan sebelumnya (k), dan mengoptimalkan posisi *centroid* dengan iteratif (Afdal et al., 2024). Sedangkan, *DBSCAN* adalah algoritma berbasis kepadatan yang mampu menemukan kluster dengan bentuk *arbitrary* dan dapat mengidentifikasi *noise* atau *outlier* dalam data, dimana *DBSCAN* bergantung pada parameter *epsilon* (ϵ) dan *minimum points* (minPts) (Tambunan et al., 2025).

Penelitian yang dilakukan oleh Rimelda Adha dan kawan kawan dengan judul “Perbandingan Algoritma *DBSCAN* dan *K-Means Clustering* untuk Pengelompokan Kasus Covid-19 di Dunia” disimpulkan bahwa *K-Means* lebih unggul dibandingkan dengan *DBSCAN* dalam mengklusterisasikan kasus Covid-19, dengan *K-Means* yang memiliki nilai SI terbaik sebesar 0,6902 pada nilai $k = 8$ (Adha et al., 2021).

Dalam studi yang dilakukan oleh Tri Amri Wijaya dan kawan kawan dengan judul “Perbandingan Algoritma *DBSCAN* dan *K-Means Clustering* untuk Pengelompokan Data Gangguan PT. PLN UID Kalselteng” menunjukkan bahwa *K-Means* lebih unggul daripada *DBSCAN* dalam mengelompokkan data gangguan, dengan nilai SI terbaik *K-Means* sebesar 0,581 yang terletak pada nilai $k = 6$ (Wijaya et al., 2024).

Walaupun perbandingan antara *K-Means* dan *DBSCAN* telah dilakukan pada penelitian sebelumnya, namun penelitian tersebut dilakukan dalam konteks

dan dataset yang berbeda, seperti data kesehatan (Covid-19) dan data gangguan listrik. Dengan dataset yang berbeda memiliki hasil yang berbeda pula, terdapat *K-Means* lebih unggul dari penelitian sebelumnya, dan ada pula penelitian yang menyatakan *DBSCAN* lebih unggul. Oleh karena itu, penelitian ini bertujuan untuk membandingkan serupa dengan konteks dataset yang berbeda yaitu data kriminal yang dapat menghasilkan perspektif baru dalam evaluasi performa kedua algoritma tersebut.

Penelitian ini bertujuan untuk membandingkan algoritma *K-Means* dan *DBSCAN* dalam mengelompokkan atau mengklusterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan data kepolisian. Dengan dataset kriminalitas yang memiliki karakteristik yang berbeda, penelitian ini diharapkan dapat memberikan perspektif baru dalam mengevaluasi performa kedua algoritma tersebut. Hasil penelitian ini diharapkan dapat menjadi rekomendasi bagi pihak kepolisian dalam menentukan metode klusterisasi yang paling baik dalam memetakan daerah rawan kriminalitas, sehingga dapat mengambil keputusan yang lebih efektif dan efisien dalam upaya pencegahan kriminalitas di Medan Timur. Selain itu, hasil pemetaan tersebut juga dapat membantu masyarakat khususnya pengemudi ojek online yang sering melayani pelanggan di Wilayah Medan Timur, sehingga pengemudi dapat lebih waspada saat melewati daerah yang rawan terhadap kriminalitas yang tinggi.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana melakukan klasterisasi daerah rawan kriminalitas di Medan Timur menggunakan algoritma *K-Means* berdasarkan data kepolisian?
2. Bagaimana melakukan klasterisasi daerah rawan kriminalitas di Medan Timur menggunakan algoritma *DBSCAN* berdasarkan data kepolisian?
3. Bagaimana perbandingan performa algoritma *K-Means* dan *DBSCAN* dalam mengklasterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan metrik evaluasi *Silhouette Index* dan *Davies-Bouldin Index*?
4. Algoritma manakah yang lebih optimal untuk klasterisasi daerah rawan kriminalitas di Medan Timur antara *K-Means* dan *DBSCAN*?

1.3 Batasan Masalah

Penelitian ini memiliki batasan permasalahan yang dibahas agar dapat menyelesaikan permasalahan utama. Berikut ini batasan masalah penelitian ini:

1. Penelitian ini hanya dilakukan di wilayah Kecamatan Medan Timur, Kota Medan, Sumatera Utara.
2. Data yang digunakan adalah data kriminalitas dari Kepolisian Sektor Medan Timur dengan periode waktu 2023-2025.
3. Metode yang digunakan adalah klasterisasi (*unsupervised learning*) dengan algoritma *K-Means* dan *DBSCAN*.
4. Penelitian ini hanya membandingkan performa algoritma menggunakan *Silhouette Index* dan *Davies-Bouldin Index*.
5. Jenis kriminalitas yang dianalisis meliputi pencurian, baik itu pencurian motor (curanmor), pencurian dengan kekerasan (curas), dan pencurian dengan pemberatan (curat).

6. Penelitian ini hanya fokus pada klasterisasi *spasial* (berdasarkan lokasi) daerah rawan kriminalitas, tidak mencakup analisis prediksi atau *forecasting* kejahatan di masa mendatang.
7. Implementasi algoritma menggunakan bahasa pemrograman python.
8. Penelitian ini tidak membahas faktor-faktor penyebab kriminalitas, melainkan fokus pada pengelompokan daerah berdasarkan tingkat kerawanan kriminalitas.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, tujuan dari penelitian ini adalah:

1. Untuk mengimplementasikan algoritma *K-Means* dalam mengklusterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan data kepolisian.
2. Untuk mengimplementasikan algoritma *DBSCAN* dalam mengklusterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan data kepolisian.
3. Untuk menganalisis dan membandingkan performa algoritma *K-Means* dan *DBSCAN* dalam mengklusterisasikan daerah rawan kriminalitas di Medan Timur berdasarkan metrik evaluasi *Silhouette Index* dan *Davies-Bouldin Index*.
4. Untuk menentukan algoritma yang lebih optimal antara *K-Means* dan *DBSCAN* dalam klasterisasi daerah rawan kriminalitas di Medan Timur berdasarkan hasil evaluasi performa.

1.5 Manfaat Penelitian

Adapun manfaat penelitian yang akan dihasilkan dalam peneliti ini adalah sebagai berikut:

1. Dapat menjadi bahan rekomendasi bagi pihak kepolisian dalam menentukan wilayah prioritas patroli dan strategi pencegahan kriminalitas.
2. Dapat membantu meningkatkan kewaspadaan masyarakat dalam beraktivitas di Wilayah Medan Timur.
3. Dapat memberikan gambaran perbandingan performa algoritma *K-Means* dan *DBSCAN* pada data kriminalitas.
4. Dapat memberikan kontribusi ilmiah dalam penerapan metode klasterisasi pada data kriminalitas dengan menerapkan algoritma *K-Means* dan *DBSCAN*.

BAB II

LANDASAN TEORI

2.1. Kriminalitas

Kriminalitas merupakan suatu tindakan atau perilaku jahat yang dilakukan oleh 1 orang atau lebih. Kejahatan adalah perilaku atau perbuatan yang disengaja dengan melanggar norma maupun hukum dan perilaku tersebut diancam dengan hukuman. Secara Yuridis, kriminalitas adalah tingkah laku yang melanggar hukum dan undang – undang dengan ancaman sanksi pidana pada pelanggarnya (Azhari et al., 2024).

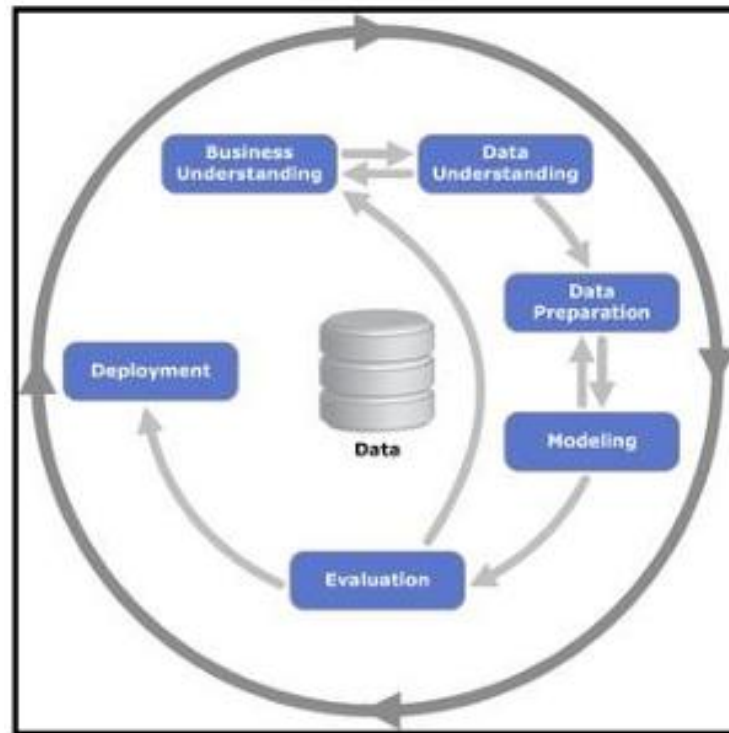
Tindakan kriminal mungkin disebabkan oleh beberapa faktor kondisi, seperti faktor kemiskinan sehingga membuat pelaku merasa kurang akan material sehingga menghalalkan segala cara, faktor kondisi mental misalnya kebencian yang mendalam terhadap korban, dan masih banyak faktor lain yang dapat mendukung tindakan kriminal (Rahmadayanti et al., 2023).

2.2. Data Mining

Data mining merupakan proses ekstraksi informasi dari data yang bervolume besar, yang dapat menganalisis data yang besar menjadi pola untuk pengambilan keputusan yang tersimpan didalam suatu basis data (Suriani, 2020). Data mining merupakan salah satu teknik otomatis yang mencari hubungan antar set data yang besar dan kompleks. Data mining juga dapat melakukan pengelompokkan, untuk mengetahui pola *universal* dari data data yang ada (Rahmadayanti et al., 2023).

2.2.1 Tahapan Data Mining

Adapun tahapan dari data mining dengan menggunakan metodologi *CRISP-DM* (*Cross-Industry Standard Process for Data Mining*) seperti pada gambar berikut:



Gambar 2. 1 Tahapan data mining menggunakan *CRISP-DM*

Sumber: (Ramadani et al., 2025)

Proses data mining terdiri dari beberapa tahap yang biasanya mengikuti metodologi *CRISP-DM* (*Cross-Industry Standard Process for Data Mining*), yaitu (Ramadani et al., 2025):

1. *Business Understanding*

Proses ini melakukan analisa latar belakang pada klusterisasi data kriminalitas untuk mengidentifikasi permasalahan yang ada. Selanjutnya digunakan untuk menentukan tujuan dan perencanaan strategi dalam penerapan proses pemodelan data mining.

2. *Data Understanding*

Proses ini mempersiapkan data yang diterima dalam bentuk excel yang data tersebut perlu dibersihkan dan melakukan pemilihan data dengan atributnya. Data tersebut didapat langsung dari pihak Kepolisian Sektor (Polsek) Kecamatan Medan Timur yaitu data kriminalitas selama 3 tahun, yaitu 2023-2025 dengan setiap wilayah yang akan diklasterisasikan kedalam masing-masing klaster.

3. *Data Preparation*

Proses ini melakukan seleksi data dengan mengelompokkan data kedalam beberapa kelompok yang telah ditentukan, dan diolah sesuai dengan keperluan pemodelan data.

4. *Modeling*

Pada proses ini, melakukan penerapan pada algoritma yang sesuai untuk mencapai tujuan penelitian. Teknik yang digunakan dalam analisis data di penelitian ini adalah *K-Means* dan *DBSCAN*, yang bertujuan untuk mengelompokkan daerah rawan kriminalitas di Medan Timur tahun 2023-2025.

5. *Evaluation*

Proses ini bertujuan untuk menilai kualitas model yang dibangun. Beberapa metode dalam evaluasi *clustering* adalah *Silhouette Score*.

6. *Deployment*

Pada proses ini, hasil penelitian disajikan dalam bentuk laporan atau visualisasi yang dapat dilihat dan diperbaharui secara berkala dengan data terbaru untuk pencegahan dan penanggulangan tindak kriminal.

2.2.2 Teknik - Teknik Data Mining

Teknik data mining merupakan metode yang digunakan untuk menggali pola atau informasi dari data dalam jumlah besar. Pemilihan teknik yang tepat, sangat bergantung pada tujuan analisis serta karakteristik data yang digunakan. Adapun teknik utama dalam data mining sebagai berikut:

1. Klasifikasi (*Classification*)

Klasifikasi adalah teknik data mining yang digunakan untuk mengelompokkan data ke dalam kelas atau kategori tertentu berdasarkan atribut yang dimiliki. Teknik ini termasuk dalam *supervised learning* karena menggunakan data yang telah memiliki label. Metode klasifikasi bertujuan untuk membangun model yang dapat digunakan untuk memprediksi kelas dari data baru. Contoh algoritma yang sering digunakan dalam klasifikasi antara lain *Decision Tree*, *Naive Bayes*, dan *Support Vector Machine (SVM)*.

Contoh penerapan klasifikasi adalah penentuan apakah suatu transaksi termasuk penipuan atau tidak, serta klasifikasi tingkat kriminalitas berdasarkan kategori tertentu.

2. Klusterisasi (*Clustering*)

Clustering adalah teknik untuk mengelompokkan data berdasarkan tingkat kemiripan tanpa menggunakan label (*unsupervised learning*). Data yang memiliki karakteristik mirip akan dikelompokkan ke dalam satu cluster, sedangkan data yang berbeda akan berada di cluster lain. Tujuan utama *clustering* adalah untuk menemukan struktur alami dalam data. Beberapa algoritma yang umum digunakan adalah *K-Means*, *DBSCAN*, dan *Hierarchical Clustering*.

Clustering sering digunakan dalam segmentasi pasar, analisis wilayah, serta pengelompokan daerah rawan kejahatan.

3. Asosiasi (*Association Rule Mining*)

Teknik asosiasi digunakan untuk menemukan hubungan atau keterkaitan antar item dalam suatu dataset. Teknik ini sering digunakan untuk mengetahui pola keterkaitan antar kejadian. Salah satu algoritma yang populer dalam teknik ini adalah *Apriori* dan *FP-Growth*. Hasil dari *association rule* biasanya berupa aturan dalam bentuk "jika-maka" (*if-then*).

Contoh penerapan adalah dalam analisis keranjang belanja (*market basket analysis*), misalnya jika seseorang membeli roti, maka kemungkinan besar juga membeli susu.

4. Regresi (*Regression*)

Regresi adalah teknik data mining yang digunakan untuk memprediksi nilai numerik atau *kontinu* berdasarkan hubungan antar variabel. Teknik ini juga termasuk dalam *supervised learning*. Regresi sering digunakan untuk melakukan prediksi, seperti prediksi jumlah penjualan, harga rumah, atau tingkat kriminalitas di masa mendatang. Algoritma yang sering digunakan antara lain *Linear Regression* dan *Polynomial Regression*.

5. Deteksi Anomali (*Anomaly Detection*)

Deteksi anomali merupakan teknik untuk menemukan data yang menyimpang atau tidak sesuai dengan pola umum dalam dataset. Data yang menyimpang ini sering disebut sebagai *outlier*. Teknik ini banyak digunakan dalam bidang keamanan, seperti deteksi *fraud* (penipuan), serta dalam sistem monitoring untuk mendeteksi aktivitas yang tidak normal.

6. Prediksi (*Prediction*)

Teknik prediksi digunakan untuk memperkirakan kejadian di masa depan berdasarkan pola data yang telah ada. Teknik ini sering dikombinasikan dengan metode klasifikasi atau regresi. Contoh penerapannya adalah prediksi jumlah kasus kejahatan, prediksi permintaan produk, dan prediksi cuaca.

2.3. *Clustering*

Clustering atau klasterisasi adalah salah satu teknik yang termasuk dalam fungsionalitas data mining (Kartikasari, 2021). Klasterisasi bertujuan untuk mengelompokkan sejumlah data menjadi kelompok data yang memiliki kesamaan tertentu (*cluster*) (Elda et al., 2021).

Klasterisasi dapat diartikan sebagai metode yang membagi-bagi data yang diimplementasikan di berbagai bidang. Dalam melakukan klasterisasi tidak ada target variable yang ditentukan sebelumnya, sehingga algoritma harus menemukan struktur tersembunyi dalam data (Thakur, 2020).

Berdasarkan metodologi yang digunakan, *clustering* memiliki beberapa kategori seperti berikut (Anfelicia et al., 2024):

1. *Partitioning methods*

Membagi data kedalam k partisi *non-overlapping*, dan setiap objek hanya menjadi anggota satu *cluster*, seperti *cluster* menggunakan *K-Means* dan *K-Medoids*.

2. *Hierarchical methods*

Membentuk hirarki *cluster* dalam bentuk *tree structure*, seperti *Hierarchical Clustering*, *Divisive*.

3. *Density-based methods*

Mengelompokkan objek berdasarkan kepadatan area, dan dapat menemukan *cluster* dengan bentuk *arbitrary*, seperti *DBSCAN* dan *Optics*.

4. *Grid-based methods*

Membagi ruang data ke dalam *grid cells*, dan klasterisasi dilakukan di dalam *grid cells*, seperti *Sting* dan *Clique*.

5. *Model-based methods*

Mengasumsikan ruang data kedalam *grid cells* dan *clustering* dilakukan pada *grid cells*, seperti *Gaussian Mixture Models*.

2.4. *K-Means*

K-Means adalah salah satu algoritma clustering yang paling populer dan banyak digunakan. *K-means* diperkenalkan oleh MacQueen pertama kali (1967) dan telah menjadi standar untuk *many clustering applications* karena kesederhanaan dan efisiensinya (Furqan et al., 2022). Adapun beberapa tahap dalam algoritma *K-Means* yang dilakukan sebagai berikut:

1. *Preprocessing*

Tahap pertama sebelum mengimplementasikan algoritma *K-Means* adalah melakukan *preprocessing data* guna memperoleh hasil klasterisasi yang maksimal (Thakur, 2020). Tahapan *preprocessing* mencakup berbagai proses seperti seleksi, transformasi, dan normalisasi data, yang disesuaikan dengan kebutuhan serta karakteristik algoritma yang diterapkan (Mochurad et al., 2023). Dalam penelitian ini, normalisasi data dilakukan menggunakan teknik *MinMax*, yaitu teknik yang bertujuan untuk menyamakan skala data,

khususnya antara nilai-nilai yang memiliki rentang besar dan kecil (Technology, 2025). Proses ini sangat penting karena disparitas skala dapat mengakibatkan bias dalam proses klasifikasi. Melalui teknik normalisasi *MinMax* yang mana seluruh nilai akan diselaraskan ke dalam rentang yang seragam sehingga perhitungan menjadi lebih presisi seperti pada persamaan berikut.

$$x' = \frac{x - \text{nilai min}}{\text{nilai max} - \text{nilai min}} \dots \dots \dots (2.1)$$

Keterangan:

x' = data per kolom

nilai min = nilai terkecil per kolom

nilai max = nilai terbesar per kolom

2. Menentukan jumlah kluster (nilai K) dan *centroid* awal

Tahap berikutnya adalah menetapkan jumlah kluster (K) dan menentukan titik pusat awal (*centroid*) untuk masing-masing kluster (Rahmadayanti et al., 2023). Jumlah kluster biasanya ditetapkan berdasarkan jumlah kategori atau segmentasi data yang diharapkan sesuai dengan objektif analisis (Dalam, 2025). Dalam algoritma *K-Means*, *centroid* awal dipilih secara random pada setiap kluster sebagai titik referensi awal dalam proses kalkulasi jarak *Euclidean*. Cara untuk mendapatkan nilai K yang optimal, dapat digunakan metode *Elbow*, yaitu metode dengan pendekatan yang menentukan jumlah kluster terbaik berdasarkan nilai *Within Sum of Square Error* (WSSE). Nilai WSSE merepresentasikan total kuadrat jarak antara data dengan *centroid*. Titik "siku" (*elbow*) pada grafik WSSE terhadap nilai K mengindikasikan jumlah

kluster yang ideal. Adapun perhitungan WSSE (*Within Sum of Square Error*) seperti pada persamaan berikut.

$$\sum_i^n \text{Jarak}(p_i, c_i)^2 \dots \dots \dots (2.2)$$

Keterangan:

P_i = data ke i

C_i = *Centroid* ke i

3. Menghitung jarak *Euclidean*

Pada tahap ini, dilakukan kalkulasi jarak antara setiap data dengan masing-masing *centroid* menggunakan formula *Euclidean Distance* (Kurniawan et al., 2022). Jarak ini digunakan untuk mengukur kedekatan suatu data terhadap pusat kluster yang telah ditetapkan. Nilai jarak terkecil dari hasil kalkulasi akan dianggap sebagai jarak terdekat, sehingga data tersebut akan dialokasikan ke dalam kluster dengan *centroid* yang paling dekat. Proses ini krusial dalam pembentukan kelompok awal yang homogen. Adapun persamaan dari *Euclidean Distance* seperti berikut.

$$d(x, c_i) = \sqrt{\sum_{j=1}^m (x_j - c_{ij})^2} \dots \dots \dots (2.3)$$

Keterangan:

d = jarak pada x dan c

x = nilai *centroid* awal

c = nilai dari baris N

m = jumlah atribut data

i & j = jumlah iterasi

Nilai terkecil dari hasil perhitungan di atas yang dipilih sebagai jarak terdekat ke nilai *centroid* setiap klaster.

4. Menghitung *Centroid* baru

Setelah pengelompokan awal berdasarkan jarak terdekat selesai dilakukan, tahap selanjutnya adalah menghitung nilai mean dari seluruh objek data dalam setiap klaster. Nilai mean ini akan menjadi *centroid* baru yang menggantikan posisi *centroid* sebelumnya. *Centroid* baru tersebut kemudian digunakan dalam kalkulasi jarak *Euclidean* pada iterasi berikutnya. Proses ini bertujuan untuk memperbarui pusat klaster agar semakin representatif terhadap data yang tergabung di dalamnya.

5. Iterasi dan Evaluasi

Proses kalkulasi dilanjutkan dengan melakukan iterasi secara berulang pada tahap penentuan *centroid* baru dan perhitungan jarak *Euclidean*, hingga posisi *centroid* tidak lagi mengalami perubahan atau telah mencapai nilai yang stabil. Iterasi ini menandakan bahwa proses klasterisasi telah mencapai titik konvergensi. Untuk mengevaluasi kualitas hasil klasterisasi, dilakukan pengujian menggunakan *Sum of Square Error* (SSE), yaitu metrik yang mengukur kedekatan data dalam suatu klaster terhadap *centroid*. Nilai SSE yang semakin rendah menunjukkan bahwa hasil pengelompokan semakin optimal. Rumus perhitungan nilai SSE dapat dijelaskan melalui persamaan berikut:

$$SSE = \sum_{k=1}^k (x_i - c_k)^2 \dots \dots \dots (2.4)$$

Nilai *Sum of Square Error* (SSE) yang rendah hingga mendekati nol mengindikasikan bahwa kualitas klasterisasi yang dihasilkan sangat baik. Hal ini menunjukkan bahwa objek-objek dalam setiap klaster memiliki jarak yang relatif dekat terhadap *centroid*. Sebaliknya, jika nilai SSE yang dihasilkan tinggi, maka hal tersebut mengindikasikan bahwa hasil pengelompokan kurang optimal atau tidak akurat.

2.5. DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*)

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) merupakan algoritma *clustering* yang bekerja berdasarkan prinsip kepadatan (*density*) data (Anfelia et al., 2024). Kepadatan dalam konteks *DBSCAN* mengacu pada jumlah titik data yang berada dalam radius *epsilon* (ϵ) dengan jumlah minimal data sebanyak *MinPts* (*Minimum Points*) (Syawali et al., n.d.). Apabila jumlah data dalam radius tersebut memenuhi atau melebihi nilai *MinPts*, termasuk titik data inti itu sendiri, maka area tersebut dianggap memiliki kepadatan yang memadai untuk membentuk sebuah klaster (Tambunan et al., 2025). Berdasarkan konsep kepadatan ini, *DBSCAN* mengklasifikasikan setiap titik data ke dalam tiga kategori status, yaitu titik inti (*core point*), titik batas (*border point*), dan titik *noise* (*outlier*) (Mochurad et al., 2023).

Proses pengelompokan menggunakan algoritma *DBSCAN* dapat dijelaskan melalui beberapa tahapan sebagai berikut :

1. Menetapkan nilai *Minimal Points* (*minPts*) dan *Epsilon* (*eps*).
2. Memilih titik awal secara acak.

3. Menghitung jarak antar titik menggunakan fungsi jarak *Euclidean*, dengan persamaan berikut:

$$E(x, y) = \sqrt{\sum_{i=0}^n (x_i - y_i)^2} \dots \dots \dots (2.5)$$

Keterangan :

$E(X, Y)$ = jarak antara titik X_i dengan titik Y_i

X_j = nilai titik 1 pada *cluster* ke- j

Y_i = nilai centroid I pada *cluster* ke- j

4. *Cluster* terbentuk berdasarkan kepadatan data.
5. Jika jumlah titik dalam radius ϵ mencapai atau melebihi $minpts$, maka titik tersebut dikategorikan sebagai titik inti, dan *cluster* akan terbentuk.
6. Jika titik tersebut adalah titik batas dan tidak ada titik lain dalam jangkauan kepadatan yang dapat dijangkau, proses akan berlanjut ke titik lainnya.
7. Langkah-langkah 3 hingga 6 akan diulang sampai semua titik telah diproses. Proses ini terus berulang, memeriksa titik-titik tetangga baru dan menambahkannya ke dalam klaster yang sama jika memenuhi kriteria kepadatan, hingga tidak ada lagi titik yang dapat ditambahkan.

2.6. Ringkasan penelitian terdahulu

Penelitian mengenai klasterisasi daerah rawan kriminalitas menggunakan algoritma *machine learning* telah banyak dilakukan oleh para peneliti sebelumnya dengan berbagai pendekatan dan metodologi. Studi-studi terdahulu mengeksplorasi penggunaan algoritma *K-Means*, *DBSCAN*, dan algoritma *clustering* lainnya dalam mengidentifikasi pola dan pengelompokan wilayah berdasarkan tingkat

kriminalitas di berbagai daerah. Penelitian-penelitian tersebut menunjukkan bahwa penerapan teknik data mining dan *clustering* dapat memberikan kontribusi signifikan dalam membantu pihak kepolisian dan pemerintah daerah untuk memetakan zona rawan kejahatan, mengoptimalkan strategi patroli, serta mengalokasikan sumber daya keamanan secara lebih efektif. Meskipun demikian, masih terdapat ruang untuk pengembangan penelitian khususnya dalam konteks Wilayah Medan Timur yang memiliki karakteristik geografis dan demografis yang unik. Berikut adalah beberapa penelitian terdahulu yang relevan dengan topik klasterisasi daerah rawan kriminalitas.

Tabel 2.1 Ringkasan Terdahulu

No.	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
1	Pengelompokan data kriminal untuk menentukan algoritma <i>K-Means</i> (Studi Kasus: Polsek Hamparan Perak) (Azhari et al., 2024)	Penelitian ini mengidentifikasi pola daerah rawan tindak kriminal di Polsek Hamparan Perak agar pihak kepolisian dapat melakukan sosialisasi dan tindakan preventif yang lebih efektif.	<i>K-Means</i>	Kelebihan: Membantu pihak berwenang memahami pola kejahatan secara lebih sistematis, sehingga strategi pencegahan dan penanggulangan kriminal dapat dirancang secara lebih tepat sasaran. Kekurangan: Ketergantungan pada jumlah cluster yang harus ditentukan sebelumnya dan sensitivitas terhadap inisialisasi centroid
2	Visualisasi Data Tindak Kejahatan Berdasarkan Jenis Kriminalitas Di Kabupaten Karawang	Penelitian ini memvisualisasikan dan mengklasifikasikan daerah berdasarkan tingkat kriminalitas agar dapat membantu pemerintah dan penegak hukum dalam	<i>K-Means</i>	Kelebihan: Memberikan gambaran distribusi daerah rawan kriminalitas secara visual dan terstruktur, yang dapat digunakan sebagai dasar dalam

No.	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
	dengan Menggunakan Algoritma <i>Clustering K-Means</i> (Rosiana et al., 2023)	pengambilan keputusan dan strategi pencegahan.		perencanaan penanggulangan kejahatan dan alokasi sumber daya secara lebih efektif. Kekurangan: Sensitivitas terhadap inisialisasi awal dan jumlah cluster yang harus ditentukan secara manual
3	<i>Identification Of Environmental Security In Relation To The Crime Case Rate In Simeulue Regency Using Density Based Spatial Clustering Method With Noise (DBSCAN)</i> (Anfelia et al., 2024)	Penelitian ini menganalisis hubungan antara keamanan lingkungan dan tingkat kejahatan di Kabupaten Simeulue serta mengidentifikasi daerah rawan kejahatan melalui analisis spasial	<i>DBSCAN</i>	Kelebihan: Memberikan wawasan bagi pihak berwenang untuk melakukan tindakan pencegahan kejahatan yang lebih tepat sasaran dan meningkatkan keamanan lingkungan melalui pengelolaan sumber daya dan kebijakan yang berbasis data spasial. Kekurangan: Penyesuaian parameter DBSCAN kurang akurat dan mempertimbangkan faktor lain seperti dinamika sosial ekonomi secara lebih mendalam.
4	<i>Clustering Hierarchical, k-Means, DBSCAN</i> (Foundations, 2021)	Penelitian ini melakukan segmentasi pasar dan mengidentifikasi kelompok-kelompok yang memiliki karakteristik serupa, seperti perilaku dan persepsi wisatawan, serta menemukan poin-poin menarik dari data foto	<i>Hierarchical clustering, k-means, dan DBSCAN</i>	Kelebihan: Membantu pengambilan keputusan dan pemasaran yang lebih tepat sasaran dalam industri pariwisata, serta meningkatkan pemahaman tentang perilaku dan preferensi wisatawan. Kekurangan: Akurasi dalam identifikasi cluster kurang dan

No.	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
				penanganan data kompleks atau tidak lengkap
5	<i>Algoritma K-Means Clustering</i> dalam Mengklasifikasi Data Daerah Rawan Tindak Kriminalitas (Polres Kepulauan Mentawai) (Aswan et al., 2021)	Penelitian ini mengelompokkan daerah rawan kriminalitas di Kepulauan Mentawai berdasarkan data kasus pencurian kendaraan bermotor selama lima tahun	<i>K-Means</i>	Kelebihan: Memberikan panduan bagi pihak kepolisian dalam penanganan dan pengawasan daerah rawan tindak kriminal, serta membantu dalam perencanaan strategi pencegahan yang lebih efektif berdasarkan hasil pengelompokan daerah. Kekurangan: Ketergantungan pada data historis yang mungkin tidak mencerminkan kondisi terbaru, serta sensitivitas algoritma terhadap pemilihan pusat cluster awal yang dapat mempengaruhi hasil clustering.

2.7. Analisis GAP

Berdasarkan tinjauan terhadap penelitian-penelitian relevan yang telah dilakukan sebelumnya, terdapat beberapa celah penelitian yang dapat diidentifikasi. Pertama, penelitian sebelumnya cenderung menggunakan algoritma tunggal, baik *K-Means* maupun *DBSCAN*, dalam menganalisis data kriminalitas. Penelitian di Polsek Hampan Perak, Kabupaten Karawang, dan Polres Kepulauan Mentawai hanya menggunakan *K-Means* yang memiliki kelemahan signifikan yaitu sensitivitas terhadap inisialisasi *centroid* awal, ketergantungan pada penentuan

jumlah *cluster* secara manual, serta tidak mampu mengidentifikasi *outlier* dan klaster dengan bentuk tidak beraturan. Sementara itu, penelitian di Kabupaten Simeulue menggunakan *DBSCAN* namun dalam konteks wilayah kepulauan yang berbeda karakteristiknya dengan wilayah perkotaan padat.

Kedua, belum ada penelitian yang secara spesifik membandingkan performa *K-Means* dan *DBSCAN* dalam konteks analisis kriminalitas di wilayah perkotaan Indonesia, khususnya Medan Timur. Penelitian yang membandingkan beberapa metode *clustering* lebih fokus pada segmentasi pasar pariwisata, sehingga konteks dan karakteristik datanya berbeda dengan data kriminalitas.

Ketiga, mayoritas penelitian sebelumnya dilakukan di wilayah dengan karakteristik geografis yang berbeda dengan Medan Timur. Medan Timur sebagai salah satu kecamatan dengan tingkat urbanisasi tinggi di Kota Medan memiliki dinamika kriminalitas yang unik dan memerlukan analisis spesifik yang belum pernah dilakukan.

Untuk mengisi celah penelitian yang ada, penelitian ini mengusulkan pendekatan komparatif dengan mengimplementasikan dua algoritma *clustering* secara bersamaan (*K-Means* dan *DBSCAN*) pada Wilayah Medan Timur. Kombinasi kedua algoritma ini diharapkan dapat memberikan beberapa keunggulan:

1. Perspektif Ganda: *K-Means* akan memberikan gambaran klasterisasi global yang terstruktur, sementara *DBSCAN* akan mengidentifikasi hotspot berdasarkan kepadatan dan mendeteksi *outlier*.

2. Validasi Silang: Hasil dari kedua algoritma dapat saling memvalidasi dan memberikan *insight* yang lebih komprehensif tentang pola sebaran kriminalitas.
3. Deteksi Anomali: *DBSCAN* akan melengkapi kelemahan *K-Means* dalam mengidentifikasi area dengan pola kriminalitas yang tidak beraturan.
4. Rekomendasi Metode Terbaik: Perbandingan performa kedua algoritma akan memberikan rekomendasi metode yang paling sesuai untuk analisis kriminalitas di wilayah perkotaan seperti Medan Timur.

Oleh karena itu, penelitian ini tidak hanya mengidentifikasi daerah rawan kriminalitas tetapi juga memberikan kontribusi metodologis dalam pemilihan algoritma *clustering* yang tepat untuk analisis keamanan publik di konteks perkotaan Indonesia.

BAB III

METODOLOGI PENELITIAN

3.1 Jenis penelitian

Penelitian yang berjudul ” Klasterisasi daerah rawan kriminalitas di Medan Timur berdasarkan data kepolisian menggunakan algoritma *K-Means* dan *DBSCAN*” menggunakan data numerik yaitu data kriminalitas dari polsek Medan Timur untuk mengukur performa algoritma mana yang terbaik dalam menghasilkan pemetaan daerah rawan kriminalitas yang dapat digunakan sebagai dasar pengambilan keputusan dalam pencegahan dan penanggulangan kejahatan berdasarkan metrik evaluasi.

Jenis penelitian ini adalah penelitian kuantitatif, dan dengan metode eksperimental. Metode eksperimental tersebut melakukan eksperimental komputasi dengan mengimplementasikan kedua algoritma, yaitu *K-Means* dan *DBSCAN*, lalu diterapkan menggunakan dataset yang sama dengan membandingkan hasil algoritma mana yang performanya lebih baik menggunakan metrik evaluasi.

3.2 Teknik pengumpulan data

Penelitian ini menggunakan teknik dokumentasi yang dilakukan dengan cara mengumpulkan data kriminalitas dari tahun 2023-2025 yang diperoleh langsung dari pihak Kepolisian Sektor (Polsek) Medan Timur. Data yang dikumpulkan dengan diinput manual meliputi, Nomor laporan, Jenis kelamin Pelapor, Umur Pelapor, Alamat Pelapor, Lokasi Tempat Kejadian Perkara (TKP), dan Jenis Tindak Pidana yang dilakukan. Teknik dokumentasi ini digunakan karena

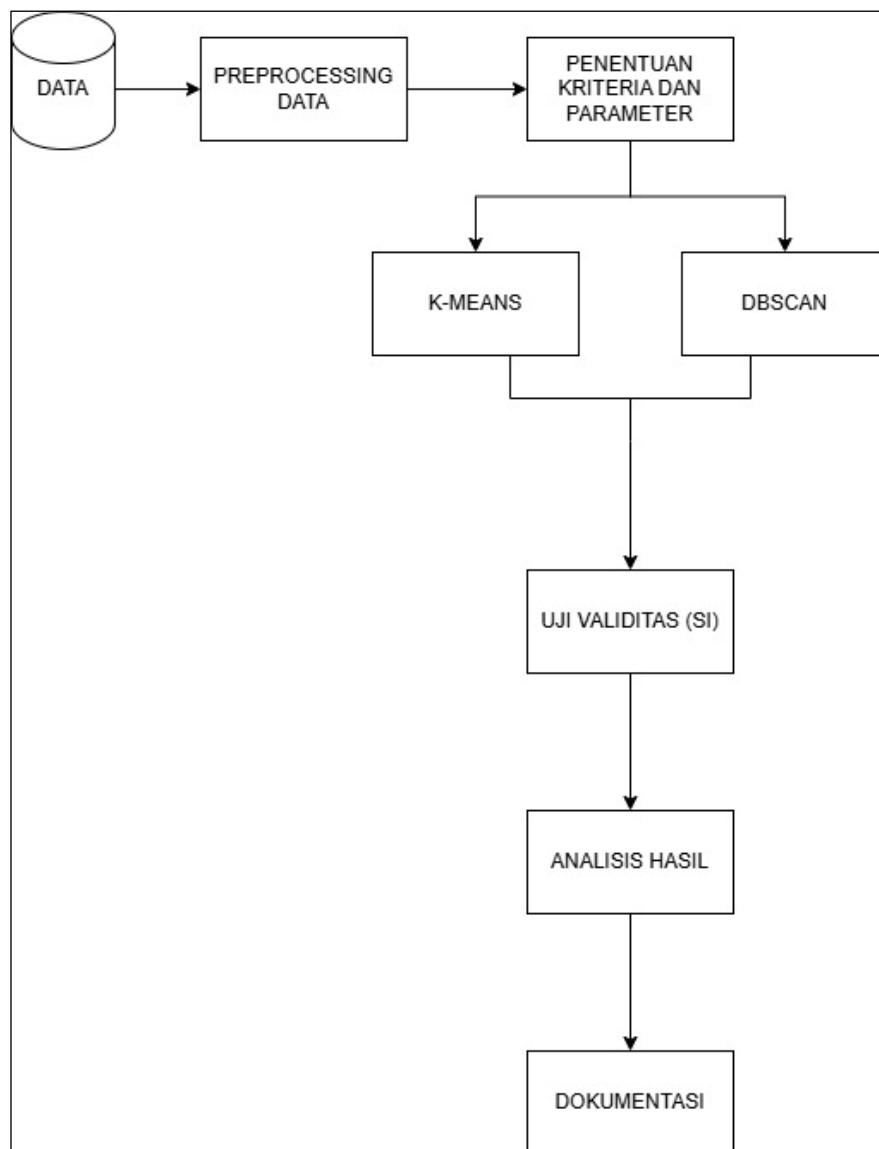
data yang dibutuhkan bersifat kuantitatif yang didapatkan dalam bentuk dokumen resmi kepolisian. Data tersebut selanjutnya akan digunakan sebagai bahan utama dalam proses analisis menggunakan algoritma *K-Means* dan *DBSCAN* untuk mengelompokkan daerah rawan kriminalitas di Kota Medan.

Tabel 2.2 Data Medan Kriminalitas Polsek Medan Timur

No.	Nama Laporan	...	Jenis kriminal	Alamat TKP	Tanggal kejadian	Jam	..
1.	LP/01/1/2023	..	CURANMOR	Jl. Bilal Kel. P. B. Darat I kec. Medan Timur	31/12/2022	19.00 Wib	..
2.	LP/02/1/2023	..	CURAS	Jl. Krakatau Kel. P.B. Darat II Kec.Medan Timur	2/1/2023	17.25 Wib	..
3.	LP/03/1/2023	..	CURANMOR	Jl. Sidomulyo Kel.P.B. darat I Kec medan timur	29/12/2022	16.00 Wib	..
4.	LP/04/1/2023	..	CURAT	Jl. Sejati Kel. Sidorame Barat I Kec. Medan Perjuangan	2/1/2023	05.00 wib	..
5.	LP/05/1/2023	..	CURAS	Jl. Sutomo Ujung Kel. Durian Kec. Medan Timur	27-12-2022	22.00 Wib	..
6.	LP/06/1/2023	..	CURANMOR	Jl. Gereja Kel.Sidorame Barat I Kec. Medan Perjuangan	5/1/2023	08.00 Wib	..
7.	LP/07/1/2023	..	CURAT	JL. Alfalah raya kle. Glugur Darat I Kec.Medan Timur	5/1/2023	11.00 Wib	..
...
1314.	LP/672/12/2025	..	curanmor	jl Perwira 2 medan pulo brayan bengkel medan timur	30/12/2025	15.20 wib	..

3.3 Arsitektur Penelitian

Arsitektur penelitian adalah proses yang dilakukan untuk menganalisis atau memproses data yang sudah dikumpulkan dengan tujuan dapat menemukan jawaban dari pertanyaan penelitian. Penelitian ini menggunakan teknik analisis data kuantitatif dengan pendekatan analisis komparatif untuk membandingkan performa kedua algoritma *clustering* dalam mengklasterisasikan daerah rawan kriminalitas di Kota Medan Timur yaitu algoritma *K- Means* dan *DBSCAN*. Adapun arsitektur penelitian dapat dilihat pada gambar berikut:



Gambar 3.1 Arsitektur Penelitian

1. Pengumpulan data

Tahap ini mengumpulkan data kriminalitas di Kecamatan Medan Timur dari tahun 2023-2024 . Data kepolisian tersebut mencakup lokasi rumah pelaku kriminalitas, jenis tindak pidana yang dilakukan, waktu terjadinya kejadian dari setiap lokasi kejadian kriminalitas.

2. *Preprocessing data*

Setelah data dikumpulkan, tahap selanjutnya yaitu *preprocessing* untuk memastikan kualitas data sebelum data di olah. Tahapan *preprocessing* seperti berikut :

1. *Data cleaning* yaitu menghapus data yang tidak lengkap, dan duplikasi.
2. *Data integration* yaitu menggabungkan data dari data tahun 2023-2025
3. *Data transformation* yaitu mengkonversikan data dengan format yang sesuai.
4. *Data normalization* yaitu melakukan normalisasi dan standardisasi data numerik agar dalam skala yang sama, sehingga tidak ada variabel yang mendominasi dalam proses *clustering*.

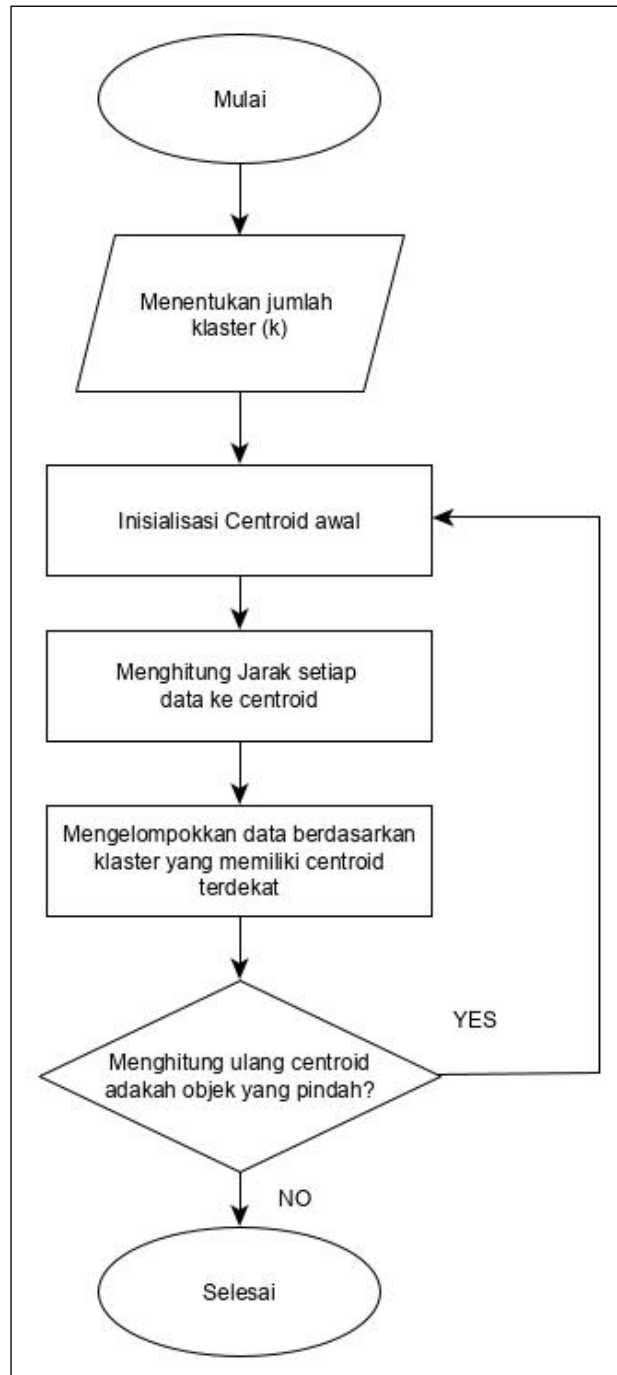
3. Penentuan kriteria dan parameter

Tahap ini melakukan penentuan kriteria dan parameter yang akan digunakan dalam proses *clustering* seperti berikut:

1. Kriteria *clustering*: menentukan atribut dan fitur yang akan digunakan sebagai dasar pengelompokkan, seperti koordinat lokasi, frekuensi kejadian, serta jenis kejahatan.
2. Parameter algoritma: seperti menentukan jumlah K yang optimal untuk *K-Means* dan nilai eps, minPts untuk *DBSCAN*.

4. Proses dengan algoritma *K-Means*

Algoritma *K-Means* digunakan untuk mengelompokkan data ke dalam sejumlah klaster berdasarkan kedekatan karakteristik data. Proses ini bertujuan untuk mengidentifikasi pola wilayah yang memiliki tingkat kerawanan kriminalitas yang serupa.



Gambar 3.2 Flowchart K-Means

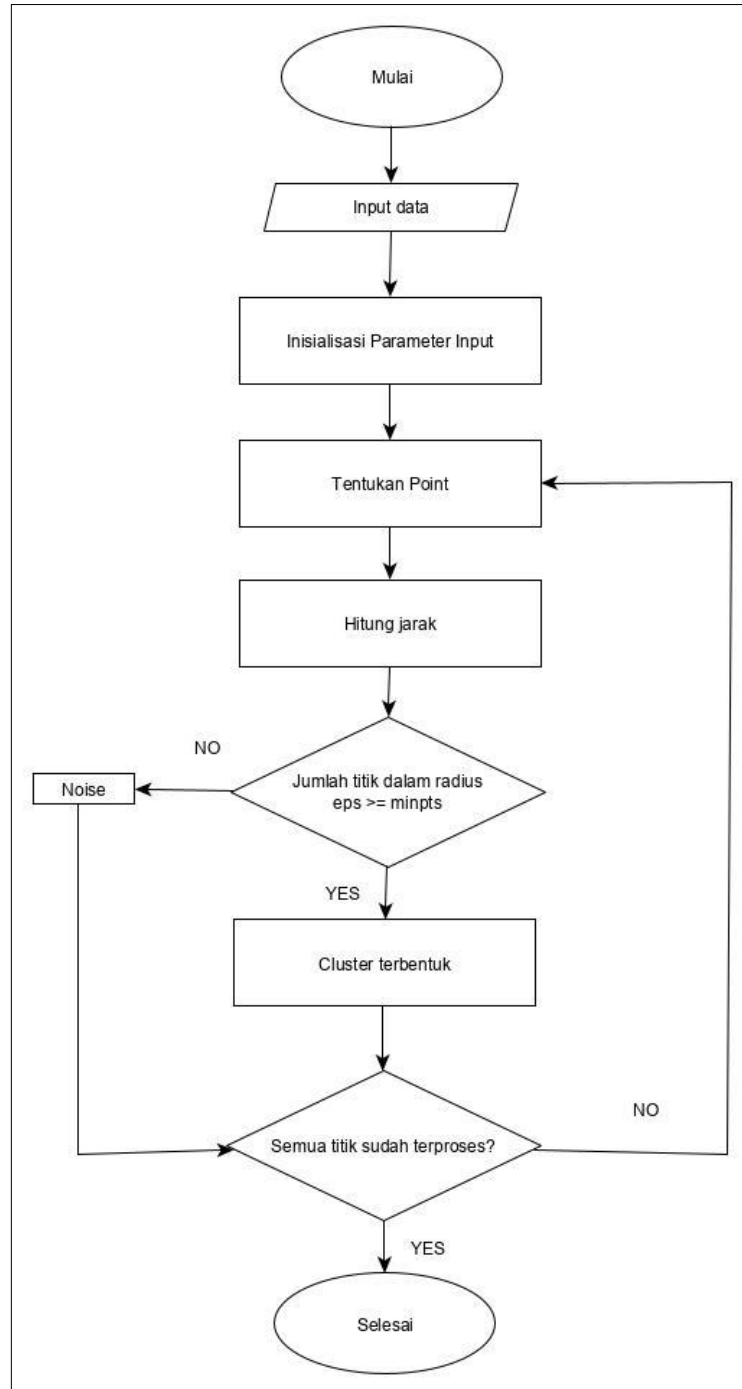
Berdasarkan gambar 3.2 dapat dilihat proses atau alur dari algoritma *K-Means*

dengan penjelasan berikut:

1. Inisialisasi *centroid* awal sebanyak K *cluster*, dan hitunglah dengan rumus pada persamaan (2.2).

2. Hitung jarak setiap data terhadap *centroid* menggunakan *Euclidian Distance* pada persamaan (2.3).
 3. Kelompokkan setiap data ke dalam kluster dengan *centroid* terdekat.
 4. Perbarui posisi *centroid* berdasarkan rata-rata data di setiap kluster.
 5. Ulangi proses hingga *centroid* tidak berubah lagi.
 6. Untuk mengevaluasi kualitas hasil klusterisasi, dilakukan pengujian menggunakan *Sum of Square Error* (SSE), yaitu dengan menggunakan rumus perhitungan pada persamaan (2.4).
5. Proses algoritma *DBSCAN* (*Density-Based Spatial Clustering of Applications with Noise*)

Algoritma *DBSCAN* (*Density-Based Spatial Clustering of Applications with Noise*) digunakan untuk membentuk kluster berdasarkan kepadatan data serta mampu mendeteksi *outlier*. Metode ini efektif untuk data spasial yang memiliki bentuk kluster tidak beraturan.



Gambar 3.3 Flowchart DBSCAN

Berdasarkan gambar 3.3 dapat dilihat proses atau alur dari DBSCAN dengan penjelasan berikut:

1. Menentukan *core points*, dengan cara melihat titik yang memiliki minPts tetangga dalam radius eps.

2. Tentukan titik awal atau p secara acak.
 3. Lakukan perhitungan Eps atau semua jarak titik yang *Density Reachable* terhadap p menggunakan rumus jarak *Euclidean* pada persamaan (2.5).
 4. Jika titik yang memenuhi Eps lebih dari MinPts maka titik p merupakan *core point* dan terbentuk sebuah cluster.
 5. Ulangi langkah 3 - 4 hingga dilakukan proses pada semua titik. Jika p merupakan titik border dan tidak ada titik yang *Density Reachable* terhadap p , maka proses dilanjutkan ke titik yang lain.
6. Analisis hasil dan perbandingan

Analisis hasil dan perbandingan, ini merupakan tahap terakhir yang melakukan evaluasi dan perbandingan performa kedua algoritma menggunakan metrik evaluasi *clustering*, yaitu *Silhouette Index*, mengukur seberapa baik suatu objek ditempatkan dalam klasternya dengan nilai antara -1 hingga 1. Oleh karena itu, menggunakan metrik evaluasi tersebut perbandingan juga dilakukan berdasarkan waktu komputasi, jumlah *cluster* yang terbentuk dengan kemampuan mendeteksi *noise/outlier*. Hasil perbandingan kemudian dianalisis untuk menentukan algoritma mana yang sesuai dan efektif dalam mengidentifikasi daerah rawan kriminalitas di Kota Medan Timur berdasarkan nilai *Silhouette Index* yang diperoleh. Algoritma dengan nilai *Silhouette Index* yang lebih tinggi dianggap memiliki performa *clustering* yang lebih baik.

BAB IV

HASIL DAN PEMBAHASAN

4.1. Deskripsi Data

Kepolisian Sektor Kota Medan Timur, Jalan Jawa, Kelurahan Gang Buntu, Kecamatan Medan Timur, Kota Medan, menyediakan data sekunder yang digunakan untuk penelitian ini.

Tabel 4.1 Data Kriminalitas Sebelum Diolah

No.	No.Laporan	..	Jenis Kriminal	Alamat Tkp	..	Tanggal Kejadian	..	Barang Kriminal
1	LP/01/1/2023	..	Curanmo r	Glugur Darat II	..	30/12/202 4	..	Beat
2	LP/02/1/2023	..	Curanmo r	Tegal Rejo	..	31/12/202 4	..	Beat
3	LP/03/1/2023	..	Curat	P Brayan Bengkel	..	2/1/2025	..	Aluminiu m
4	LP/04/1/2023	..	Curanmo r	Sidodad i	..	2/1/2025	..	Beat
5	LP/05/1/2023	..	Curanmo r	Tegal Rejo	..	27/12/202 4	..	Scoopy
...
131 4	LP/672/12/202 5	..	Curanmo r	P Brayan Bengkel	..	30/12/202 5	..	Honda

Dataset ini terdiri dari lima belas kolom, yaitu nomor, no laporan, tanggal laporan, jenis kelamin pelapor, umur pelapor, alamat pelapor, jenis kriminal, alamat tempat kejadian perkara, hari kejadian, tanggal kejadian, jam, kategori waktu, status laporan, barang kriminal yang diambil secara manual dari buku besar kriminalitas dalam kurun waktu tiga tahun, yaitu 2023-2025.

Setelah data kriminalitas diambil secara manual, selanjutnya dilakukan penghapusan variabel atau fitur yang tidak diperlukan, sehingga menghasilkan tabel berikut:

Tabel 4.2 Fitur Dataset yang Digunakan

No .	Jenis Kriminal	Kelurahan	Kecamatan	Hari	Bulan	Tahun	Jam	Kategori Waktu
1	Curanmor	Pulo Brayan Darat I	Medan Timur	Sabtu	12	2022	19.00 Wib	Malam
2	Curas	Pulo Brayan Darat II	Medan Timur	Senin	1	2023	17.25 Wib	Sore
3	Curanmor	Pulo Brayan Darat I	Medan Timur	Kamis	12	2022	16.00 Wib	Sore
4	Curat	Sidorame Barat I	Medan Perjuangan	Senin	1	2023	05.00 Wib	Pagi
5	Curas	Durian	Medan Timur	Selasa	12	2022	22.00 Wib	Malam

Selanjutnya adalah melakukan pemilihan atribut yang mengacu pada variabel-variabel kriminalitas yang relevan dalam mengklasterisasikan daerah rawan kriminalitas, yang dapat dilihat pada tabel 4.3 berikut ini:

Tabel 4.3 Atribut yang Digunakan

No	Atribut	Keterangan
1	No	Nomor data
2	Jenis Kriminal	Jenis kejahatan
3	Kelurahan	Lokasi kejadian
4	Kecamatan	Wilayah kecamatan
5	Hari	Hari kejadian
6	Bulan	Bulan kejadian
7	Tahun	Tahun kejadian
8	Jam	Waktu kejadian
9	Kategori Waktu	Kategori waktu kejadian

Jumlah data yang digunakan dalam penelitian ini sebanyak 1314 data kejadian kriminal. Setiap data memiliki beberapa atribut yang digunakan untuk proses analisis, yaitu no, jenis kriminal, kelurahan, kecamatan, hari, bulan, tahun, jam, kategori waktu.

Atribut tersebut kemudian digunakan untuk mengetahui pola kejadian kriminalitas berdasarkan jenis kriminalitas, lokasi kejadian/ tempat kejadian perkara, serta waktu terjadi kriminalitas. Jenis kriminalitas yang dianalisis dalam penelitian ini terdiri dari:

1. Curanmor (pencurian kendaraan bermotor)
2. Curat (pencurian dengan pemberatan)
3. Curas (pencurian dengan kekerasan)

Data kejadian kriminalitas tersebut tersebar dalam beberapa kelurahan di wilayah polsek medan timur, yang mencakup 2 kecamatan yaitu Medan Timur dan Medan Perjuangan, serta 20 kelurahan yaitu Durian, Gaharu, Gang Buntu, Glugur Darat I, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Pulo Brayan Darat I, Pulo Brayan Darat II, Sidodadi, Tegal Rejo, Sidorame Barat I, Sidorame Barat II, Sidorame Timur, Sei Kera Hilir I, Sei Kera Hilir II, Sei Kera Hulu, Pahlawan, Pandau Hilir.

4.2. Preprocessing Data

Preprocessing data merupakan tahapan penting diawal dalam proses pengolahan data untuk *machine learning*. Tahap ini dilakukan dengan menggunakan *python*. Tujuannya adalah untuk menyiapkan data mentah agar menjadi data yang sesuai format dan relevan untuk diolah. Proses *preprocessing* meliputi *cleaning data*, *integration data*, *transformasi data*, serta *normalization data*. Kualitas *preprocessing* sangat menentukan kinerja model dalam mengklasterisasikan daerah rawan kriminalitas di wilayah medan timur.

4.2.1. *Data Cleaning*

Data cleaning atau pembersihan data merupakan salah satu tahapan *krusial* dalam proses *preprocessing* sebelum data digunakan untuk analisis atau pemodelan data mining. Tujuan utama dari data cleaning adalah memastikan kualitas data yang digunakan memenuhi standar yang diperlukan sehingga hasil analisis yang diperoleh lebih akurat dan dapat diandalkan.

Pada penelitian ini, proses *data cleaning* dilakukan terhadap dataset kriminalitas Kota Medan yang terdiri dari 1.314 baris dan 9 kolom. Proses *cleaning* diimplementasikan menggunakan bahasa pemrograman *Python* dengan memanfaatkan *library pandas*. Berikut adalah penjelasan lengkap setiap tahapan beserta *syntax* program yang digunakan.

1. *Load data*

Tahapan pertama adalah memuat dataset ke dalam lingkungan *Python*. *Library pandas* digunakan untuk membaca file berformat Microsoft Excel (.xlsx) ke dalam struktur *DataFrame*, yaitu struktur data tabular dua dimensi yang memudahkan proses manipulasi data.

```

cleaning_coba > ...
1 # DATA CLEANING - Dataset Kriminalitas Kota Medan
2 # File input : try.xlsx
3 # File output: data_cleaned_coba.xlsx
4
5 import pandas as pd
6
7 # 1. LOAD DATA
8 df = pd.read_excel("try.xlsx")
9
10 print("=" * 50)
11 print("SEBELUM CLEANING")
12 print("=" * 50)
13 print(f"Jumlah baris   : {df.shape[0]}")
14 print(f"Jumlah kolom   : {df.shape[1]}")
15 print(f"\nMissing values : \n{df.isnull().sum()}")
16 print(f"\nDuplikasi       : {df.duplicated().sum()} baris")

```

Gambar 4.1 Code Python Load Data

Fungsi `pd.read_excel()` membaca file Excel dan menyimpannya ke dalam variabel `df`. Atribut `df.shape` mengembalikan *tuple* (baris, kolom), sedangkan `df.isnull().sum()` menghitung jumlah nilai kosong pada setiap kolom sebagai bahan evaluasi awal. *Pandas* adalah *library* untuk membaca, memanipulasi, dan menyimpan data tabel (Excel/CSV).

2. Menghapus Kolom yang Tidak Relevan

Tidak semua kolom dalam dataset diperlukan untuk proses analisis. Kolom yang hanya berisi nomor urut atau identifier unik tanpa nilai analitis perlu dihapus agar tidak mengganggu proses pemodelan.

```

19 # 2. HAPUS KOLOM TIDAK DIPERLUKAN
20 # Kolom 'No.' hanya berisi nomor urut, tidak relevan untuk analisis
21 df.drop(columns=["No."], inplace=True)

```

Gambar 4.2 Code Python Penghapusan Kolom

Fungsi `df.drop()` digunakan untuk menghapus kolom yang disebutkan dalam parameter `columns`. Parameter `inplace=True` memastikan perubahan langsung diterapkan pada *DataFrame* tanpa perlu membuat variabel baru.

3. Missing Values

Missing values atau nilai yang hilang adalah kondisi di mana suatu sel dalam dataset tidak memiliki nilai. Berdasarkan hasil eksplorasi data, ditemukan nilai kosong pada empat kolom sebagaimana ditampilkan pada tabel berikut.

Tabel 4. 4 *Missing Values* dan Metode Penanganan

Kolom	Jumlah Missing	Metode Penanganan
Kelurahan	2 baris	Diisi nilai 'Tidak Diketahui'
Bulan	1 baris	Diisi dengan nilai modus
Tahun	1 baris	Diisi dengan nilai modus
Kategori Waktu	1 baris	Diisi dengan nilai modus

```
# 3. TANGANI MISSING VALUES
# Kolom 'Kelurahan' → 2 data kosong, isi dengan 'Tidak Diketahui'
df["Kelurahan"].fillna("Tidak Diketahui", inplace=True)

# Kolom 'Bulan' → 1 data kosong, isi dengan modus (bulan paling sering)
modus_bulan = df["Bulan"].mode()[0]
df["Bulan"].fillna(modus_bulan, inplace=True)

# Kolom 'Tahun' → 1 data kosong, isi dengan modus
modus_tahun = df["Tahun"].mode()[0]
df["Tahun"].fillna(modus_tahun, inplace=True)

# Kolom 'Kategori Waktu' → 1 data kosong, isi dengan modus
modus_kategori = df["Kategori Waktu"].str.strip().mode()[0]
df["Kategori Waktu"].fillna(modus_kategori, inplace=True)
```

Gambar 4.3 Code *Python* Penanganan *Missing Values*

Fungsi *fillna()* digunakan untuk mengisi nilai kosong dengan nilai yang ditentukan. Kolom Kelurahan diisi dengan string 'Tidak Diketahui' karena tidak ada acuan yang tepat untuk mengimputasi lokasi. Kolom Bulan, Tahun, dan Kategori Waktu diisi menggunakan nilai modus (nilai yang paling sering muncul) melalui fungsi *mode()[0]*.

4. Pembersihan *Whitespace* pada Kolom *String*

Whitespace adalah karakter spasi atau tab yang tidak terlihat namun terdapat di awal atau akhir nilai string. Keberadaan *whitespace* dapat menyebabkan nilai yang sebenarnya sama dianggap berbeda oleh sistem, misalnya "Malam" dan "Malam " akan dianggap sebagai dua kategori yang berbeda.

```
# 4. BERSIHKAN WHITESPACE PADA KOLOM STRING
kolom_string = ["Jenis Kriminal", "Kelurahan", "Kecamatan",
                "Hari", "Kategori Waktu", "Jam"]

for kolom in kolom_string:
    df[kolom] = df[kolom].str.strip()
```

Gambar 4.4 Code *Python* Pembersihan *Whitespace*

Metode `.str.strip()` pada *pandas* digunakan untuk menghapus spasi yang ada di awal dan akhir setiap nilai *string*. Perulangan *for* diterapkan pada seluruh kolom bertipe *string* agar proses pembersihan berjalan secara efisien.

5. Koreksi *Typo* pada Kolom Jenis Kriminal

Hasil eksplorasi data menunjukkan adanya kesalahan pengetikan (*typo*) pada kolom Jenis Kriminal. Nilai 'Cuart' dan 'Cursa' merupakan bentuk *typo* dari 'Curat' dan 'Curas'. Kesalahan ini perlu dikoreksi agar kategori data menjadi konsisten dan tidak memengaruhi hasil analisis.

```
# 5. KOREKSI TYPO PADA 'Jenis Kriminal'
# Temuan: 'Cuart' dan 'Cursa' adalah typo dari 'Curat' dan 'Curas'
koreksi_jenis = {
    "Cuart": "Curat", # typo → seharusnya Curat
    "Cursa": "Curas", # typo → seharusnya Curas
}
df["Jenis Kriminal"].replace(koreksi_jenis, inplace=True)
```

Gambar 4.5 Code *Python* Koreksi *Typo* Jenis Kriminal

Fungsi `replace()` menerima argumen berupa *dictionary* yang memetakan nilai lama ke nilai baru. Seluruh kemunculan nilai yang salah akan diganti secara otomatis di seluruh baris *DataFrame*.

6. Koreksi *Anomali* pada Kolom Tahun

Ditemukan nilai *anomali* pada kolom Tahun, yaitu nilai 225 yang secara logis tidak mungkin merupakan tahun yang valid dalam konteks dataset kriminalitas Kota Medan. Nilai tersebut diduga merupakan kesalahan penginputan dari angka 2025.

```
# 6. KOREKSI ANOMALI PADA KOLOM 'Tahun'
# Ditemukan nilai 225 yang merupakan kesalahan input (seharusnya 2025)
df["Tahun"].replace(225.0, 2025.0, inplace=True)
```

Gambar 4.6 Code Python Koreksi Anomali Tahun

Penanganan anomali dilakukan dengan mengganti nilai yang tidak valid menggunakan nilai yang dianggap benar berdasarkan konteks data. Validasi ini penting karena nilai tahun yang salah akan memengaruhi analisis tren kejahatan berdasarkan waktu.

7. Konversi Tipe Data

Kolom Bulan dan Tahun terbaca sebagai tipe *data float* (bilangan desimal) oleh *pandas* akibat adanya *missing values* sebelumnya. Setelah *missing values* ditangani, kedua kolom perlu dikonversi ke tipe *integer* agar sesuai dengan representasi semantiknya sebagai angka bulat.

```
# 7. KONVERSI TIPE DATA
# Bulan dan Tahun seharusnya bertipe integer
df["Bulan"] = df["Bulan"].astype(int)
df["Tahun"] = df["Tahun"].astype(int)
```

Gambar 4.7 Code Python Konversi Tipe data

Metode *.astype(int)* mengonversi tipe data kolom ke *integer*. Konversi ini dilakukan setelah proses imputasi *missing values* selesai, karena *pandas* tidak dapat mengonversi kolom yang masih memiliki nilai NaN ke tipe *integer*.

8. Standarisasi Format Kolom Jam

Kolom Jam memiliki format "*HH.MM Wib*" namun terdapat inkonsistensi berupa penulisan huruf kecil dan typo "Iwb" yang seharusnya "WIB". Standarisasi format dilakukan agar seluruh data pada kolom ini memiliki representasi yang seragam.

```
# 8. BERSIHKAN FORMAT KOLOM 'Jam'
# Format jam: "08.00 Wib", "09.00 Iwb" (typo Wib) → standarkan ke "HH.MM WIB"
df["Jam"] = df["Jam"].str.upper() # uppercase semua
df["Jam"] = df["Jam"].str.replace("IWB", "WIB") # koreksi typo 'Iwb'
df["Jam"] = df["Jam"].str.replace("WIB", "WIB") # pastikan konsisten
```

Gambar 4. 8 Code *Python* Standarisasi Format Jam

Metode `.str.upper()` mengubah seluruh karakter menjadi huruf kapital, sedangkan `.str.replace()` mengganti *substring* tertentu dengan nilai yang benar. Kedua operasi ini memastikan konsistensi format pada kolom Jam di seluruh baris data.

9. Penghapusan Data Duplikat

Data duplikat adalah baris yang memiliki nilai identik pada seluruh kolomnya. Keberadaan data duplikat dapat menyebabkan bias pada model analisis karena suatu kejadian dihitung lebih dari satu kali.

```
df.drop_duplicates(inplace=True)
df.reset_index(drop=True, inplace=True)
```

Gambar 4.9 Code *Python* Penghapusan Data Duplikat

Fungsi `drop_duplicates()` menghapus semua baris yang merupakan duplikasi dari baris lain. Setelah penghapusan, fungsi `reset_index(drop=True)` digunakan untuk menyetel ulang indeks *DataFrame* agar tetap berurutan dari 0 tanpa menyimpan indeks lama.

10. Rekapitulasi Proses Data Cleaning

Tabel berikut merangkum seluruh tahapan data cleaning yang telah dilakukan beserta fungsi *Python* yang digunakan, masalah yang ditangani, dan hasilnya.

Tabel 4.5 Tahapan *Data Cleaning*

No	Tahapan	Masalah	Fungsi <i>Python</i>	Hasil
1	Hapus Kolom	Kolom No. tidak relevan	<code>drop()</code>	8 kolom tersisa
2	<i>Missing Values</i>	4 kolom dengan nilai kosong	<code>fillna()</code> , <code>mode()</code>	0 <i>missing values</i>

3	<i>Whitespace</i>	Spasi berlebih dikolom <i>string</i>	<i>str.strip()</i>	Nilai bersih
4	<i>Typo Nilai</i>	Cuart → Curat, Cursa → Curas	<i>replace()</i>	3 kategori valid
5	<i>Anomali Tahun</i>	Nilai 225 → seharusnya 2025	<i>replace()</i>	Tahun valid
6	Tipe Data	Bulan & Tahun bertipe <i>float</i>	<i>astype(int)</i>	Tipe <i>integer</i>
7	Format Jam	Typo Iwb → WIB	<i>str.upper()</i> , <i>str.replace()</i>	Format seragam
8	Duplikasi	Pengecekan data duplikat	<i>drop_duplicates()</i>	Tidak ada duplikat

Berdasarkan seluruh tahapan yang telah dilakukan, dataset kriminalitas Kota Medan yang semula memiliki berbagai permasalahan kualitas data kini telah menjadi dataset yang bersih, konsisten, dan siap untuk diproses pada tahapan selanjutnya. Proses data cleaning ini merupakan fondasi yang sangat penting untuk memastikan hasil analisis data mining yang akurat dan dapat dipercaya.

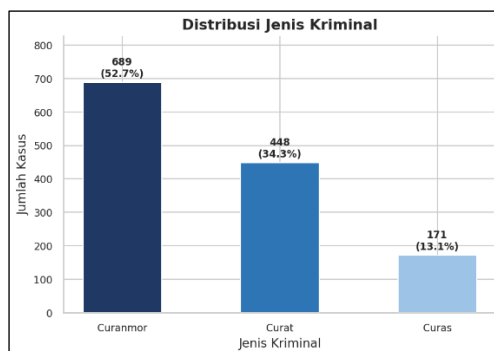
4.2.2. Distribusi Data

Distribusi data merupakan gambaran awal mengenai karakteristik dan sebaran nilai pada setiap atribut dalam dataset. Analisis distribusi data dilakukan sebelum proses pemodelan dengan tujuan untuk memahami pola, proporsi, dan ketidakseimbangan data yang mungkin memengaruhi kualitas hasil analisis data mining.

Pada penelitian ini, distribusi data dianalisis menggunakan bahasa pemrograman Python dengan memanfaatkan library *pandas* untuk perhitungan statistik deskriptif, serta library *matplotlib* dan *seaborn* untuk pembuatan visualisasi grafik.

1. Distribusi Jenis Kriminal

Visualisasi pertama menampilkan distribusi frekuensi berdasarkan jenis tindak kriminal yang terjadi. Grafik batang (bar chart) dipilih karena efektif untuk membandingkan frekuensi antar kategori.

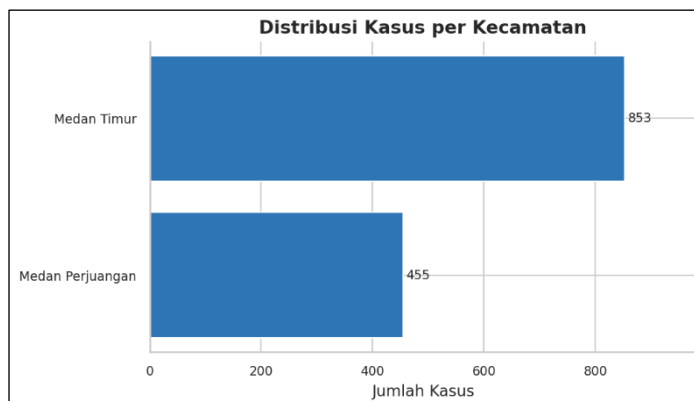


Gambar 4.10 Grafik Distribusi Jenis Kriminal

Berdasarkan grafik di atas, jenis kriminal yang paling dominan adalah Curanmor (pencurian kendaraan bermotor) dengan 689 kasus (52,7%), diikuti oleh Curat (pencurian dengan pemberatan) sebanyak 448 kasus (34,3%), dan Curas (pencurian dengan kekerasan) sebanyak 171 kasus (13,1%). Dominasi Curanmor mengindikasikan bahwa pengamanan kendaraan bermotor menjadi prioritas utama dalam upaya pencegahan kriminalitas.

2. Distribusi Kasus per Kecamatan

Distribusi berdasarkan wilayah kecamatan ditampilkan menggunakan horizontal bar chart agar label kecamatan dapat terbaca dengan jelas.

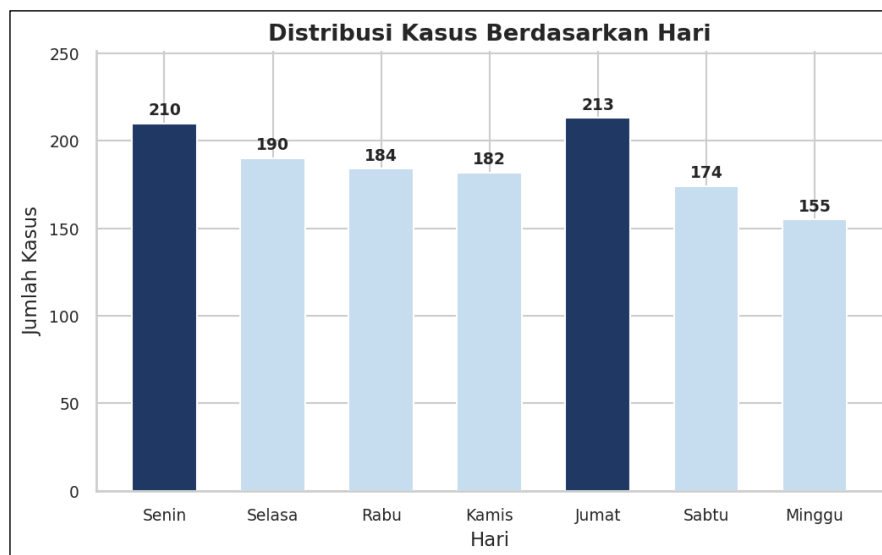


Gambar 4.11 Grafik Distribusi per Kecamatan

Kecamatan Medan Timur mendominasi dengan 853 kasus (65,3%), jauh melampaui Kecamatan Medan Perjuangan dengan 455 kasus (34,7%). Perbedaan yang signifikan ini mengindikasikan bahwa Kecamatan Medan Timur merupakan wilayah dengan tingkat kejahatan yang lebih tinggi dan memerlukan perhatian lebih dalam hal keamanan.

3. Distribusi Kasus Berdasarkan Hari

Distribusi hari kejadian dianalisis untuk mengidentifikasi pola kriminalitas mingguan. Hari diurutkan secara kronologis dari Senin hingga Minggu agar pola dapat terbaca dengan mudah.

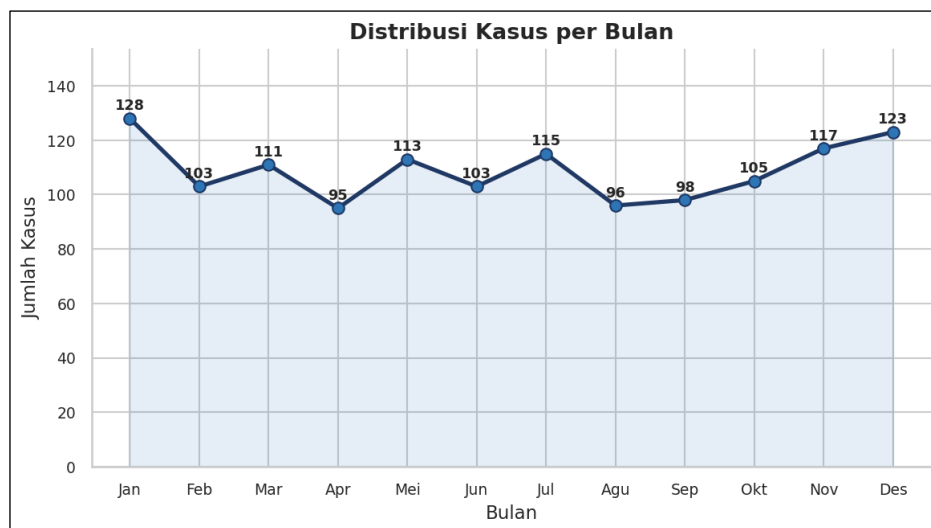


Gambar 4.12 Grafik Distribusi per Hari

Metode `.reindex()` digunakan untuk mengurutkan hasil `value_counts` sesuai urutan hari yang ditentukan secara manual dalam list `hari_order`. Hari Jumat dan Senin menjadi hari dengan kasus tertinggi (masing-masing 213 dan 210 kasus), sementara Minggu merupakan hari dengan kasus paling sedikit (155 kasus).

4. Distribusi Kasus per Bulan

Distribusi bulanan ditampilkan menggunakan *line chart* dengan area bawah yang diarsir (*area chart*) untuk memberikan gambaran tren sepanjang tahun secara lebih intuitif.

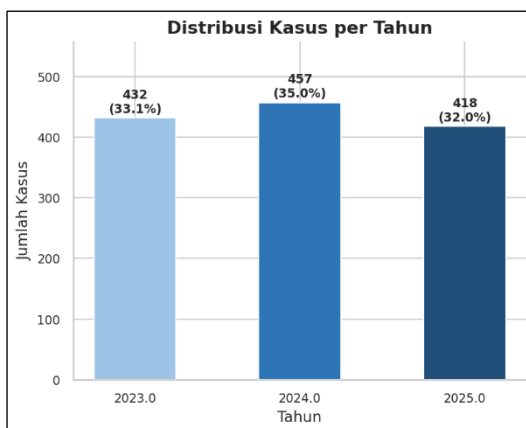


Gambar 4.13 Grafik Distribusi per Bulan

Fungsi `ax.fill_between()` digunakan untuk memberikan warna arsiran di bawah garis grafik, sehingga *area chart* terbentuk dan memudahkan pembaca melihat fluktuasi data. Parameter `alpha=0.12` mengatur tingkat transparansi arsiran. Bulan Januari mencatat kasus tertinggi (128 kasus) dan bulan April mencatat kasus terendah (95 kasus).

5. Distribusi Kasus per Tahun

Distribusi tahunan menampilkan tren jumlah kasus dari tahun ke tahun, yang berguna untuk melihat apakah angka kriminalitas cenderung meningkat, menurun, atau stabil.

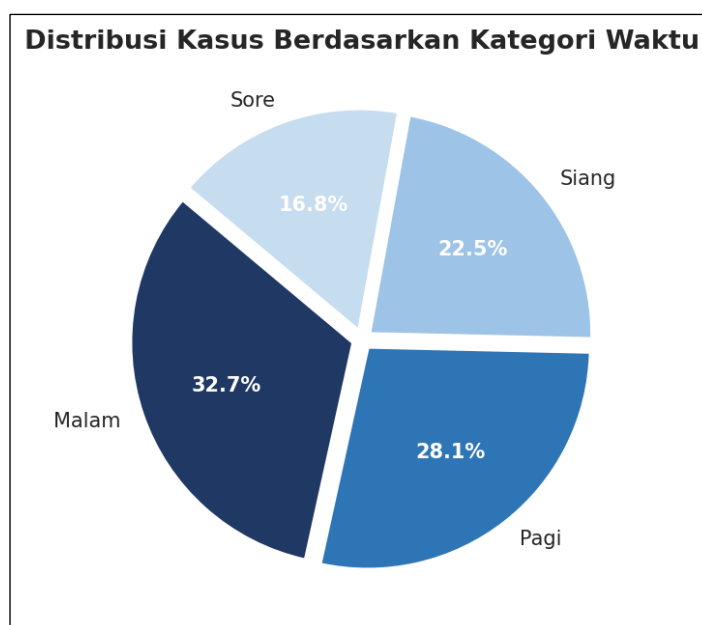


Gambar 4.14 Grafik Distribusi per Tahun

Data kasus terbanyak berasal dari tahun 2024 (457 kasus) disusul tahun 2023 (432 kasus) dan 2025 (418 kasus). Pola ini menunjukkan bahwa data secara substansial berasal dari rentang tahun 2023–2025.

6. Distribusi Berdasarkan Kategori Waktu

Distribusi waktu kejadian divisualisasikan menggunakan *pie chart* untuk menampilkan proporsi setiap kategori waktu secara persentase.



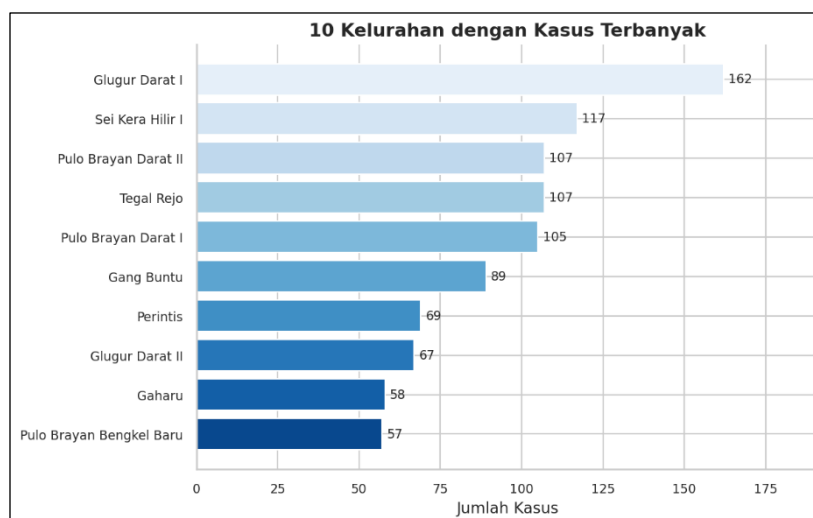
Gambar 4.15 Grafik Distribusi per Kategori_waktu

Parameter `autopct='%1.1f%%'` pada fungsi `ax.pie()` secara otomatis menghitung dan menampilkan persentase setiap irisan. Parameter `explode`

memberikan efek pemisahan kecil pada setiap irisan agar tampak lebih jelas. Waktu Malam mendominasi dengan 32,7% kasus, diikuti Pagi (28,1%), Siang (22,5%), dan Sore (16,8%).

7. Top 10 Kelurahan dengan Kasus Terbanyak

Distribusi kelurahan ditampilkan secara terbatas pada 10 kelurahan teratas menggunakan metode head(10) agar visualisasi tetap mudah dibaca.

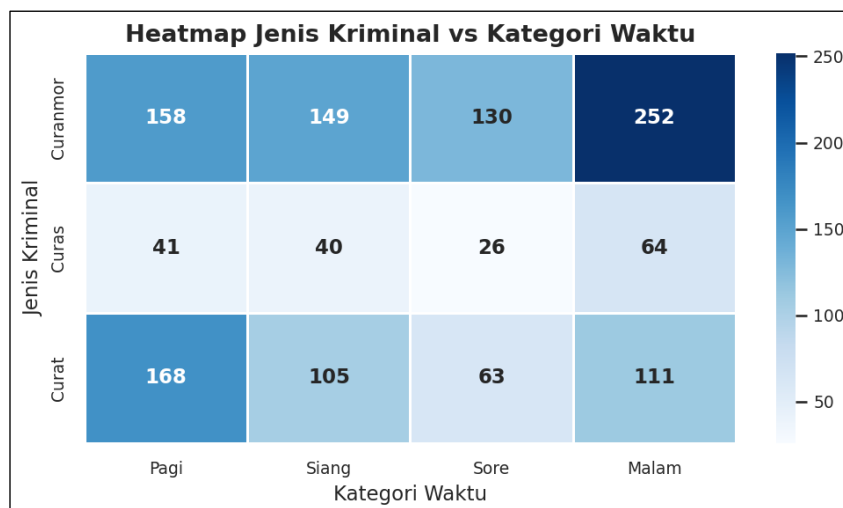


Gambar 4.16 Grafik 10 Kelurahan dengan Kasus Terbanyak

Kelurahan Glugur Darat I mencatat kasus tertinggi dengan 162 kejadian, diikuti oleh Tegal Rejo (117 kasus) dan Pulo Brayan Darat II (107 kasus). Konsentrasi kasus pada kelurahan-kelurahan ini memberikan informasi penting bagi pihak keamanan dalam menentukan lokasi prioritas patroli dan pengawasan.

8. Heatmap Jenis Kriminal vs Kategori Waktu

Heatmap digunakan untuk menganalisis hubungan antara jenis kriminal dan kategori waktu secara bersamaan. Visualisasi ini memberikan gambaran pada kombinasi mana frekuensi kejadian paling tinggi terjadi.



Gambar 4.17 Heatmap Jenis Kriminal VS Kategori Waktu

Fungsi `pd.crosstab()` membuat tabel silang (*contingency table*) yang menghitung frekuensi kemunculan kombinasi dua variabel kategorik. Fungsi `sns.heatmap()` kemudian memvisualisasikan tabel tersebut dengan gradasi warna, di mana warna yang lebih gelap menunjukkan frekuensi yang lebih tinggi. Berdasarkan heatmap, Curanmor paling sering terjadi pada malam hari (252 kasus), sedangkan Curat paling banyak terjadi pada pagi hari (168 kasus).

9. Ringkasan Hasil Analisis Distribusi Data

Berdasarkan seluruh visualisasi yang telah dihasilkan, berikut adalah ringkasan temuan utama dari analisis distribusi data kriminalitas Kota Medan.

Tabel 4. 6 Hasil Analisis Distribusi Data

Aspek	Temuan Utama	Keterangan
Jenis Kriminal	Curanmor (52,7%)	Kasus paling dominan
Kecamatan	Medan Timur (65,3%)	Wilayah paling rawan
Kelurahan	Glugur Darat I (162 kasus)	Kelurahan tersibuk
Hari	Jumat & Senin (213 & 210)	Hari paling rawan
Bulan	Januari (128 kasus)	Bulan tertinggi
Tahun	2024 (457 kasus)	Tahun terbanyak
Waktu	Malam (32,7%)	Waktu paling rawan
Pola	Curanmor malam = 252 kasus	Kombinasi tertinggi

Secara keseluruhan, analisis distribusi data memberikan pemahaman awal yang komprehensif mengenai karakteristik dataset kriminalitas Kota Medan.

Temuan-temuan ini tidak hanya berguna sebagai dasar interpretasi hasil data mining, tetapi juga dapat menjadi informasi awal yang berharga bagi pemangku kepentingan dalam menyusun strategi pencegahan dan penanggulangan kejahatan yang lebih efektif dan tepat sasaran.

Berikut program python untuk menampilkan hasil analisis distribusi grafik data kriminalitas yang dijelaskan sebelumnya:

```

241 # 11. RINGKASAN AKHIR
242 print("\n" + "=" * 55)
243 print("RINGKASAN DISTRIBUSI DATA")
244 print("=" * 55)
245 print(f"Total kasus      : {len(df),}")
246 print(f"Rentang tahun      : {df['Tahun'].min()} - {df['Tahun'].max()}")
247 print(f"Jenis kriminal      : {df['Jenis Kriminal'].nunique()} kategori")
248 print(f"Kecamatan          : {df['Kecamatan'].nunique()} kecamatan")
249 print(f"Kelurahan          : {df['Kelurahan'].nunique()} kelurahan")
250 print(f"Kriminal terbanyak  : {df['Jenis Kriminal'].value_counts().idxmax()} "
251       f"({df['Jenis Kriminal'].value_counts().max():,} kasus)")
252 print(f"Waktu paling rawan  : {df['Kategori Waktu'].value_counts().idxmax()} "
253       f"({df['Kategori Waktu'].value_counts().max():,} kasus)")
254 print(f"Hari paling rawan   : {df['Hari'].value_counts().idxmax()} "
255       f"({df['Hari'].value_counts().max():,} kasus)")
256 print(f"\n📁 Semua grafik tersimpan di folder 'output_distribusi/')

```

Gambar 4.18 Program Python distribusi grafik

4.2.3. *Featuring Engineering dan Agrerasi Data*

Feature engineering adalah proses membentuk fitur-fitur baru dari data mentah agar dapat diproses oleh algoritma klasterisasi. Pada tahap ini, data yang semula berjumlah 1.312 baris (satu baris per kejadian) diringkas menjadi 20 baris (satu baris per kelurahan) melalui proses agregasi. Lima fitur yang dibentuk ditampilkan pada Tabel 4.7 berikut.

Tabel 4.7 Fitur Yang Digunakan Setelah *Cleaning Data*

No.	Fitur	Keterangan
1	Total_kriminal	total keseluruhan kasus kriminal yang terjadi di kelurahan tersebut.
2	Curanmor	jumlah kasus pencurian kendaraan bermotor per kelurahan.
3	Curas	jumlah kasus pencurian dengan kekerasan per kelurahan.

4	Curat	jumlah kasus pencurian dengan pemberatan per kelurahan
5	Rata_waktu	Rata-rata waktu kejadian yang diencode secara numerik (Pagi=1, Siang=2, Sore=3, Malam=4).

Seluruh proses *feature engineering* dan agregasi data diimplementasikan dalam satu blok kode Python menggunakan *library pandas*. Proses dibagi menjadi tiga tahap utama: menghitung total kasus, memisahkan kasus per jenis kriminal dengan *pivot_table()*, dan menghitung rata-rata waktu kejadian.

```

1 import pandas as pd
2
3 # Baca dataset hasil cleaning
4 df = pd.read_csv('data_cleaned.csv')
5 # Fitur 1: total_kriminal
6 total = df.groupby('kelurahan').size().reset_index(name='total_kriminal')
7 # Fitur 2-4: curanmor, curas, curat
8 pivot = df.pivot_table(index='kelurahan', columns='jenis_kriminal',
9                          aggfunc='size', fill_value=0).reset_index()
10 pivot = pivot[['kelurahan', 'curanmor', 'curas', 'curat']]
11 # Fitur 5: rata_waktu
12 waktu = df.groupby('kelurahan')['jam'].mean().round(2).reset_index()
13 waktu.columns = ['kelurahan', 'rata_waktu']
14 # Gabungkan semua fitur (agregasi akhir)
15 df_agregasi = total.merge(pivot, on='kelurahan').merge(waktu, on='kelurahan')
16 df_agregasi = df_agregasi.sort_values('total_kriminal',
17                                     ascending=False).reset_index(drop=True)
18 df_agregasi.to_csv('hasil_agregasi_kelurahan.csv', index=False)

```

Gambar 4.19 Code python feature engineering dan agregasi

Hasil proses agregasi menghasilkan tabel dengan 20 baris dan 5 fitur numerik. Tabel 4.8 berikut menampilkan keseluruhan data per kelurahan yang akan digunakan sebagai input algoritma klasterisasi.

Tabel 4.8 Hasil Agregasi Data per Kelurahan

Kelurahan	total kriminal	curanmor	curas	curat	rata waktu
Glugur Darat I	43	23	6	14	2.558.140
Sidorame Barat II	58	25	7	26	2.655.172
Pulo Brayon Bengkel Baru	89	34	25	30	2.629.213
Gang Buntu	162	117	15	30	2.567.901
Tegal Rejo	67	48	2	17	2.626.866
Gaharu	34	17	2	15	2.060.606
Pandau Hilir	33	20	5	8	2.060.606
Pulo Brayon Darat I	69	33	6	30	2.420.290

Sidorame Timur	56	25	9	22	2.517.857
Pahlawan	57	22	13	22	2.596.491
Perintis	105	70	14	21	2.790.476
Sidorame Barat I	107	56	18	33	2.616.822
Sei Kera Hilir II	117	57	8	52	2.350.427
Glugur Darat II	13	4	2	7	1.923.077
Sei Kera Hulu	33	18	1	14	2.151.515
Durian	35	19	9	7	2.600.000
Pulo Brayon Darat II	47	26	8	13	2.638.298
Pulo Brayon Bengkel	32	9	2	21	2.343.750
Sidodadi	44	21	9	14	2.522.727
Sei Kera Hilir I	107	45	10	52	2.738.318

Analisis korelasi dilakukan untuk mengukur seberapa kuat hubungan linear antar fitur menggunakan metode *Pearson Correlation*. Nilai korelasi berkisar antara -1 hingga +1, di mana nilai mendekati 1 menunjukkan korelasi positif kuat, mendekati -1 menunjukkan korelasi negatif kuat, dan mendekati 0 menunjukkan tidak ada hubungan linear.

```
# Hitung matriks korelasi
fitur = ['total_kriminal', 'curanmor', 'curas', 'curat', 'rata_waktu']
korelasi = df_agregasi[fitur].corr().round(2)

# Tampilkan heatmap
sns.heatmap(korelasi, annot=True, fmt='.2f',
            cmap='coolwarm', vmin=-1, vmax=1,
            linewidths=0.5, square=True)
plt.title('Heatmap Korelasi Antar Fitur')
plt.tight_layout()
plt.savefig('feature_engineering_korelasi.png', dpi=150)
```

Gambar 4. 20 Code Python Analisis Korelasi

Tabel 4.9 berikut menampilkan nilai korelasi Pearson antar seluruh fitur. Nilai yang dicetak tebal menunjukkan korelasi yang cukup kuat ($|r| > 0,5$).

Tabel 4.9 Matriks Korelasi Pearson Antar Fitur

Fitur	total_kriminal	curanmor	curas	curat	rata_waktu
total_kriminal	1.00	0.94	0.63	0.75	0.53
curanmor	0.94	1.00	0.49	0.51	0.45
curas	0.63	0.49	1.00	0.40	0.60

curat	0.75	0.51	0.40	1.00	0.37
rata waktu	0.53	0.45	0.60	0.37	1.00

Berdasarkan matriks korelasi pada Tabel 4.9, diperoleh empat temuan penting. Pertama, fitur curanmor memiliki korelasi tertinggi terhadap total_kriminal ($r = 0.94$), diikuti oleh curat ($r = 0.75$). Hal ini menunjukkan bahwa kelurahan dengan kasus curanmor dan curat cenderung memiliki total kriminalitas lebih tinggi. Kedua, fitur curas memiliki korelasi sedang ($r = 0.63$) terhadap total_kriminal. Kelima fitur yang dibentuk memiliki karakteristik yang saling melengkapi dan tidak redundan. Data hasil agregasi ini selanjutnya akan dinormalisasi menggunakan metode *Min-Max Normalization* sebelum dimasukkan ke dalam algoritma *K-Means* dan *DBSCAN*.

4.2.4. Normalisasi Data (*Min-Max Normalization*)

Normalisasi data yang dilakukan menggunakan metode min-max normalization untuk menyamakan skala seluruh fitur ke dalam rentang nilai 0-1, sehingga tidak ada satu fitur pun yang mendominasi proses perhitungan jarak Euclidean. Formula normalisasi Min-Max terdapat pada persamaan 2.1.

Hasil normalisasi x' akan selalu berada dalam rentang $[0, 1]$. Nilai 0.0000 diperoleh ketika nilai data sama dengan x_{min} (nilai terendah pada fitur tersebut), sedangkan nilai 1.0000 diperoleh ketika nilai data sama dengan x_{max} (nilai tertinggi). Nilai di antara 0 dan 1 menunjukkan posisi relatif data terhadap seluruh rentang nilai fitur tersebut.

Tabel 4.10 Nilai Min, Max, dan Range Setiap Fitur

Fitur	Nilai Terkecil	Nilai Terbesar	Range	Keterangan
Total Kriminal	13	162	149	$13 - 162 = 149$
Curanmor	4	117	113	$4 - 117 = 113$
Curas	1	25	24	$1 - 25 = 24$

Curat	7	52	45	$7 - 52 = 45$
Rata Waktu	1.92	2.79	0.86	$1.92 - 2.79 = 0.86$

Proses normalisasi diimplementasikan tanpa *library machine learning*. Nilai min dan max dicari menggunakan *perulangan for* biasa, kemudian rumus *Min-Max* diterapkan secara eksplisit pada setiap nilai.

```
def hitung_min(kolom_data):
    """Menghitung nilai minimum secara manual tanpa fungsi bawaan."""
    minimum = kolom_data[0]
    for nilai in kolom_data[1:]:
        if nilai < minimum:
            minimum = nilai
    return minimum

def hitung_max(kolom_data):
    """Menghitung nilai maksimum secara manual tanpa fungsi bawaan."""
    maksimum = kolom_data[0]
    for nilai in kolom_data[1:]:
        if nilai > maksimum:
            maksimum = nilai
    return maksimum

min_val = {}
max_val = {}
for f in fitur_cols:
    data_fitur = list(df[f])
    min_val[f] = hitung_min(data_fitur)
    max_val[f] = hitung_max(data_fitur)

print("\n" + "=" * 70)
print("NILAI MIN, MAX, DAN SELISIH SETIAP FITUR")
print("=" * 70)
print(f" {'Fitur':<20} {'Min':>10} {'Max':>10} {'Selisih':>10}")
print(f" {'-'*20} {'-'*10} {'-'*10} {'-'*10}")
for f in fitur_cols:
    selisih = max_val[f] - min_val[f]
    print(f" {f:<20} {min_val[f]:>10.4f} {max_val[f]:>10.4f} {selisih:>10.4f}")
```

Gambar 4.21 Code Python Normalisasi Min-Max (tanpa library)

Selanjutnya dilakukan perhitungan manual normalisasi tiap fitur dengan kelurahan. Berikut hitungan manual normalisasi per kelurahan:

1. Fitur Total_Kriminal ($min = 13$, $max = 162$, $range = 149$)

Tabel 4.11 Normalisasi total_kriminal

Kelurahan	Nilai Asli	Cara Hitung	Hasil
Durian	43	$(43-13)/149 = 30/149$	0.2013

Gaharu	58	$(58-13)/149 = 45/149$	0.3020
Gang Buntu	89	$(89-13)/149 = 76/149$	0.5101
Glugur Darat I	162	$(162-13)/149 = 149/149$	1.0000
Glugur Darat II	67	$(67-13)/149 = 54/149$	0.3624
Pahlawan	34	$(34-13)/149 = 21/149$	0.1409
Pandau Hilir	33	$(33-13)/149 = 20/149$	0.1342
Perintis	69	$(69-13)/149 = 56/149$	0.3758
Pulo Brayan Bengkel	56	$(56-13)/149 = 43/149$	0.2886
Pulo Brayan Bengkel Baru	57	$(57-13)/149 = 44/149$	0.2953
Pulo Brayan Darat I	105	$(105-13)/149 = 92/149$	0.6174
Pulo Brayan Darat II	107	$(107-13)/149 = 94/149$	0.6309
Sei Kera Hilir I	117	$(117-13)/149 = 104/149$	0.6980
Sei Kera Hilir II	13	$(13-13)/149 = 0/149$	0.0000
Sei Kera Hulu	33	$(33-13)/149 = 20/149$	0.1342
Sidodadi	35	$(35-13)/149 = 22/149$	0.1477
Sidorame Barat I	47	$(47-13)/149 = 34/149$	0.2282
Sidorame Barat II	32	$(32-13)/149 = 19/149$	0.1275
Sidorame Timur	44	$(44-13)/149 = 31/149$	0.2081
Tegal Rejo	107	$(107-13)/149 = 94/149$	0.6309

2. Fitur Curanmor ($min = 4, max = 117, range = 113$)

Tabel 4.12 Normalisasi Curanmor

Kelurahan	Nilai Asli	Cara Hitung	Hasil
Durian	23	$(23-4)/113 = 19/113$	0.1681
Gaharu	25	$(25-4)/113 = 21/113$	0.1858
Gang Buntu	34	$(34-4)/113 = 30/113$	0.2655
Glugur Darat I	117	$(117-4)/113 = 113/113$	1.0000
Glugur Darat II	48	$(48-4)/113 = 44/113$	0.3894
Pahlawan	17	$(17-4)/113 = 13/113$	0.1150
Pandau Hilir	20	$(20-4)/113 = 16/113$	0.1416
Perintis	33	$(33-4)/113 = 29/113$	0.2566
Pulo Brayan Bengkel	25	$(25-4)/113 = 21/113$	0.1858
Pulo Brayan Bengkel Baru	22	$(22-4)/113 = 18/113$	0.1593
Pulo Brayan Darat I	70	$(70-4)/113 = 66/113$	0.5841
Pulo Brayan Darat II	56	$(56-4)/113 = 52/113$	0.4602
Sei Kera Hilir I	57	$(57-4)/113 = 53/113$	0.4690
Sei Kera Hilir II	4	$(4-4)/113 = 0/113$	0.0000
Sei Kera Hulu	18	$(18-4)/113 = 14/113$	0.1239
Sidodadi	19	$(19-4)/113 = 15/113$	0.1327
Sidorame Barat I	26	$(26-4)/113 = 22/113$	0.1947
Sidorame Barat II	9	$(9-4)/113 = 5/113$	0.0442
Sidorame Timur	21	$(21-4)/113 = 17/113$	0.1504
Tegal Rejo	45	$(45-4)/113 = 41/113$	0.3628

3. Fitur Curas ($min = 1$, $max = 25$, $range = 24$)

Tabel 4.13 Normalisasi Curas

Kelurahan	Nilai Asli	Cara Hitung	Hasil
Durian	6	$(6-1)/24 = 5/24$	0.2083
Gaharu	7	$(7-1)/24 = 6/24$	0.2500
Gang Buntu	25	$(25-1)/24 = 24/24$	1.0000
Glugur Darat I	15	$(15-1)/24 = 14/24$	0.5833
Glugur Darat II	2	$(2-1)/24 = 1/24$	0.0417
Pahlawan	2	$(2-1)/24 = 1/24$	0.0417
Pandau Hilir	5	$(5-1)/24 = 4/24$	0.1667
Perintis	6	$(6-1)/24 = 5/24$	0.2083
Pulo Brayan Bengkel	9	$(9-1)/24 = 8/24$	0.3333
Pulo Brayan Bengkel Baru	13	$(13-1)/24 = 12/24$	0.5000
Pulo Brayan Darat I	14	$(14-1)/24 = 13/24$	0.5417
Pulo Brayan Darat II	18	$(18-1)/24 = 17/24$	0.7083
Sei Kera Hilir I	8	$(8-1)/24 = 7/24$	0.2917
Sei Kera Hilir II	2	$(2-1)/24 = 1/24$	0.0417
Sei Kera Hulu	1	$(1-1)/24 = 0/24$	0.0000
Sidodadi	9	$(9-1)/24 = 8/24$	0.3333
Sidorame Barat I	8	$(8-1)/24 = 7/24$	0.2917
Sidorame Barat II	2	$(2-1)/24 = 1/24$	0.0417
Sidorame Timur	9	$(9-1)/24 = 8/24$	0.3333
Tegal Rejo	10	$(10-1)/24 = 9/24$	0.3750

4. Fitur Curat ($min = 7$, $max = 52$, $range = 45$)

Tabel 4.14 Normalisasi Curat

Kelurahan	Nilai Asli	Cara Hitung	Hasil
Durian	14	$(14-7)/45 = 7/45$	0.1556
Gaharu	26	$(26-7)/45 = 19/45$	0.4222
Gang Buntu	30	$(30-7)/45 = 23/45$	0.5111
Glugur Darat I	30	$(30-7)/45 = 23/45$	0.5111
Glugur Darat II	17	$(17-7)/45 = 10/45$	0.2222
Pahlawan	15	$(15-7)/45 = 8/45$	0.1778
Pandau Hilir	8	$(8-7)/45 = 1/45$	0.0222
Perintis	30	$(30-7)/45 = 23/45$	0.5111
Pulo Brayan Bengkel	22	$(22-7)/45 = 15/45$	0.3333
Pulo Brayan Bengkel Baru	22	$(22-7)/45 = 15/45$	0.3333
Pulo Brayan Darat I	21	$(21-7)/45 = 14/45$	0.3111
Pulo Brayan Darat II	33	$(33-7)/45 = 26/45$	0.5778
Sei Kera Hilir I	52	$(52-7)/45 = 45/45$	1.0000
Sei Kera Hilir II	7	$(7-7)/45 = 0/45$	0.0000
Sei Kera Hulu	14	$(14-7)/45 = 7/45$	0.1556
Sidodadi	7	$(7-7)/45 = 0/45$	0.0000
Sidorame Barat I	13	$(13-7)/45 = 6/45$	0.1333

Sidorame Barat II	21	$(21-7)/45 = 14/45$	0.3111
Sidorame Timur	14	$(14-7)/45 = 7/45$	0.1556
Tegal Rejo	52	$(52-7)/45 = 45/45$	1.0000

5. Fitur Rata_Waktu ($min = 1.9231$, $max = 2.7905$, $range = 0.8674$)

Tabel 4.15 Normalisasi Rata_waktu

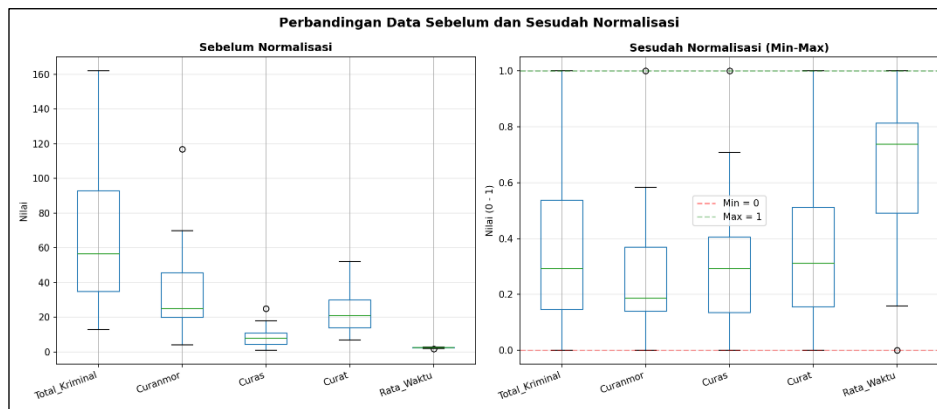
Kelurahan	Nilai Asli	Cara Hitung	Hasil
Durian	2.5581	$(2.5581-1.9231)/0.8674 = 0.6351/0.8674$	0.7321
Gaharu	2.6552	$(2.6552-1.9231)/0.8674 = 0.7321/0.8674$	0.8440
Gang Buntu	2.6292	$(2.6292-1.9231)/0.8674 = 0.7061/0.8674$	0.8141
Glugur Darat I	2.5679	$(2.5679-1.9231)/0.8674 = 0.6448/0.8674$	0.7434
Glugur Darat II	2.6269	$(2.6269-1.9231)/0.8674 = 0.7038/0.8674$	0.8114
Pahlawan	2.0606	$(2.0606-1.9231)/0.8674 = 0.1375/0.8674$	0.1586
Pandau Hilir	2.0606	$(2.0606-1.9231)/0.8674 = 0.1375/0.8674$	0.1586
Perintis	2.4203	$(2.4203-1.9231)/0.8674 = 0.4972/0.8674$	0.5732
Pulo Brayon Bengkel	2.5179	$(2.5179-1.9231)/0.8674 = 0.5948/0.8674$	0.6857
Pulo Brayon Bengkel Baru	2.5965	$(2.5965-1.9231)/0.8674 = 0.6734/0.8674$	0.7764
Pulo Brayon Darat I	2.7905	$(2.7905-1.9231)/0.8674 = 0.8674/0.8674$	1.0000
Pulo Brayon Darat II	2.6168	$(2.6168-1.9231)/0.8674 = 0.6937/0.8674$	0.7998
Sei Kera Hilir I	2.3504	$(2.3504-1.9231)/0.8674 = 0.4274/0.8674$	0.4927
Sei Kera Hilir II	1.9231	$(1.9231-1.9231)/0.8674 = 0.0000/0.8674$	0.0000
Sei Kera Hulu	2.1515	$(2.1515-1.9231)/0.8674 = 0.2284/0.8674$	0.2634
Sidodadi	2.6000	$(2.6000-1.9231)/0.8674 = 0.6769/0.8674$	0.7804
Sidorame Barat I	2.6383	$(2.6383-1.9231)/0.8674 = 0.7152/0.8674$	0.8246
Sidorame Barat II	2.3438	$(2.3438-1.9231)/0.8674 = 0.4207/0.8674$	0.4850
Sidorame Timur	2.5227	$(2.5227-1.9231)/0.8674 = 0.5997/0.8674$	0.6913
Tegal Rejo	2.7383	$(2.7383-1.9231)/0.8674 = 0.8152/0.8674$	0.9399

Berdasarkan hasil hitungan manual normalisasi tiap fitur atau atribut yang digunakan, maka rekap tabel normalisasi minmax seluruhnya yaitu:

Tabel 4.16 Normalisasi MinMax Keseluruhan Fitur

Kelurahan	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Durian	0.201342	0.168142	0.208333	0.155556	0.732146
Gaharu	0.302013	0.185841	0.25	0.422222	0.844012
Gang Buntu	0.510067	0.265487	1	0.511111	0.814085
Glugur Darat I	1	1	0.583333	0.511111	0.7434
Glugur Darat II	0.362416	0.389381	0.041667	0.222222	0.811378
Pahlawan	0.14094	0.115044	0.041667	0.177778	0.158553
Pandau Hilir	0.134228	0.141593	0.166667	0.022222	0.158553
Perintis	0.375839	0.256637	0.208333	0.511111	0.573223
Pulo Brayan Bengkel	0.288591	0.185841	0.333333	0.333333	0.685705
Pulo Brayan Bengkel Baru	0.295302	0.159292	0.5	0.333333	0.77636
Pulo Brayan Darat I	0.61745	0.584071	0.541667	0.311111	1
Pulo Brayan Darat II	0.630872	0.460177	0.708333	0.577778	0.7998
Sei Kera Hilir I	0.697987	0.469027	0.291667	1	0.49268
Sei Kera Hilir II	0	0	0.041667	0	0
Sei Kera Hulu	0.134228	0.123894	0	0.155556	0.26336
Sidodadi	0.147651	0.132743	0.333333	0	0.780405
Sidorame Barat I	0.228188	0.19469	0.291667	0.133333	0.824558
Sidorame Barat II	0.127517	0.044248	0.041667	0.311111	0.484982
Sidorame Timur	0.208054	0.150442	0.333333	0.155556	0.69132
Tegal Rejo	0.630872	0.362832	0.375	1	0.939868

Gambar 4.22 berikut menunjukkan perbandingan distribusi nilai fitur sebelum dan sesudah normalisasi. Setelah normalisasi, semua fitur berada pada rentang 0–1 sehingga tidak ada fitur yang mendominasi perhitungan jarak dalam algoritma klusterisasi.



Gambar 4.22 Perbandingan Data Sebelum Dan Sesudah Normalisasi In-Max

Berdasarkan seluruh perhitungan normalisasi yang telah dilakukan terhadap 20 kelurahan dan 5 fitur, diperoleh beberapa temuan penting. Secara keseluruhan, seluruh nilai berhasil dikonversi ke dalam rentang 0.0000 hingga 1.0000, yang diverifikasi dengan memastikan bahwa nilai minimum setiap fitur menghasilkan angka 0 dan nilai maksimumnya menghasilkan angka 1.

Pada fitur `total_kriminal`, Kelurahan Glugur Darat I mencatatkan nilai normalisasi tertinggi sebesar 1,0000 dengan 162 kasus, sementara Kelurahan Sei Kera Hilir II mendapat nilai terendah 0,0000 dengan hanya 13 kasus. Untuk fitur `curanmor`, nilai tertinggi 1,0000 yaitu Glugur Darat I yang mencatat 117 kasus, sedangkan Sei Kera Hilir II menempati posisi terendah dengan nilai 0,0000 pada 4 kasus.

Pada fitur `curas`, Kelurahan Gang Buntu unggul dengan nilai normalisasi 1,0000 atas 25 kasus, sementara Sei Kera Hulu mencatat nilai terendah 0,0000 dengan hanya 1 kasus. Adapun pada fitur `curat`, dua kelurahan secara bersama-sama

memperoleh nilai tertinggi 1,0000, yakni Sei Kera Hilir I, dan Tegal Rejo yang masing-masing mencatat 52 kasus, sedangkan Kelurahan Sidodadi dan Sei Kera Hilir II berada di posisi terendah dengan nilai 0,0000 pada 7 kasus. Terakhir, pada fitur rata_waktu, Kelurahan Pulo Brayan Darat I memperoleh nilai tertinggi 1,0000 dengan kategori waktu 2 yaitu siang, sedangkan Kelurahan Sei Kera Hilir II mendapat nilai terendah 0,0000 dengan kategori waktu 1 yaitu pagi.

4.3. Implementasi Algoritma K-Means

4.3.1 Penentuan WSSE dan membuat (Metode elbow)

Sebelum menjalankan algoritma K-Means, nilai K (jumlah kluster) yang optimal harus ditentukan terlebih dahulu. WSSE (Within Sum of Square Error) adalah jumlah total kuadrat jarak antara setiap data dengan centroid klasternya masing-masing. Semakin kecil nilai WSSE, semakin kompak kluster yang terbentuk. Metode Elbow digunakan dengan menghitung nilai WSSE untuk berbagai nilai K, kemudian memilih K pada titik siku (elbow) grafik WSSE. WSSE dihitung dengan rumus pada persamaan 2.2.

1. Hitung WSSE untuk setiap K ($k=2$ sampai $k=8$)

K-Means dijalankan 7 kali untuk $K=2,3,4,5,6,7,8$. Setiap iterasi menyimpan nilai inerti, yang merupakan WSSE (Within Sum of Squared Error). Berikut code menghitung WSSE untuk setiap k pada program python.

```

# PHASE 3: BUAT GRAFIK ELBOW
fig, axes = plt.subplots(1, 2, figsize=(14, 6))
fig.suptitle('Metode Elbow untuk Menentukan Nilai K Optimal',
             fontweight='bold', fontsize=14)

# --- Grafik kiri: WSSE ---
axes[0].plot(list(K_range), wsse_list, 'o-',
             color='#2196F3', linewidth=2.5, markersize=9,
             markerfacecolor='white', markeredgewidth=2.5)
axes[0].axvline(x=k_optimal, color='red', linestyle='--',
               alpha=0.7, linewidth=2, label=f'K Optimal = {k_optimal}')
axes[0].scatter([k_optimal], [wsse_list[k_optimal-2]],
               color='red', s=150, zorder=5)

for i, (k, w) in enumerate(zip(K_range, wsse_list)):
    axes[0].annotate(f'{w:.4f}', (k, w),
                    textcoords="offset points",
                    xytext=(0, 12), ha='center', fontsize=9,
                    fontweight='bold')

axes[0].set_title('Grafik WSSE vs Nilai K', fontweight='bold', fontsize=12)
axes[0].set_xlabel('Jumlah Klaster (K)', fontsize=11)
axes[0].set_ylabel('WSSE (Within Sum of Square Error)', fontsize=11)
axes[0].set_xticks(list(K_range))
axes[0].legend(fontsize=10)
axes[0].grid(True, alpha=0.3)

# --- Grafik kanan: Penurunan WSSE ---
penurunan_list = [wsse_list[i] - wsse_list[i+1]
                  for i in range(len(wsse_list)-1)]

```

Gambar 4.23 Hitung WSSE

Berdasarkan gambar tersebut, terdapat kode python untuk menghitung WSSE dan nilai K, yang menghasilkan output seperti pada tabel berikut:

Tabel 4.17 Output hitung WSSE

Nilai K	WSSE	Keterangan
2	3.3160	Penurunan besar
3	2.0742	Titik siku (Elbow) - K Optimal
4	1.4672	Penurunan mulai melambat
5	1.0661	Penurunan kecil
6	0.8716	Penurunan kecil
7	0.7113	Penurunan sangat kecil
8	0.5999	Hampir stabil

Menentukan elbow yaitu dengan menghitung selisih penurunan WSSE antar K. K yang memiliki penurunan terbesar merupakan titik elbow, serta nilai K optimal.

2. Verifikasi WSSE manual untuk K optimal

Rumus jarak kuadrat (bukan jarak biasa, tapi langsung dikuadratkan) seperti pada persamaan 2.3. Jumlahkan semua Jarak², itulah WSSE. Berikut code python hitungan manual WSSE:

```

96 print(f"PERHITUNGAN MANUAL WSSE UNTUK K={k_optimal}")
97 print(f"{' '*55}")
98
99 km_opt = KMeans(n_clusters=k_optimal, random_state=42, n_init=10)
100 km_opt.fit(X_norm)
101 labels = km_opt.labels_
102 centroids = km_opt.cluster_centers_
103
104 for i, c in enumerate(centroids):
105     print(f" Centroid C{i}: {c.round(4)}")
106
107 print(f"\n {'Kelurahan':<30} {'Klaster':>8} {'Jarak^2':>10}")
108 print(f" {'-'*28} {'-'*8} {'-'*10}")
109
110 total_wsse = 0
111 for i, row in agg.iterrows():
112     x = X_norm[i]
113     k_label = labels[i]
114     centroid = centroids[k_label]
115     jarak_kuadrat = np.sum((x - centroid) ** 2)

```

Gambar 4.24 Hitungan Manual WSSE

Berdasarkan kode pada gambar tersebut didapatkan hasil atau output pada terminal seperti berikut:

C0 = [0.3027, 0.2407, 0.3042, 0.2578, 0.7719]

C1 = [0.694, 0.5115, 0.5917, 0.7200, 0.758]

C2 = [0.1074, 0.085, 0.0583, 0.1333, 0.2131]

Kemudian untuk hasil perhitungan manual yang di dapat pada output di terminal, disusun pada tabel berikut:

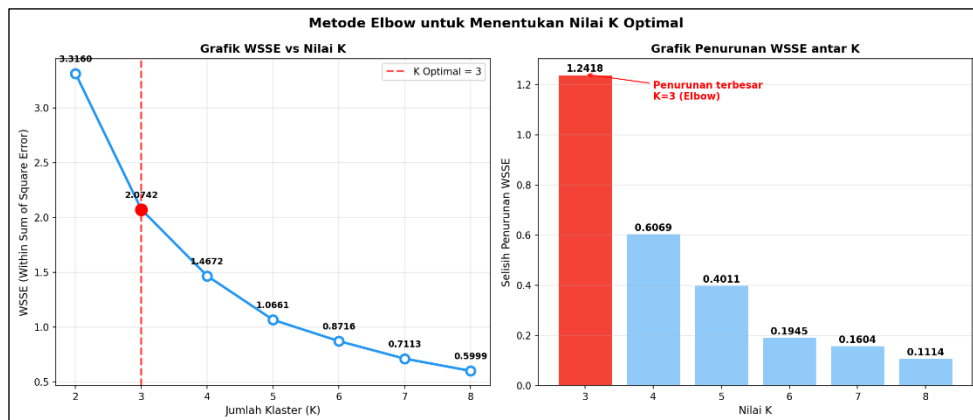
Tabel 4.18 Hasil jarak penentuan WSSE

Kelurahan	Klaster	Jarak ²
Durian	K=0	0.0199
Gaharu	K=0	0.0407
Gang Buntu	K=2	0.2896
Glugur Darat I	K=2	0.3514
Glugur Darat II	K=0	0.1047
Pahlawan	K=1	0.0073
Pandau Hilir	K=1	0.0310
Perintis	K=0	0.1167

Pulo Brayan Bengkel	K=0	0.0141
Pulo Brayan Bengkel Baru	K=0	0.0595
Pulo Brayan Darat I	K=2	0.1662
Pulo Brayan Darat II	K=2	0.0277
Sei Kera Hilir I	K=2	0.3029
Sei Kera Hilir II	K=1	0.0822
Sei Kera Hulu	K=1	0.0087
Sidodadi	K=0	0.0869
Sidorame Barat I	K=0	0.0219
Sidorame Barat II	K=1	0.1079
Sidorame Timur	K=0	0.0217
Tegal Rejo	K=2	0.2130
TOTAL WSSE		2.0742

3. Visualisasi dua grafik

Grafik menampilkan kurva WSSE dari $k=2$ hingga $k=8$. Bentuknya seperti siku lengan (elbow). Titik dimana kurva mulai mendatar adalah K optimal yang ditandai garis merah putus-putus pada gambar berikut:



Gambar 4.25 Metode Elbow untuk nilai K

Berdasarkan Tabel 4.17 dan Gambar 4.30, dapat diamati bahwa penurunan nilai WSSE paling signifikan terjadi dari $K=2$ ke $K=3$, kemudian penurunan mulai melambat setelah $K=3$. Titik siku (elbow) yang paling jelas terlihat pada $K=3$. Oleh karena itu, nilai $K=3$ dipilih sebagai jumlah kluster yang optimal dalam penelitian ini, yang secara semantik merepresentasikan tiga kategori tingkat kerawanan: Rawan Tinggi, Rawan Sedang, dan Rawan Rendah.

4.3.2 Perhitungan Manual K-Means

1. Inisialisasi Centroid Awal (Iterasi 0)

Centroid awal dipilih secara acak (random_state=42):

$C_0 = [0.3027, 0.2407, 0.3042, 0.2578, 0.7719]$

$C_1 = [0.694, 0.5115, 0.5917, 0.7200, 0.758]$

$C_2 = [0.1074, 0.085, 0.0583, 0.1333, 0.2131]$

Setiap centroid adalah vektor 5 dimensi sesuai 5 fitur: [Total_Kriminal, Curanmor, Curas, Curat, Rata_Waktu]

2. Iterasi 1: Hitung Jarak Euclidean

Inilah inti algoritma. Diulang maksimal 20 kali dengan 3 tahap di setiap iterasi:

```

210 prev_labels = None
211
212 for iterasi in range(1, 20):
213     print(f"\n{'='*65}")
214     print(f"ITERASI {iterasi}")
215     print(f"{'='*65}")
216
217     # --- Hitung jarak Euclidean ---
218     distances = np.zeros((len(X_norm), K))
219     for i in range(len(X_norm)):
220         for k in range(K):
221             distances[i, k] = np.sqrt(np.sum((X_norm[i] - centroids[k])**2))
222
223     labels = np.argmin(distances, axis=1)
224
225     # --- Tampilkan tabel jarak ---

```

	0.3442	0.6252	0.6296	0.3442	0
Perintis	0.3442	0.6252	0.6296	0.3442	0
Pulo Brayen Bengkel	0.1311	0.7013	0.6181	0.1311	0
Pulo Brayen Bengkel Baru	0.2253	0.6643	0.7702	0.2253	0
Pulo Brayen Darat I	0.5729	0.4893	1.1806	0.4893	1
Pulo Brayen Darat II	0.6499	0.2054	1.1743	0.2054	1
Sei Kera Hillir I	0.9151	0.4905	1.1748	0.4905	1
Sei Kera Hillir II	0.9385	1.4624	0.2867	0.2867	2
Sei Kera Hulu	0.6353	1.1734	0.0931	0.0931	2
Sidodadi	0.3210	1.0137	0.6474	0.3210	0
Sidorame Barat I	0.1615	0.8694	0.6745	0.1615	0
Sidorame Barat II	0.4726	1.0409	0.3284	0.3284	2
Sidorame Timur	0.1868	0.8696	0.5650	0.1868	0
Tegal Rejo	0.8407	0.4296	1.3156	0.4296	1

```

Pembaruan Centroid:
C0 = rata-rata 9 anggota = [0.2677 0.2026 0.2778 0.2519 0.7466]

```

Gambar 4. 26 Perhitungan Manual K-Means

- $distance[i,k] = np.sqrt(\dots)$ = tiap kelurahan akan dihitung jaraknya ke dalam tiga centroid, hasilnya akan ditampilkan dalam tabel $d(c_0)$, $d(c_1)$, $d(c_2)$.
- $labels = np.argmin(distance, axis=1)$ = setiap kelurahan dimasukkan ke klaster dengan jarak terpendek.

- c. $\text{New_centroid}[k] = \dots =$ centroid baru dihitung dari rata-rata koordinat semua anggota di tiap klaster.
- d. If $\text{np.array_equal}(\dots) =$ cek konvergen, jika tidak berubah, maka loop berhenti.

Contoh perhitungan manual kelurahan durian, menggunakan persamaan (2.3):

$$x = [0.2013, 0.1681, 0.2083, 0.1556, 0.7321]$$

$$C0 = [0.3027, 0.2407, 0.3042, 0.2578, 0.7719]$$

$$\begin{aligned} d(\text{Durian}, C0) &= \sqrt{[(0.2013-0.3027)^2 + (0.1681-0.2407)^2 + (0.2083-0.3042)^2 \\ &\quad + (0.1556-0.2578)^2 + (0.7321-0.7719)^2]} \\ &= \sqrt{[0.01028 + 0.00527 + 0.00920 + 0.01044 + 0.00158]} \\ &= \sqrt{0.03677} \\ &= 0.1917 \leftarrow \text{terdekat} \rightarrow \text{masuk K0} \end{aligned}$$

Kemudian dilanjutkan perhitungan manual tiap kelurahan dengan rumus sebelumnya untuk mengitung jarak dan menentukan klaster, seperti tabel berikut:

Tabel 4.19 Hasil Iterasi 1

Kelurahan	d(C0)	d(C1)	d(C2)	Minimum	→ Klaster
Durian	0.1917	0.9093	0.5551	0.1917	K0
Gaharu	0.1954	0.6874	0.7525	0.1954	K0
Gang Buntu	0.7706	0.5549	1.2591	0.5549	K1
Glugur Darat I	1.0980	0.6134	1.5276	0.6134	K1
Glugur Darat II	0.3121	0.8234	0.7238	0.3121	K0
Pahlawan	0.7025	1.1911	0.0852	0.0852	K2
Pandau Hilir	0.6991	1.2153	0.1760	0.1760	K2
Perintis	0.3442	0.6252	0.6296	0.3442	K0
Pulo Brayan Bengkel	0.1311	0.7013	0.6181	0.1311	K0
Pulo Brayan Bengkel Baru	0.2253	0.6643	0.7702	0.2253	K0
Pulo Brayan Darat I	0.5729	0.4893	1.1806	0.4893	K1
Pulo Brayan Darat II	0.6499	0.2054	1.1743	0.2054	K1
Sei Kera Hilir I	0.9151	0.4905	1.1748	0.4905	K1

Sei Kera Hilir II	0.9385	1.4624	0.2867	0.2867	K2
Sei Kera Hulu	0.6353	1.1734	0.0931	0.0931	K2
Sidodadi	0.3210	1.0137	0.6474	0.3210	K0
Sidorame Barat I	0.1615	0.8694	0.6745	0.1615	K0
Sidorame Barat II	0.4726	1.0409	0.3284	0.3284	K2
Sidorame Timur	0.1868	0.8696	0.5650	0.1868	K0
Tegal Rejo	0.8407	0.4296	1.3156	0.4296	K1

3. Perbarui Centroid setelah Iterasi 1

Centroid baru = rata-rata semua anggota klaster:

- a. C0 baru (9 anggota: Durian, Gaharu, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Timur):

$$C0 \text{ baru} = [0.2677, 0.2026, 0.2778, 0.2519, 0.7466]$$

- b. C1 baru (6 anggota: Gang Buntu, Glugur Darat I, Pulo Brayan Darat I, Pulo Brayan Darat II, Sei Kera Hilir I, Tegal Rejo):

$$C1 \text{ baru} = [0.6812, 0.5236, 0.5833, 0.6519, 0.7983]$$

- c. C2 baru (5 anggota: Pahlawan, Pandau Hilir, Sei Kera Hilir II, Sei Kera Hulu, Sidorame Barat II):

$$C2 \text{ baru} = [0.1074, 0.085, 0.0583, 0.1333, 0.2131]$$

4. Iterasi 2: Hitung Ulang dengan Centroid Baru

Dengan centroid baru, hitung ulang jarak semua data. Hasilnya penugasan klaster tidak berubah maka algoritma konvergen di Iterasi 2.

Tabel 4.20 Hasil Iterasi 2

Kelurahan	d(C0)	d(C1)	d(C2)	Minimum	→ Klaster
Durian	0.1410	0.8648	0.5551	0.1410	K0
Gaharu	0.2019	0.6510	0.7525	0.2019	K0
Gang Buntu	0.8100	0.5381	12.591	0.5381	K1
Glugur Darat I	11.544	0.5928	15.276	0.5928	K1
Glugur Darat II	0.3236	0.7732	0.7238	0.3236	K0
Pahlawan	0.6563	11.774	0.0852	0.0852	K2

Pandau Hilir	0.6575	11.935	0.1760	0.1760	K2
Perintis	0.3416	0.6129	0.6296	0.3416	K0
Pulo Brayan Bengkel	0.1189	0.6670	0.6181	0.1189	K0
Pulo Brayan Bengkel Baru	0.2440	0.6249	0.7702	0.2440	K0
Pulo Brayan Darat I	0.6366	0.4077	11.806	0.4077	K1
Pulo Brayan Darat II	0.7019	0.1663	11.743	0.1663	K1
Sei Kera Hilir I	0.9384	0.5504	11.748	0.5504	K1
Sei Kera Hilir II	0.8884	14.470	0.2867	0.2867	K2
Sei Kera Hulu	0.5865	11.540	0.0931	0.0931	K2
Sidodadi	0.2949	0.9619	0.6474	0.2949	K0
Sidorame Barat I	0.1481	0.8173	0.6745	0.1481	K0
Sidorame Barat II	0.4152	10.218	0.3284	0.3284	K2
Sidorame Timur	0.1473	0.8267	0.5650	0.1473	K0
Tegal Rejo	0.8741	0.4616	13.156	0.4616	K1

Berdasarkan hasil perhitungan jarak pada Iterasi 2 sebagaimana ditampilkan pada Tabel 4.20, setiap kelurahan kembali ditetapkan ke cluster dengan jarak minimum. Hasilnya menunjukkan bahwa Durian, Gaharu, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, dan Sidorame Timur masuk ke dalam Klaster K0, yang merepresentasikan wilayah dengan tingkat kriminalitas sedang.

Setelah membandingkan hasil pengelompokan pada Iterasi 2 dengan Iterasi 1, diperoleh kesimpulan bahwa tidak terdapat perubahan penugasan klaster pada seluruh kelurahan. Karena komposisi anggota setiap cluster tidak mengalami perubahan, maka nilai centroid juga tidak akan berubah apabila dihitung kembali. Kondisi ini menandakan bahwa algoritma K-Means telah konvergen pada Iterasi 2, sehingga proses iterasi dihentikan dan hasil pengelompokan pada iterasi ini ditetapkan sebagai hasil akhir clustering.

Tabel 4.21 Hasil Akhir K-Means

Kelurahan	Klaster	Kategori	SSE
Gang Buntu	K1	Rawan Tinggi	0.2896

Glugur Darat I	K1	Rawan Tinggi	0.3514
Pulo Brayan Darat I	K1	Rawan Tinggi	0.1662
Pulo Brayan Darat II	K1	Rawan Tinggi	0.0277
Sei Kera Hilir I	K1	Rawan Tinggi	0.3029
Tegal Rejo	K1	Rawan Tinggi	0.2130
Durian	K0	Rawan Sedang	0.0199
Gaharu	K0	Rawan Sedang	0.0407
Glugur Darat II	K0	Rawan Sedang	0.1047
Perintis	K0	Rawan Sedang	0.1167
Pulo Brayan Bengkel	K0	Rawan Sedang	0.0141
Pulo Brayan Bengkel Baru	K0	Rawan Sedang	0.0595
Sidodadi	K0	Rawan Sedang	0.0869
Sidorame Barat I	K0	Rawan Sedang	0.0219
Sidorame Timur	K0	Rawan Sedang	0.0217
Pahlawan	K2	Rawan Rendah	0.0073
Pandau Hilir	K2	Rawan Rendah	0.0310
Sei Kera Hilir II	K2	Rawan Rendah	0.0822
Sei Kera Hulu	K2	Rawan Rendah	0.0087
Sidorame Barat II	K2	Rawan Rendah	0.1079
Total SSE			2.0742

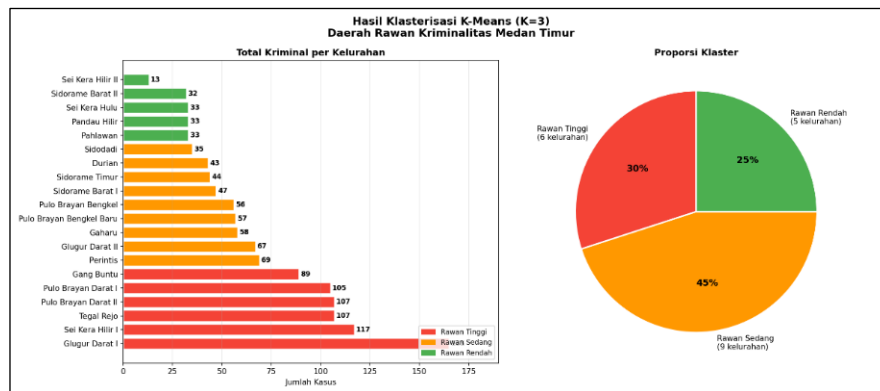
Berdasarkan Tabel 4.21, Klaster 1 (Rawan Tinggi) didominasi oleh kelurahan-kelurahan dengan jumlah kasus di atas 89 kasus, meliputi Glugur Darat I sebagai yang tertinggi, diikuti Sei Kera Hilir I, serta Tegal Rejo, Pulo Brayan Darat II, Pulo Brayan Darat I, dan Gang Buntu. Kelurahan-kelurahan ini merupakan prioritas utama bagi pihak kepolisian dalam mengintensifkan patroli.

Klaster 0 (Rawan Sedang) terdiri dari 9 kelurahan dengan rentang 35-69 kasus, sedangkan Klaster 2 (Rawan Rendah) mencakup kelurahan dengan jumlah kasus antara 13-33 kasus.

5. Visualisasi Hasil Implementasi Algoritma K-Means

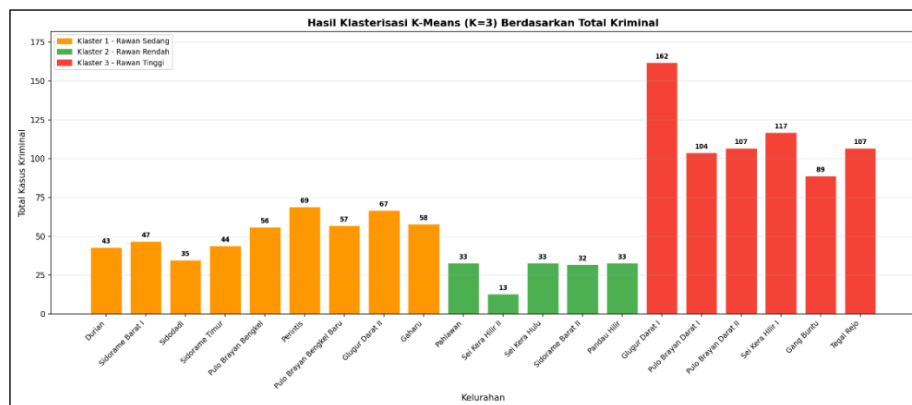
Hasil klasterisasi K-Means dengan $K=3$ terhadap 20 kelurahan di Kecamatan Medan Timur menghasilkan tiga kelompok tingkat kerawanan kriminalitas, yaitu Rawan Tinggi, Rawan Sedang, dan Rawan Rendah. Pembagian klaster didasarkan pada lima fitur utama yakni Total Kriminal, Curanmor, Curas,

Curat, dan Rata-rata Waktu Kejadian. Berikut hasil visualisasi klusterisasi k-means dengan nilai $K=3$.



Gambar 4.27 Hasil klusterisasi K-Means

Visualisasi klaster menunjukkan bahwa sebagian besar kelurahan (45%) berada pada tingkat rawan sedang, dengan hanya 25% yang tergolong aman. Hal ini mengindikasikan bahwa permasalahan kriminalitas di Kecamatan Medan Timur cukup merata dan membutuhkan pendekatan penanganan yang terdiferensiasi sesuai karakteristik masing-masing klaster.



Gambar 4.28 Hasil klusterisasi berdasarkan total kriminal

Grafik ini secara visual memperlihatkan kesenjangan yang jelas antara kelurahan rawan tinggi dan rawan rendah. Glugur Darat I dengan 162 kasus memiliki jumlah hampir 12 kali lipat dibanding Sei Kera Hilir II yang hanya 13 kasus. Hal ini mengindikasikan bahwa distribusi kriminalitas di Kecamatan

Medan Timur tidak merata, sehingga diperlukan kebijakan keamanan yang terfokus dan terdiferensiasi terutama pada kelurahan-kelurahan di klaster Rawan Tinggi.

4.4. Implementasi Algoritma DBSCAN

4.4.1. Penentuan parameter DBSCAN

Algoritma DBSCAN memerlukan dua parameter utama: epsilon (eps) yang menentukan radius lingkungan suatu titik, dan minPts yang menentukan jumlah minimum titik dalam radius eps untuk membentuk core point. Penentuan nilai eps optimal dilakukan menggunakan metode K-Nearest Neighbor (KNN) Distance Plot dengan $k=3$ (sesuai nilai minPts).

Dari KNN Distance Plot, titik yang menunjukkan kenaikan tajam (elbow) berada di sekitar nilai 0.2, sehingga $\text{eps}=0.2$ dipilih sebagai parameter optimal. Kombinasi parameter yang diuji beserta hasil evaluasinya dirangkum sebagai berikut: pada $\text{eps}=0.2$ dengan $\text{minPts}=2$ menghasilkan 2 klaster dengan 11 noise dan Silhouette Index 0.6708, yang merupakan nilai terbaik di antara semua kombinasi yang diuji. Nilai $\text{minPts}=2$ dipilih mengingat jumlah total objek yang relatif kecil (20 kelurahan), sehingga threshold yang terlalu tinggi akan menyebabkan terlalu banyak noise.

1. Tentukan minPts

Nilai minPts ditentukan berdasarkan aturan baku dalam literatur DBSCAN, dijelaskan dalam tabel berikut:

Tabel 4.22 Aturan Baku DBSCAN

Aturan	Rumus	Hasil di data kamu
Rumus umum	$\text{minPts} \geq \text{dimensi} + 1$	$5 + 1 = 6$

(Ester et al.)		
Rumus alternatif	$\text{minPts} = 2 \times \text{dimensi}$	$2 \times 5 = 10$
Untuk dataset kecil	$\text{minPts} = 2$ (minimum)	2

Data hanya memiliki 20 objek (kelurahan). Jika $\text{minPts} = 6$ atau 10, hampir semua titik akan jadi noise karena sulit menemukan 6–10 tetangga dalam radius kecil. Oleh karena itu $\text{minPts} = 2$ dipilih sebagai nilai yang paling sesuai untuk dataset kecil.

Berikut penentuan minPts pada program python:

```
print("[PHASE 5] PENENTUAN PARAMETER eps (KNN Distance Plot)")
print("-"*70)

kelurahan = agg['kelurahan'].tolist()
n = len(agg)

k_nn = 2 # sesuai minPts - 1
nbrs = NearestNeighbors(n_neighbors=k_nn).fit(X_norm)
distances_knn, _ = nbrs.kneighbors(X_norm)
knn_dist = np.sort(distances_knn[:, k_nn-1])

print(f"\nJarak ke-{k_nn} tetangga terdekat (diurutkan):")
for i, d in enumerate(knn_dist):
    print(f" Titik ke-{i+1:2d}: {d:.4f}")

# Grafik KNN Distance
plt.figure(figsize=(8, 5))
plt.plot(range(1, n+1), knn_dist, 'o-', color='r', linewidth=2, markersize=7)
plt.axhline(y=0.2, color='red', linestyle='--', linewidth=2, label='eps = 0.2 (titik elbow)')
plt.title('KNN Distance Plot untuk Menentukan Nilai eps', fontweight='bold', fontsize=13)
plt.xlabel('Titik Data (diurutkan)', fontsize=11)
```

Gambar 4.29 Code Python minPts

Dengan menghasilkan output diterminal sebagai berikut :

```
LANGKAH 1: PENENTUAN minPts
=====
Jumlah fitur (dimensi) = 5
Jumlah objek (n) = 20
Rumus umum : minPts >= dimensi+1 = 6
Untuk dataset kecil : minPts = 2 (minimum)
=> minPts dipilih = 2
```

Gambar 4.30 Output minPts

Penentuan nilai parameter minPts dilakukan menggunakan pendekatan berbasis dimensi data. Berdasarkan kode Python pada Gambar 4.34, nilai minPts ditetapkan menggunakan rumus $\text{minPts} = \text{dimensi} + 1$, di mana jumlah fitur (dimensi) yang digunakan adalah 5, sehingga diperoleh nilai minPts minimum sebesar 6. Namun karena dataset yang digunakan tergolong kecil dengan jumlah objek sebanyak 20 kelurahan, maka nilai minPts disesuaikan menjadi 2 sesuai ketentuan untuk dataset kecil.

2. Tentukan eps (KNN Distance Plot)

Nilai eps ditentukan menggunakan KNN Distance Plot dengan langkah berikut:

a. Hitung jarak ke- k tetangga terdekat ($k = \text{minPts}$)

Untuk setiap titik, hitung jarak ke tetangga terdekat ke-2 (sesuai $\text{minPts}=2$),

lalu urutkan dari kecil ke besar:

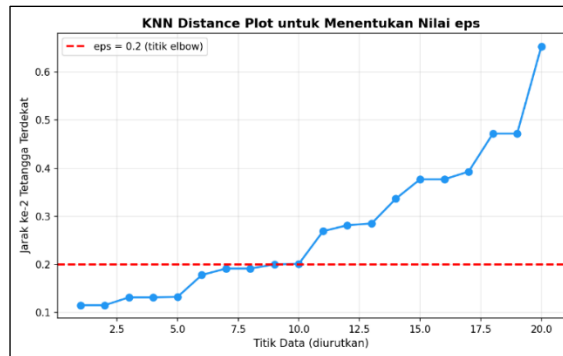
Tabel 4.23 $k = \text{minPts}$

No	Kelurahan	Jarak ke-2 Tetangga
1	Pahlawan	0.1155
2	Sei Kera hulu	0.1155
3	Durian	0.1319
4	Sidorame Timur	0.1319
5	Sidorame Barat I	0.1329
6	Sidodadi	0.1783
7	Pandau Hilir	0.1917
8	Pulo Brayan Bengkel	0.1917
9	Pulo Brayan Bengkel Baru	0.2002
10	Gaharu	0.2014 (kenaikan mulai tajam)
11	Perintis	0.2693
...
20	Glugur Darat I	0.6534

Jarak ke-2 tetangga terdekat dari setiap kelurahan, diurutkan dari jarak terkecil ke terbesar, sebagai dasar penentuan titik elbow untuk parameter eps DBSCAN.

- b. Cari titik siku (elbow) pada grafik

Grafik KNN Distance Plot menunjukkan kenaikan tajam mulai di sekitar nilai 0.20. Titik siku ini menunjukkan batas antara titik yang berdekatan (dalam kluster) dan titik yang jauh (noise). Oleh karena itu $\text{eps} = 0.20$ dipilih.



Gambar 4.31 Penentuan Parameter DBSCAN

Nilai $\text{eps} = 0.20$ dipilih sebagai parameter optimal karena memenuhi dua kriteria sekaligus. Pertama, sesuai dengan titik elbow pada KNN Distance Plot, dan kedua, menghasilkan Silhouette Index sebesar 0.667 yang berada di atas batas SI baik (0.5) dengan jumlah noise yang masih wajar sebanyak 11 titik. Nilai ini mencerminkan keseimbangan terbaik antara kualitas kluster dan cakupan data yang terklusterisasi.

3. Validasi dengan Eksperimen Semua Kombinasi

Setelah mendapat kandidat eps dari KNN plot, dilakukan eksperimen untuk semua kombinasi eps dan minPts dan dievaluasi menggunakan Silhouette Index:

Tabel 4.24 Hasil SI dari eps dan minPts

eps	minPts	Kluster	Noise	SI	Keterangan
0.15	2	2	15	0.7897	SI tertinggi tapi noise terlalu banyak (75%)
0.18	2	2	14	0.7640	Noise masih sangat banyak (70%)

0.20	2	2	11	0.6708	Seimbang, maka dipilih
0.20	3	2	11	0.6708	Sama dengan minPts=2
0.22	2	2	10	0.6539	SI lebih rendah
0.30	2	2	7	0.5336	Noise berkurang tapi SI turun
0.40	2	2	3	0.4296	Terlalu longgar, kluster tidak representatif

Meskipun $\text{eps}=0.15$ menghasilkan SI lebih tinggi (0.7897), nilai ini menghasilkan 15 dari 20 kelurahan (75%) sebagai noise, terlalu banyak data yang tidak terkluster sehingga tidak informatif. $\text{eps}=0.20$ memberikan keseimbangan terbaik antara kualitas kluster ($\text{SI}=0.6708$, kategori "baik") dan jumlah data yang berhasil dikelompokkan.

4.4.2. Perhitungan Manual DBSCAN

1. Parameter dan Konsep Dasar

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) adalah algoritma klusterisasi berbasis kepadatan yang tidak memerlukan jumlah kluster ditentukan di awal. Algoritma ini menggunakan dua parameter utama:

Tabel 4.25 Parameter DBSCAN

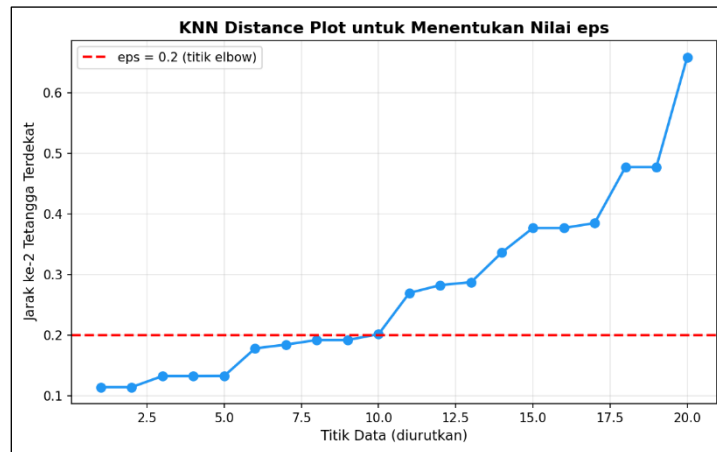
Parameter	Nilai	Keterangan
eps (ϵ)	0.20	Radius lingkungan maksimum suatu titik
minPts	2	Minimum titik dalam radius eps untuk jadi Core Point

Setiap titik diklasifikasikan menjadi tiga jenis berdasarkan parameter tersebut:

Tabel 4.26 Jenis Titik parameter

Jenis Titik	Syarat	Keterangan
Core Point	$ N(p) + 1 \geq \text{minPts}$	Inti kluster, memiliki cukup tetangga
Border Point	$ N(p) + 1 < \text{minPts}$	Berada dalam jangkauan Core Point
Noise / Outlier	$ N(p) + 1 < \text{minPts}$	Tidak terjangkau oleh Core Point manapun

Grafik KNN Distance Plot di atas menampilkan jarak ke-2 tetangga terdekat dari setiap kelurahan yang telah diurutkan dari terkecil ke terbesar, sebagai metode visual untuk menentukan nilai eps optimal pada algoritma DBSCAN.



Gambar 4.32 Grafik Elbow nilai eps

Berdasarkan metode elbow ini, $\text{eps} = 0.20$ ditetapkan sebagai nilai optimal, karena titik-titik di bawah garis merah cenderung membentuk kluster, sedangkan titik-titik di atasnya berpotensi dikategorikan sebagai noise oleh algoritma DBSCAN.

2. Perhitungan Jarak Euclidean

Sebelum menentukan tetangga setiap titik, dihitung terlebih dahulu jarak Euclidean antara semua pasangan kelurahan menggunakan rumus pada persamaan (2.5). Berikut adalah empat contoh perhitungan manual jarak Euclidean untuk pasangan titik yang menjadi tetangga ($\text{jarak} \leq \text{eps} = 0.20$) dan yang bukan tetangga:

Contoh 1:

Tabel 4.27 d(Durian, Sidorame Barat I)

Titik	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Durian (p)	0.2013	0.1681	0.2083	0.1556	0.7321

Sidorame Barat I (q)	0.2282	0.1947	0.2917	0.1333	0.8246
(p-q) ²	0.0007	0.0007	0.0069	0.0005	0.0085

$$d(\text{Durian, Sidorame Barat I}) = \sqrt{(0.0007 + 0.0007 + 0.0069 + 0.0005 + 0.0085)}$$

$$= \sqrt{0.0174} = 0.1319 \leq 0.20 = \text{TETANGGA}$$

Contoh 2:

Tabel 4.28 d(Durian, Sidorame Timur)

Titik	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Durian (p)	0.2013	0.1681	0.2083	0.1556	0.7321
Sidorame Timur (q)	0.2080	0.1504	0.3333	0.1556	0.6913
(p-q) ²	0.000044	0.000313	0.01562	0.0000	0.0017

$$d(\text{Durian, Sidorame Timur}) = \sqrt{(0.00004 + 0.00031 + 0.01562 + 0.0000 + 0.00178)}$$

$$= \sqrt{0.01775} = 0.1332 \leq 0.20 = \text{TETANGGA}$$

Contoh 3:

Tabel 4.29 d(Pahlawan, Pandau Hilir)

Titik	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Pahlawan (p)	0.14094	0.115044	0.041667	0.177778	0.158553
Pandau Hilir (q)	0.134228	0.141593	0.166667	0.022222	0.158553
(p-q) ²	0.000045	0.00070	0.015625	0.02418	0.000000

$$d(\text{Pahlawan, Pandau Hilir}) = \sqrt{(0.00004 + 0.00070 + 0.01562 + 0.01777 + 0.0000)}$$

$$= \sqrt{0.034072} = 0.1846 \leq 0.20 = \text{TETANGGA}$$

Contoh 4:

Tabel 4.30 d(Pahlawan, Sei Kera Hulu)

Titik	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Pahlawan (p)	0.14094	0.115044	0.041667	0.177778	0.158553
Sei Kera Hulu (q)	0.134228	0.123894	0	0.155556	0.26336
$(p-q)^2$	0.000045	0.000078	0.041667	0.00049	0.010983

$$\begin{aligned}
 d(\text{Pahlawan, Sei Kera Hulu}) &= \sqrt{(0.00004 + 0.00007 + 0.04166 + 0.0004 + \\
 &\quad 0.01098)} \\
 &= \sqrt{0.013556} = 0.1164 \leq 0.20 = \text{TETANGGA}
 \end{aligned}$$

Contoh 5:

Tabel 4.31 d(Durian, Gaharu)

Titik	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu
Durian (p)	0.2013	0.1681	0.2083	0.1556	0.7321
Gaharu (q)	0.302013	0.185841	0.25	0.422222	0.844012
$(p-q)^2$	0.009739	0.000297	0.001736	0.071111	0.013381

$$\begin{aligned}
 d(\text{Durian, Gaharu}) &= \sqrt{(0.009739 + 0.000297 + 0.001736 + 0.071111 + \\
 &\quad 0.013381)} \\
 &= \sqrt{0.096264} = 0.3103 > 0.20 = \text{BUKAN TETANGGA}
 \end{aligned}$$

4. Penentuan Eps-Neighborhood dan Status Titik

Setelah semua jarak dihitung, ditentukan tetangga setiap titik dalam radius $\text{eps} = 0.20$. Jumlah tetangga ditambah satu (dirinya sendiri) dibandingkan dengan $\text{minPts} = 2$ untuk menentukan status titik.

Tabel 4.32 Status Titik Tetangga

No	Kelurahan	Tetangga dalam eps (jarak)	$ N(p) $	$ N +1$	Status
----	-----------	----------------------------	----------	---------	--------

1	Durian	Sidorame Barat I, Sidorame Timur	2	3	CORE POINT
2	Gaharu	—	0	1	NOISE
3	Gang Buntu	—	0	1	NOISE
4	Glugur Darat I	—	0	1	NOISE
5	Glugur Darat II	—	0	1	NOISE
6	Pahlawan	Pandau Hilir (0.1846), Sei Kera Hulu (0.1164)	2	3	CORE POINT
7	Pandau Hilir	Pahlawan (0.1846)	1	2	CORE POINT
8	Perintis	—	0	1	NOISE
9	Pulo Brayan Bengkel	Pulo Brayan Baru (0.1931), Sidorame Timur (0.1976)	2	3	CORE POINT
10	Pulo Brayan Baru	Pulo Brayan Bengkel (0.1931)	1	2	CORE POINT
11	Pulo Brayan Darat I	—	0	1	NOISE
12	Pulo Brayan Darat II	—	0	1	NOISE
13	Sei Kera Hilir I	—	0	1	NOISE
14	Sei Kera Hilir II	—	0	1	NOISE
15	Sei Kera Hulu	Pahlawan (0.1164)	1	2	CORE POINT
16	Sidodadi	Sidorame Barat I (0.1774), Sidorame Timur (0.1910)	2	3	CORE POINT
17	Sidorame Barat I	Durian (0.1339), Sidodadi (0.1774), Sidorame Timur (0.1532)	3	4	CORE POINT
18	Sidorame Barat II	—	0	1	NOISE
19	Sidorame Timur	Durian (0.1332), PB Bengkel (0.1976), Sidodadi (0.1910), Sidorame Barat I (0.1532)	4	5	CORE POINT
20	Tegal Rejo	—	0	1	NOISE

5. Proses Pembentukan Kluster (*Expand Cluster*)

DBSCAN memproses setiap titik secara berurutan. Jika suatu titik adalah Core Point dan belum memiliki kluster, algoritma membentuk kluster baru dan memperluas keanggotaannya melalui fungsi *Expand Cluster*.

Tabel 4.33 Proses Pembentukan Kluster DBSCAN

No Titik	Kelurahan	Jumlah Tetangga	Status
0	Durian	3 titik	CORE

	→ Sidorame Barat I		Tetangga
	→ Sidorame Timur		Tetangga
1	Gaharu	1 titik	NOISE
2	Gang Buntu	1 titik	NOISE
3	Glugur Darat I	1 titik	NOISE
4	Glugur Darat II	1 titik	NOISE
5	Pahlawan	2 titik	CORE
	→ Sei Kera Hulu		Tetangga
6	Pandau Hilir	1 titik	NOISE
7	Perintis	1 titik	NOISE
8	Pulo Brayan Bengkel	3 titik	CORE
	→ Pulo Brayan Bengkel Baru		Tetangga
	→ Sidorame Timur		Tetangga
9	Pulo Brayan Bengkel Baru	2 titik	CORE
	→ Pulo Brayan Bengkel		Tetangga
10	Pulo Brayan Darat I	1 titik	NOISE
11	Pulo Brayan Darat II	1 titik	NOISE
12	Sei Kera Hilir I	1 titik	NOISE
13	Sei Kera Hilir II	1 titik	NOISE
14	Sei Kera Hulu	2 titik	CORE
	→ Pahlawan		Tetangga
15	Sidodadi	3 titik	CORE
	→ Sidorame Barat I		Tetangga
	→ Sidorame Timur		Tetangga
16	Sidorame Barat I	4 titik	CORE
	→ Durian		Tetangga
	→ Sidodadi		Tetangga
	→ Sidorame Timur		Tetangga
17	Sidorame Barat II	1 titik	NOISE
18	Sidorame Timur	5 titik	CORE
	→ Durian		Tetangga
	→ Pulo Brayan Bengkel		Tetangga
	→ Sidodadi		Tetangga
	→ Sidorame		Tetangga
19	Tegal Rejo	1	NOISE

6. Hasil Akhir Klasterisasi DBSCAN

Setelah seluruh titik diproses membentuk klaster, diperoleh hasil klasterisasi sebagai berikut:

Tabel 4.34 Hasil Akhir Klasterisasi DBSCAN

N o.	Kelurahan	Total_Kriminal	Curanmor	Curas	Curat	Rata_Waktu	DBSCAN_Label
1	Durian	43	23	6	14	2,55814	0
2	Gaharu	58	25	7	26	2,655172	-1
3	Gang Buntu	89	34	25	30	2,629213	-1

4	Glugur Darat I	162	117	15	30	2,567901	-1
5	Glugur Darat II	67	48	2	17	2,626866	-1
6	Pahlawan	34	17	2	15	2,060606	1
7	Pandau Hilir	33	20	5	8	2,060606	-1
8	Perintis	69	33	6	30	2,42029	-1
9	Pulo Brayan Bengkel	56	25	9	22	2,517857	0
10	Pulo Brayan Bengkel Baru	57	22	13	22	2,596491	0
11	Pulo Brayan Darat I	105	70	14	21	2,790476	-1
12	Pulo Brayan Darat II	107	56	18	33	2,616822	-1
13	Sei Kera Hilir I	117	57	8	52	2,350427	-1
14	Sei Kera Hilir II	13	4	2	7	1,923077	-1
15	Sei Kera Hulu	33	18	1	14	2,151515	1
16	Sidodadi	35	19	9	7	2,6	0
17	Sidorame Barat I	47	26	8	13	2,638298	0
18	Sidorame Barat II	32	9	2	21	2,34375	-1
19	Sidorame Timur	44	21	9	14	2,522727	0
20	Tegal Rejo	107	45	10	52	2,738318	-1

Berdasarkan tabel hasil akhir dbscan pada tabel 4.31, maka pada tabel berikut dapat ringkasan hasil algoritma DBSCAN

Tabel 4.35 Ringkasan Hasil DBSCAN

Klaster	Kategori	Jumlah Kelurahan	Anggota
Klaster 0	Rawan Sedang	6 kelurahan	Durian, PB Bengkel, PB Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Timur

Klaster 1	Rawan Rendah	2 kelurahan	Pahlawan, Sei Kera Hulu
Noise / Outlier	Tidak Terklaster	12 kelurahan (55%)	Pandau Hilir, Gaharu, Gang Buntu, Glugur Darat I/II, Perintis, PB Darat I/II, Sei Kera Hilir I/II, Sidorame Barat II, Tegal Rejo
TOTAL	-	20 kelurahan	-

Sebanyak 12 kelurahan (55%) diklasifikasikan sebagai noise bukan berarti kelurahan tersebut tidak penting. Justru sebaliknya, kelurahan seperti Glugur Darat I (165 kasus), Tegal Rejo (107 kasus), Sei Kera Hilir I (117 kasus) merupakan kelurahan dengan tingkat kriminalitas tinggi namun memiliki profil kriminalitas yang sangat unik dan berbeda jauh dari kelurahan lainnya (jarak Euclidean > 0.20 ke semua tetangga). Dalam konteks DBSCAN, titik noise justru merupakan anomali yang paling memerlukan perhatian khusus dari pihak kepolisian karena tidak mengikuti pola umum.

1. Visualisasi Hasil DBSCAN

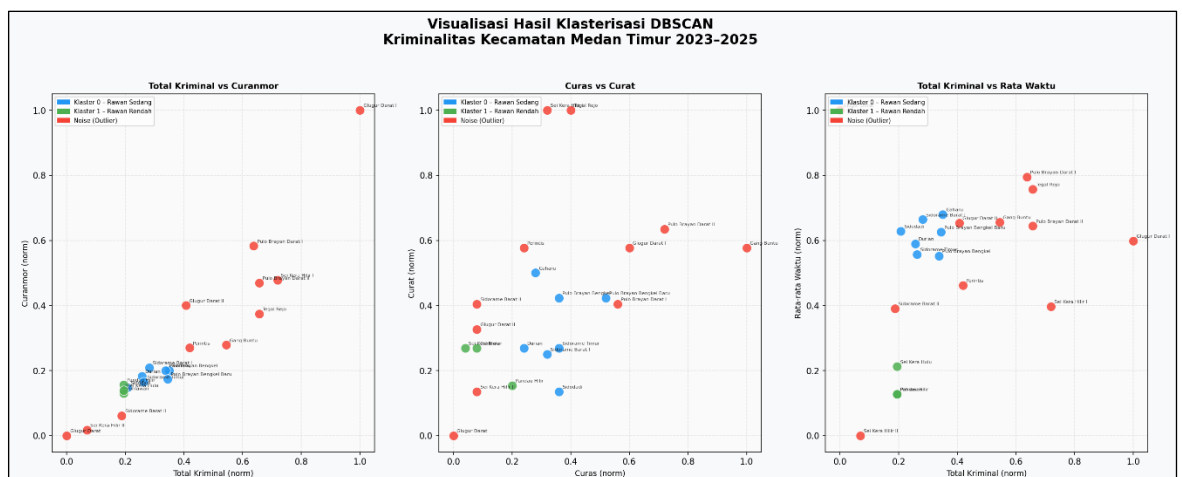
Hasil klasterisasi DBSCAN dengan parameter $\text{eps}=0.2$ dan $\text{minPts}=2$ menghasilkan 2 klaster aktif dan 1 kelompok noise dari 20 kelurahan di Kecamatan Medan Timur, dengan rincian: Noise/Outlier (12 kelurahan), Klaster 0 Rawan Sedang (6 kelurahan), dan Klaster 1 Rawan Rendah (2 kelurahan).

Klaster -1 (Outlier), Glugur Darat I (161 kasus) dan Pulo Brayon Darat I (105 kasus) diidentifikasi sebagai outlier oleh DBSCAN karena pola kriminalitasnya sangat berbeda dari kelurahan lain — khususnya dominasi ekstrim kasus Curanmor. Kondisi ini justru menandakan kedua kelurahan tersebut merupakan titik panas (hotspot) kriminalitas yang sangat signifikan, sehingga

dikategorikan sebagai Daerah Sangat Rawan. Tegal Rejo (4 kasus) juga menjadi outlier karena terlalu sedikit kasus untuk masuk cluster manapun.

Klaster 1 (Rawan Tinggi), Gang Buntu (89 kasus) dan Pulo Brayan Darat II (107 kasus) membentuk Klaster 1. Keduanya memiliki karakteristik serupa berupa tingginya kasus Curas (kekerasan) dan Curat dibandingkan kelurahan lain, dengan kejadian yang sering terjadi pada malam hari. Klaster ini dikategorikan sebagai Daerah Rawan Tinggi.

Klaster 0 (Rawan Sedang), Tujuh kelurahan (Durian, Gaharu, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi) membentuk Klaster 0. Kelurahan-kelurahan ini memiliki tingkat kriminalitas menengah dengan distribusi jenis kejahatan yang lebih merata. Klaster ini dikategorikan sebagai Daerah Rawan Sedang.



Gambar 4.33 Visualisasi klusterisasi DBSCAN

Jumlah Klaster	Ditentukan di awal (K=3)	Ditentukan otomatis oleh algoritma
Normalisasi Data	MinMaxScaler	MinMaxScaler
Fitur yang Digunakan	Total_Kriminal, Curanmor, Curas, Curat, Rata Waktu	Total_Kriminal, Curanmor, Curas, Curat, Rata Waktu

4.5.2. Hasil Klasterisasi Algoritma

Algoritma K-Means dengan K=3 berhasil mengelompokkan seluruh 20 kelurahan ke dalam tiga klaster yang merepresentasikan tingkat kerawanan kriminalitas. Tabel berikut menyajikan hasil klasterisasi lengkap beserta nilai fitur masing-masing kelurahan.

Tabel 4.37 Hasil Klasterisasi K-means

Kelurahan	Total	Curanmor	Curas	Curat	Avg Waktu	Klaster	Kategori
Gang Buntu	89	34	25	30	2.63	1	Rawan Tinggi
Glugur Darat I	162	117	15	30	2.57	1	Rawan Tinggi
Pulo Brayon Darat I	105	70	14	21	2.79	1	Rawan Tinggi
Pulo Brayon Darat II	107	56	18	33	2.62	1	Rawan Tinggi
Sei Kera Hilir I	117	57	8	52	2.35	1	Rawan Tinggi
Tegal Rejo	107	45	10	52	2.74	1	Rawan Tinggi
Durian	43	23	6	14	2.56	0	Rawan Sedang
Gaharu	58	25	7	26	2.66	0	Rawan Sedang
Glugur Darat II	67	48	2	17	2.63	0	Rawan Sedang
Perintis	69	33	6	30	2.42	0	Rawan Sedang
Pulo Brayon Bengkel	56	25	9	22	2.52	0	Rawan Sedang
Pulo Brayon Bengkel Baru	57	22	13	22	2.60	0	Rawan Sedang
Sidodadi	35	19	9	7	2.60	0	Rawan Sedang

Sidorame Barat I	47	26	8	13	2.64	0	Rawan Sedang
Sidorame Timur	44	21	9	14	2.52	0	Rawan Sedang
Pahlawan	34	17	2	15	2.06	2	Rawan Rendah
Pandau Hilir	33	20	5	8	2.06	2	Rawan Rendah
Sei Kera Hilir II	13	4	2	7	1.92	2	Rawan Rendah
Sei Kera Hulu	33	18	1	14	2.15	2	Rawan Rendah
Sidorame Barat II	32	9	2	21	2.34	2	Rawan Rendah

Berdasarkan hasil klasterisasi K-Means, Klaster 1(Rawan Tinggi) dengan 6 kelurahan (Gang Buntu, Glugur Darat I, Pulo Brayan Darat I, Pulo Brayan Darat II, Sei Kera Hilir I, Tegal Rejo). Kelurahan dalam klaster ini memiliki total kriminal berkisar antara 89 hingga 162 kasus dengan dominasi kasus Curanmor. Klaster 0 (Rawan Sedang) dengan 9 kelurahan (Durian, Gaharu, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Timur). Total kriminal berkisar antara 35 hingga 69 kasus. Klaster 2 (Rawan Rendah): 5 kelurahan (Pahlawan, Pandau Hilir, Sei Kera Hilir II, Sei Kera Hulu, Sidorame Barat II). Total kriminal berkisar antara 13 hingga 34 kasus, dengan rata-rata waktu kejadian lebih awal (pagi-siang).

Algoritma DBSCAN dengan parameter $\text{eps}=0.2$ dan $\text{min_samples}=2$ menghasilkan pola klasterisasi yang berbeda secara signifikan dari K-Means. DBSCAN hanya memasukkan titik-titik dengan kepadatan tinggi ke dalam klaster, sementara titik yang terisolir atau berbeda jauh dari kelompok terdekat dikategorikan sebagai noise. Tabel berikut menampilkan hasil lengkap klasterisasi DBSCAN.

Tabel 4.38 Hasil Klasterisasi DBSCAN

Kelurahan	Total	Curanmor	Curas	Curat	Avg Waktu	Label	Kategori
Durian	43	23	6	14	2.56	0	Rawan Sedang
Pulo Brayan Bengkel	56	25	9	22	2.52	0	Rawan Sedang
Pulo Brayan Bengkel Baru	57	22	13	22	2.60	0	Rawan Sedang
Sidodadi	35	19	9	7	2.60	0	Rawan Sedang
Sidorame Barat I	47	26	8	13	2.64	0	Rawan Sedang
Sidorame Timur	44	21	9	14	2.52	0	Rawan Sedang
Pahlawan	34	17	2	15	2.06	1	Rawan Rendah
Sei Kera Hulu	33	18	1	14	2.15	1	Rawan Rendah
Gaharu	58	25	7	26	2.66	-1	Noise/Outlier
Gang Buntu	89	34	25	30	2.63	-1	Noise/Outlier
Glugur Darat I	162	117	15	30	2.57	-1	Noise/Outlier
Glugur Darat II	67	48	2	17	2.63	-1	Noise/Outlier
Pandau Hilir	33	20	5	8	2.06	-1	Noise/Outlier
Perintis	69	33	6	30	2.42	-1	Noise/Outlier
Pulo Brayan Darat I	105	70	14	21	2.79	-1	Noise/Outlier
Pulo Brayan Darat II	107	56	18	33	2.62	-1	Noise/Outlier
Sei Kera Hilir I	117	57	8	52	2.35	-1	Noise/Outlier
Sei Kera Hilir II	13	4	2	7	1.92	-1	Noise/Outlier
Sidorame Barat II	32	9	2	21	2.34	-1	Noise/Outlier
Tegal Rejo	107	45	10	52	2.74	-1	Noise/Outlier

Hasil klasterisasi DBSCAN menunjukkan distribusi sebagai berikut, Kluster 0 (Rawan Sedang) dengan 6 kelurahan (Durian, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Timur). Kelurahan-kelurahan ini memiliki karakteristik kriminalitas yang saling berdekatan dalam ruang fitur. Kluster 1 (Rawan Rendah) dengan 2 kelurahan (Pahlawan dan Sei Kera Hulu). Kedua kelurahan ini memiliki profil kriminalitas yang serupa, yakni total kriminal

rendah dengan waktu kejadian rata-rata lebih pagi. Noise/Outlier (Label -1) dengan 12 kelurahan (60%). Kelurahan-kelurahan ini memiliki profil kriminalitas yang unik dan tidak cukup dekat dengan kelurahan lain untuk membentuk klaster berdasarkan parameter yang ditentukan. Sebagian besar noise adalah kelurahan dengan karakteristik ekstrem seperti Glugur Darat I (total 162 kasus) maupun yang memiliki kombinasi fitur tidak lazim.

Tingginya persentase noise (60%) pada DBSCAN mengindikasikan bahwa data kriminalitas di Kecamatan Medan Timur memiliki distribusi yang cukup tersebar dan heterogen. Banyak kelurahan memiliki profil kriminalitas yang unik sehingga sulit dikelompokkan berdasarkan kepadatan dengan parameter yang dipilih.

4.5.3. Evaluasi Kualitas Klaster Menggunakan Silhouette Index dan Davies-Bouldin Index

Untuk mengevaluasi kualitas hasil klasterisasi yang diperoleh dari algoritma K-Means dan DBSCAN, penelitian ini menggunakan dua metode evaluasi, yaitu Silhouette Index (SI) dan Davies-Bouldin Index (DBI). Silhouette Index merupakan metode evaluasi yang mengukur seberapa baik suatu data berada dalam klasternya sendiri dibandingkan dengan klaster lain. Nilai SI berada pada rentang -1 hingga 1. Nilai yang mendekati 1 menunjukkan bahwa data berada pada klaster yang tepat dan terpisah dengan baik dari klaster lain, sedangkan nilai mendekati 0 menunjukkan bahwa struktur klaster kurang jelas. Jika nilai negatif muncul, maka kemungkinan data berada pada klaster yang tidak tepat.

Davies-Bouldin Index (DBI) merupakan metrik evaluasi yang mengukur rasio antara sebaran intra-klaster dan jarak antar-klaster. DBI dihitung sebagai rata-

rata dari nilai maksimum perbandingan antara jumlah rata-rata jarak dalam kluster terhadap jarak antar centroid kluster. Berbeda dengan Silhouette Index, pada DBI semakin kecil nilainya semakin baik kualitas kluster yang terbentuk, karena nilai DBI yang rendah mengindikasikan kluster yang kompak dan terpisah dengan baik. Nilai DBI = 0 berarti kluster sempurna tanpa tumpang tindih.

Tabel 4.39 Interpretasi sillhouutte index

Rentang Nilai	Interpretasi
0.71 – 1.00	Struktur kluster sangat kuat
0.51 – 0.70	Struktur kluster sedang
0.26 – 0.50	Struktur kluster lemah
0.00 – 0.25	Tidak ada struktur kluster yang jelas
< 0	Data kemungkinan berada pada kluster yang salah

1. Evaluasi K-means

Pada penelitian ini evaluasi menggunakan algoritma K-Means dilakukan terhadap seluruh 20 kelurahan yang terdapat pada dataset. Perhitungan dilakukan dengan menghitung nilai yang berdasarkan hasil klasterisasi K-Means dengan jumlah kluster $K=3$.

a = rata-rata jarak kelurahan tersebut ke semua anggota kluster sendiri maka semakin kecil semakin baik (semakin kompak)

b = rata-rata jarak ke kluster terdekat lainnya, semakin besar semakin baik (semakin terpisah)

s = Silhouette Score = $(b - a) / \max(a, b)$, nilai berkisar -1 hingga 1

Berikut output dari program python hitungan manual SI pada algoritma K-Means:

Tabel 4.40 Nilai SI K-Means

Kelurahan	Kluster (K)	a(i)	b(i)	s(i)
Durian	0	0.2698	0.5787	0.5337
Gaharu	0	0.3189	0.7634	0.5823
Gang Buntu	2	0.7529	0.8343	0.0976

Glugur Darat I	2	0.8209	1.1741	0.3008
Glugur Darat II	0	0.4176	0.7405	0.4360
Pahlawan	1	0.2441	0.6927	0.6475
Pandau Hilir	1	0.2963	0.6929	0.5723
Perintis	0	0.4264	0.6474	0.3413
Pulo Brayan Bengkel	0	0.2664	0.6377	0.5822
Pulo Brayan Bengkel Baru	0	0.3435	0.7570	0.5462
Pulo Brayan Darat I	2	0.6649	0.6748	0.0147
Pulo Brayan Darat II	2	0.5355	0.7314	0.2679
Sei Kera Hilir I	2	0.7754	0.9531	0.1864
Sei Kera Hilir II	1	0.3830	0.9148	0.5814
Sei Kera Hulu	1	0.2495	0.6268	0.6020
Sidodadi	0	0.3684	0.6744	0.4537
Sidorame Barat I	0	0.2703	0.6951	0.6111
Sidorame Barat II	1	0.4252	0.4712	0.0975
Sidorame Timur	0	0.2686	0.5919	0.5461
Tegal Rejo	2	0.7057	0.8930	0.2097
Rata-rata Silhouette Index (SI):				0.4105

Dari Tabel diatas terlihat bahwa K-Means memperoleh Silhouette Index rata-rata sebesar 0.4105. Terdapat 3 kelurahan dengan nilai $s(i)$ di bawah 0.25, yaitu Pulo Brayan Darat I ($s=0.0147$), Gang Buntu ($s=0.0976$), dan Sidorame Barat II ($s=0.0975$). Nilai $s(i)$ yang sangat rendah ini mengindikasikan bahwa ketiga kelurahan tersebut berada di perbatasan antara dua klaster, sehingga pengelompokannya kurang tegas. Nilai $SI = 0.4105$ termasuk dalam kategori "Struktur Lemah" berdasarkan interpretasi Kaufman dan Rousseeuw ($SI < 0.5$).

2. Evaluasi DBSCAN

Perhitungan Silhouette Index untuk DBSCAN hanya dilakukan pada 8 kelurahan yang berhasil dimasukkan ke dalam klaster (non-noise). Kelurahan yang dikategorikan sebagai noise (label -1) tidak diikutsertakan dalam perhitungan SI. Tabel 4.5 menyajikan nilai silhouette untuk kelurahan non-noise DBSCAN.

Tabel 4.41 Nilai SI DBSCAN

Kelurahan	Klaster (K)	a(i)	b(i)	s(i)
Durian	0	0.2153	0.5612	0.6164
Pahlawan	1	0.1155	0.6821	0.8307
Pulo Brayon Bengkel	0	0.2524	0.6170	0.5910
Pulo Brayon Bengkel Baru	0	0.3051	0.7787	0.6082
Sei Kera Hulu	1	0.1155	0.6154	0.8123
Sidodadi	0	0.2726	0.6722	0.5945
Sidorame Barat I	0	0.2033	0.6831	0.7024
Sidorame Timur	0	0.1887	0.5803	0.6749
Rata-rata Silhouette Index (SI):				0.5173

DBSCAN menghasilkan Silhouette Index rata-rata sebesar 0.5173 pada kelurahan non-noise (14 kelurahan terklaster). Nilai SI = 0.5173 masuk dalam kategori “Struktur Sedang” ($0.5 \leq SI < 0.7$), yang menunjukkan bahwa klaster yang terbentuk memiliki kualitas pemisahan yang cukup baik. Nilai Davies-Bouldin Index DBSCAN sebesar 0.5976 yang lebih rendah dibandingkan K-Means (0.835) mengonfirmasi bahwa klaster DBSCAN lebih kompak dan lebih terpisah satu sama lain. Konsistensi nilai s(i) yang tinggi pada kelurahan-kelurahan seperti Pahlawan ($s=0.8307$) dan Sei Kera Hulu ($s=0.8123$) mengindikasikan bahwa DBSCAN berhasil mengidentifikasi kelurahan-kelurahan yang memang memiliki karakteristik kriminalitas serupa.

4.5.4. Perbandingan hasil evaluasi klaster

Berikut Tabel perbandingan hasil evaluasi klasterisasi yang merangkum perbandingan komprehensif antara K-Means dan DBSCAN berdasarkan berbagai kriteria evaluasi yang relevan.

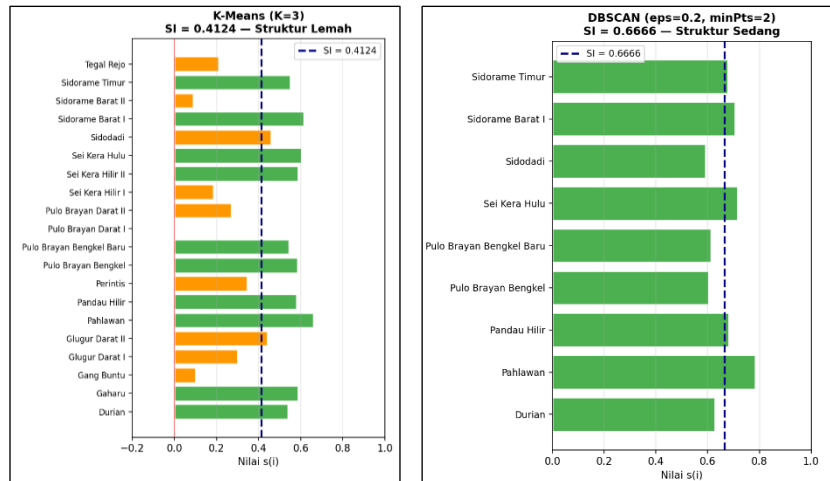
Tabel 4.42 Perbandingan Hasil Evaluasi

Kriteria Evaluasi	K-Means	DBSCAN
Jumlah Klaster Terbentuk	3 klaster	2 klaster + noise

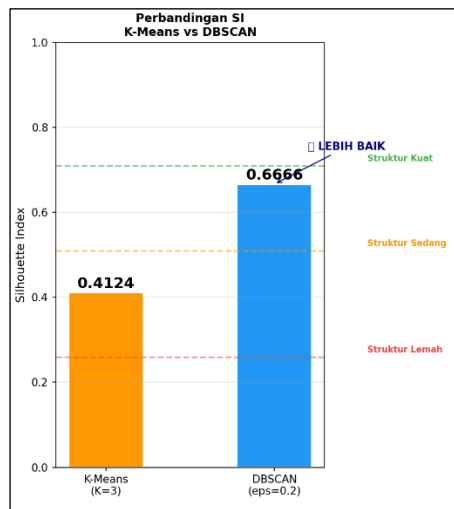
Jumlah Data Terklaster	20 kelurahan (100%)	8 kelurahan (40%)
Jumlah Noise/Outlier	0 (tidak ada)	12 kelurahan (60%)
Silhouette Index (SI)	0.4105	0.5173
Kategori Kualitas Klaster	Struktur Lemah ($SI < 0.5$)	Struktur Sedang ($0.5 \leq SI < 0.7$)
Nilai $s(i)$ Tertinggi	0.6475 (Pahlawan)	0.8307 (Pahlawan)
Nilai $s(i)$ Terendah	0.0147 (Pulo Brayan Darat I)	0.5910 (Pulo Brayan Bengkel)
Kelurahan $s(i) < 0.25$	3 kelurahan (Gang Buntu, Pulo Brayan Darat I, Sidorame Barat II)	0 kelurahan
Kelurahan $s(i) \geq 0.5$	12 dari 20 kelurahan (60%)	8 dari 8 kelurahan (100%)
Davies-Bouldin Index (DBI)	0.835	0.5976
Selisih SI	DBSCAN lebih unggul: selisih 0.1068 (DBSCAN - K-Means)	

Berdasarkan tabel diatas, tabel menunjukkan bahwa DBSCAN unggul secara signifikan dalam hal kualitas klaster baik dari sisi Silhouette Index maupun Davies-Bouldin Index. DBSCAN menghasilkan SI sebesar 0.5173 (struktur sedang) dibandingkan K-Means dengan SI 0.4105 (struktur lemah), dan DBI DBSCAN sebesar 0.5976 lebih rendah dari DBI K-Means sebesar 0.835, yang mengindikasikan bahwa klaster DBSCAN lebih kompak dan lebih terpisah. Namun DBSCAN memiliki kelemahan dalam cakupan data karena hanya mampu mengklasterkan 40% dari total kelurahan dan mengategorikan 60% sebagai noise. Sementara itu, K-Means berhasil mengklasterkan 100% data tetapi menghasilkan beberapa pengelompokan yang kurang tepat (nilai $s(i)$ rendah). Selisih DBI antara kedua algoritma sebesar 0.2374 menunjukkan perbedaan kualitas klaster yang cukup substansial, di mana semakin rendah DBI semakin baik pemisahan klaster yang terbentuk.

Grafik perbandingan Silhouette Index dan Davies-Bouldin Index menunjukkan perbedaan kualitas kluster yang dihasilkan oleh algoritma K-Means dan DBSCAN. Berikut hasil visualisasi algoritma berdasarkan evaluasi metrik Silhouette Index dan Davies-Bouldin Index dari implementasi website penelitian.



Gambar 4.35 Visualisasi SI K-Means dan DBSCAN



Gambar 4.36 Perbandingan SI

Pada algoritma K-Means dengan nilai K = 3, diperoleh nilai Silhouette Index sebesar 0.4105 yang termasuk dalam kategori struktur kluster lemah, sehingga menunjukkan bahwa pemisahan antar kluster belum terlalu optimal. Sementara itu, algoritma DBSCAN dengan parameter eps = 0.2 dan minPts = 5

menghasilkan nilai Silhouette Index sebesar 0.5173 yang termasuk dalam kategori struktur kluster sedang. Nilai tersebut lebih tinggi dibandingkan K-Means, sehingga menunjukkan bahwa DBSCAN mampu menghasilkan kluster yang lebih kompak dan memiliki pemisahan yang lebih baik antar kluster pada data kriminalitas yang dianalisis. Selain itu, evaluasi menggunakan Davies-Bouldin Index (DBI) menunjukkan nilai K-Means sebesar 0.835 dan DBSCAN sebesar 0.5976, di mana nilai DBI yang lebih rendah pada DBSCAN mengonfirmasi kualitas kluster yang lebih baik. Dengan demikian, berdasarkan evaluasi menggunakan Silhouette Index dan Davies-Bouldin Index, algoritma DBSCAN memberikan hasil klusterisasi yang lebih baik dibandingkan K-Means.

4.5.5. Kesimpulan Perbandingan Algoritma

Berdasarkan analisis komprehensif yang telah dilakukan, dapat disimpulkan DBSCAN lebih unggul dari sisi kualitas kluster dengan Silhouette Index 0.5173 (kategori Struktur Sedang) dan Davies-Bouldin Index 0.5976, dibandingkan K-Means dengan SI 0.4105 (kategori Struktur Lemah) dan DBI 0.835. Nilai DBI yang lebih rendah pada DBSCAN mengonfirmasi bahwa kluster yang dihasilkan DBSCAN memiliki koheisi internal yang lebih baik dan separasi antar-kluster yang jauh lebih tegas.

K-Means lebih unggul dari sisi cakupan karena mampu mengklusterkan 100% data (20 kelurahan), sedangkan DBSCAN hanya mengklusterkan 40% data (8 kelurahan) dan mengategorikan 60% sisanya sebagai noise. Untuk tujuan penelitian ini, DBSCAN dipilih sebagai algoritma yang lebih baik berdasarkan metrik evaluasi Silhouette Index. Namun, untuk keperluan kebijakan yang

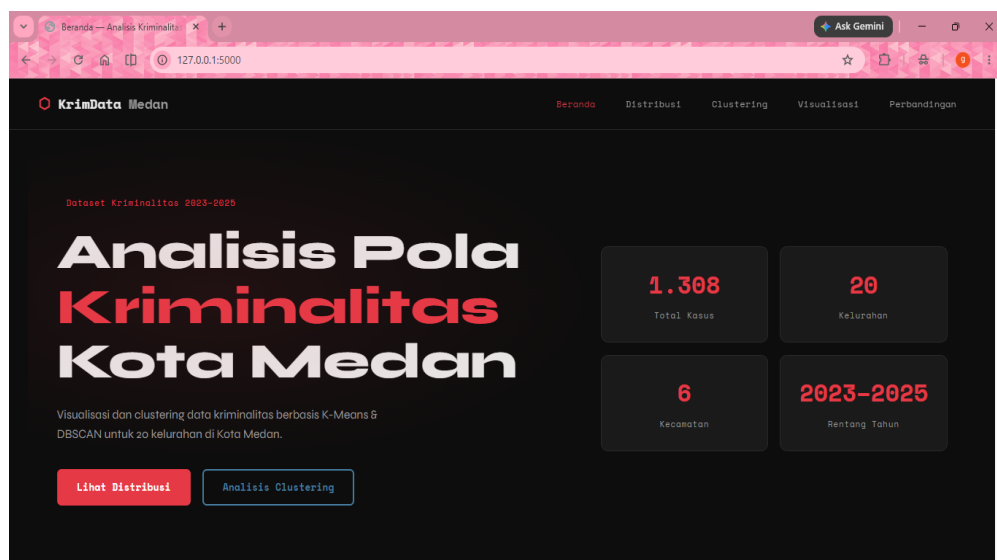
membutuhkan kategorisasi lengkap seluruh wilayah, K-Means tetap menjadi pilihan yang lebih praktis karena tidak menghasilkan noise.

Kelurahan Pahlawan dan Sei Kera Hulu memperoleh nilai silhouette tertinggi dalam DBSCAN (masing-masing 0.8307 dan 0.8123), mengindikasikan bahwa kedua kelurahan ini memiliki profil kriminalitas yang paling homogen dan terdefinisi dengan baik dibandingkan kelurahan lainnya.

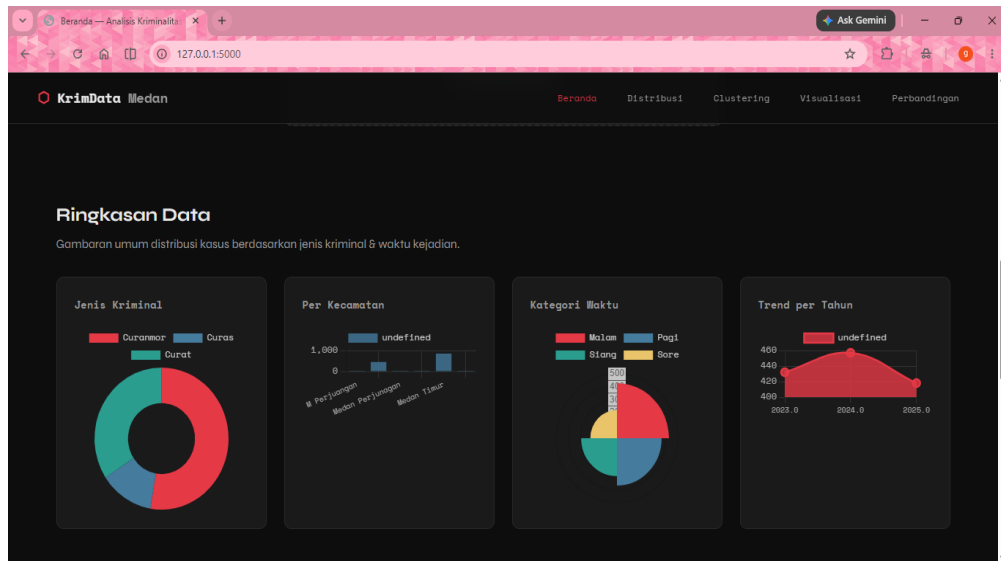
4.5.6. Hasil Implementasi Website

Penelitian ini juga mengembangkan sistem visualisasi berbasis website bernama KrimData Medan yang dapat diakses secara lokal pada alamat 127.0.0.1:1000. Website ini menyajikan hasil analisis klusterisasi kriminalitas secara interaktif dan memudahkan interpretasi hasil penelitian. Website terdiri atas beberapa halaman utama, yaitu Beranda, Distribusi, Clustering, Visualisasi, dan Perbandingan.

1. Beranda



Gambar 4.37 Tampilan Halaman Beranda

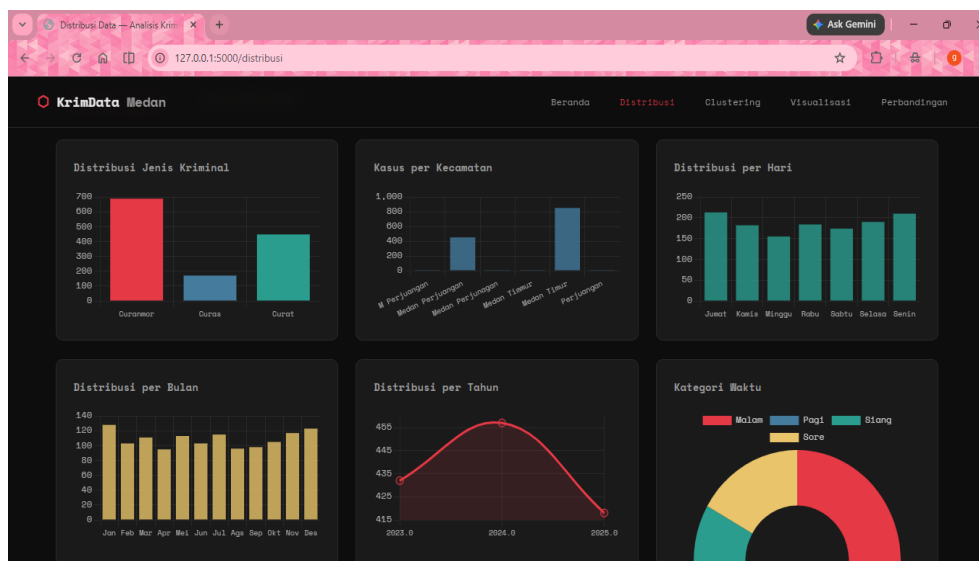


Gambar 4.38 Tampilan Ringkasan Data pada Halaman Beranda

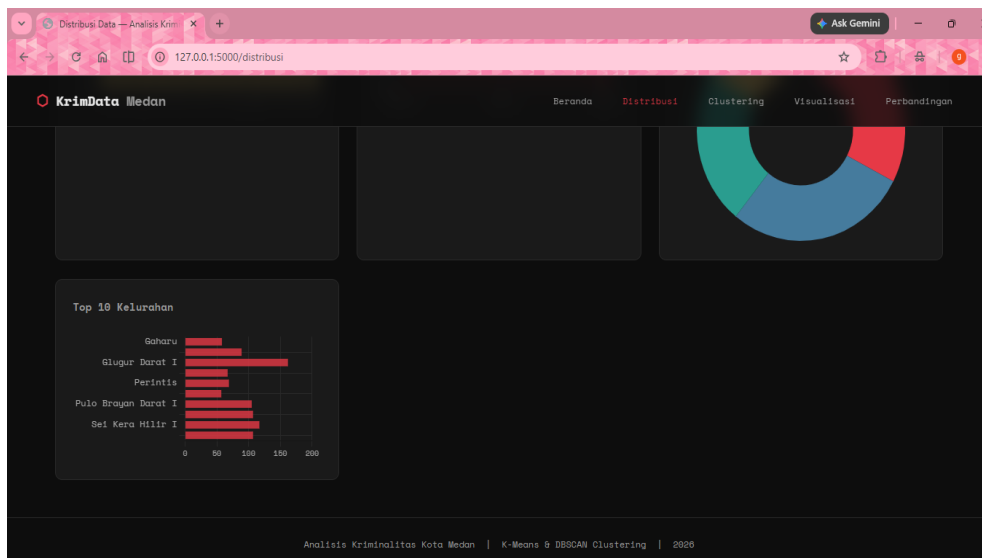
Halaman Beranda menampilkan ringkasan dataset kriminalitas 2023–2025 dengan statistik utama: 1.308 total kasus, 20 kelurahan, 6 kecamatan, dan rentang tahun 2023–2025. Halaman tersebut ditampilkan dengan memasukkan file excel ke dalam website.

2. Distribusi

Halaman Distribusi menyajikan visualisasi distribusi data meliputi distribusi jenis kriminal (curanmor, curas, curat), kasus per kecamatan, distribusi per hari, distribusi per bulan, distribusi per tahun, serta kategori waktu kejadian (malam, pagi, siang, sore). Berdasarkan data distribusi, kelurahan dengan jumlah kriminal tertinggi adalah Gaharu, Glugur Darat I, Perintis, Pulo Brayan Darat I, dan Sei Kera Hilir I.



Gambar 4.39 Tampilan Halaman Distribusi

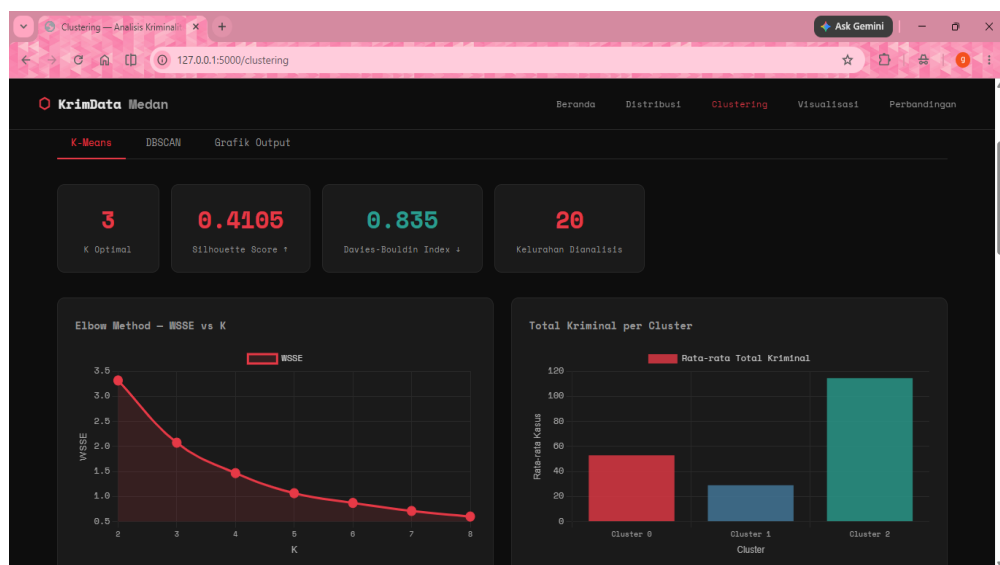


Gambar 4.40 Tampilan Halaman Distribusi Lanjutan

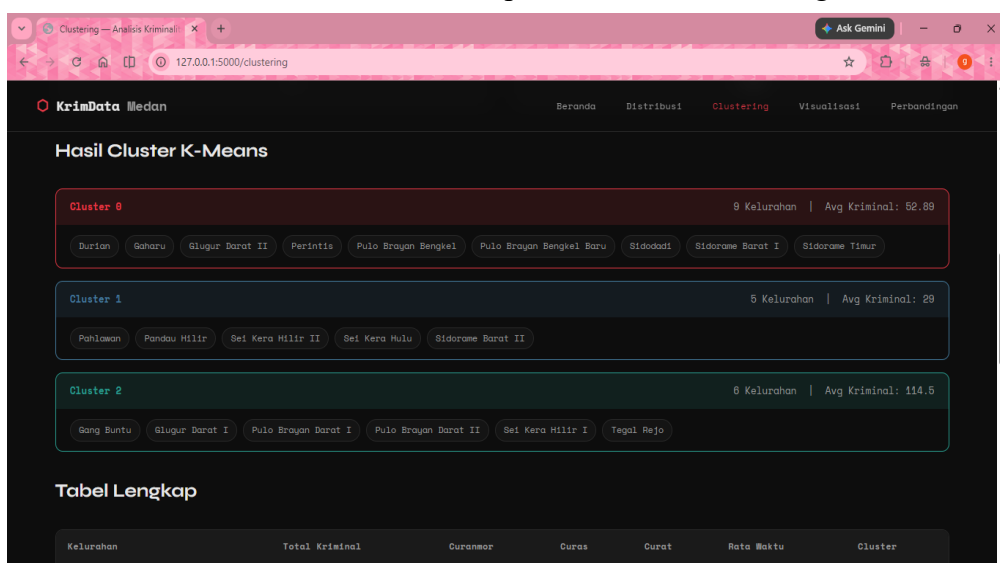
3. Clustering

Halaman Clustering menyajikan hasil pengelompokan untuk kedua algoritma secara terpisah. Pada tab K-Means, website menampilkan nilai K optimal = 3, Silhouette Score = 0.4105, Davies-Bouldin Index = 0.835, dan jumlah kelurahan teranalisis = 20. Grafik Elbow Method (BSSE vs K) ditampilkan untuk memvalidasi pemilihan $K=3$. Hasil cluster K-Means dari website: Cluster 0 (Rawan Sedang) terdiri dari 9 kelurahan (Durian, Gaharu, Glugur

Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Timur) dengan rata-rata kriminal 90.33; Cluster 1 (Rawan Rendah) terdiri dari 5 kelurahan (Pahlawan, Pandau Hilir, Sei Kera Hilir II, Sei Kera Hulu, Sidorame Barat II) dengan rata-rata kriminal 20; Cluster 2 (Sangat Rawan) terdiri dari 6 kelurahan (Gang Buntu, Glugur Darat I, Pulo Brayan Darat I, Pulo Brayan Darat II, Sei Kera Hilir I, Tegal Rejo) dengan rata-rata kriminal 114.5.



Gambar 4.41 Hasil K-Means pada Halaman Clustering

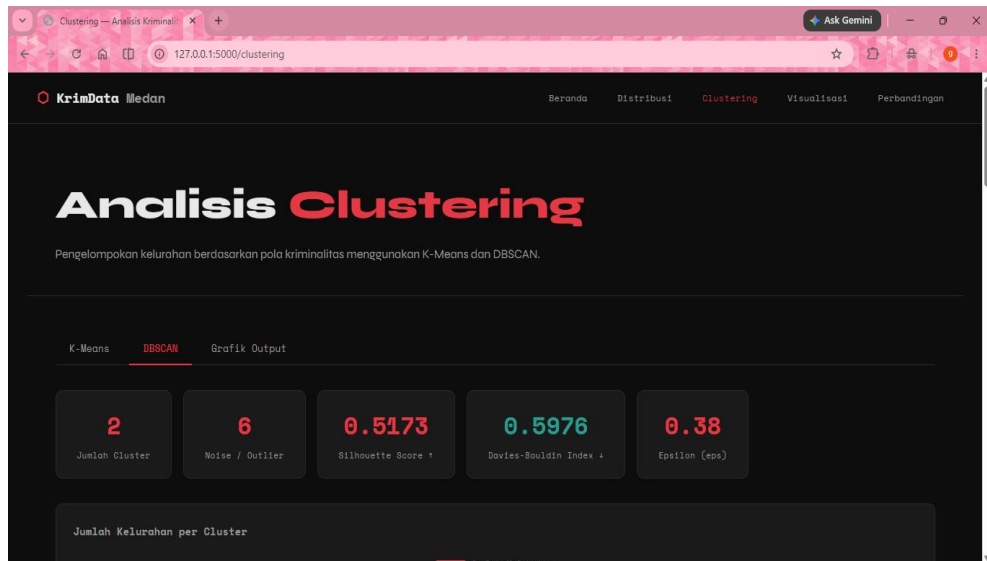


Gambar 4.42 Hasil K-Means pada Halaman Clustering lanjutan

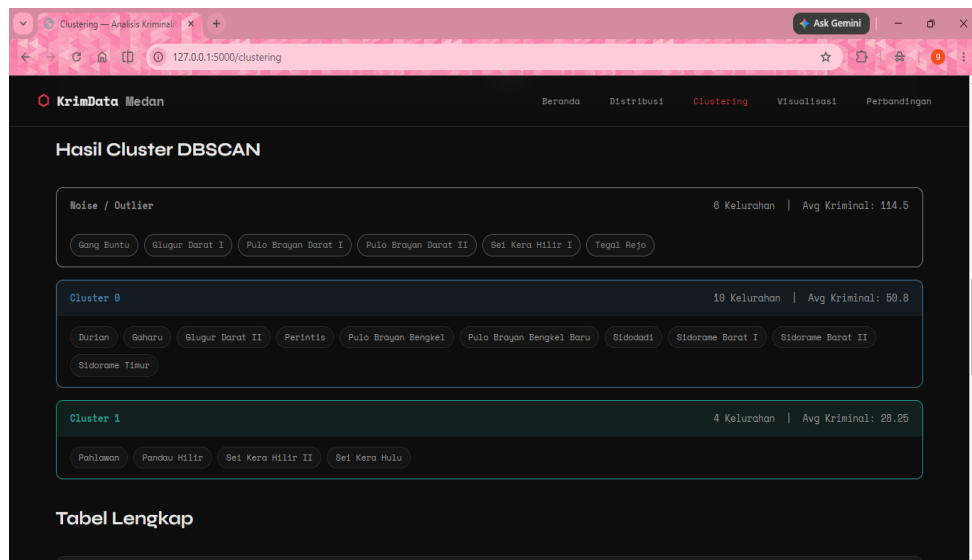
Kelurahan	Total Kriminal	Curanmor	Curas	Curat	Rata Waktu	Cluster
Durian	43	23	6	14	2.56	0
Gaharu	68	26	7	26	2.66	0
Gang Buntu	89	34	28	30	2.63	2
Glugur Darat I	162	117	15	30	2.57	2
Glugur Darat II	67	48	2	17	2.63	0
Pahlawan	34	17	2	15	2.06	1
Pandau Hilir	33	20	5	8	2.06	1
Perintis	69	33	6	30	2.42	0
Pulo Brayan Bengkel	66	26	9	22	2.62	0

Gambar 4.43 Hasil K-Means pada Halaman Clustering lanjutan

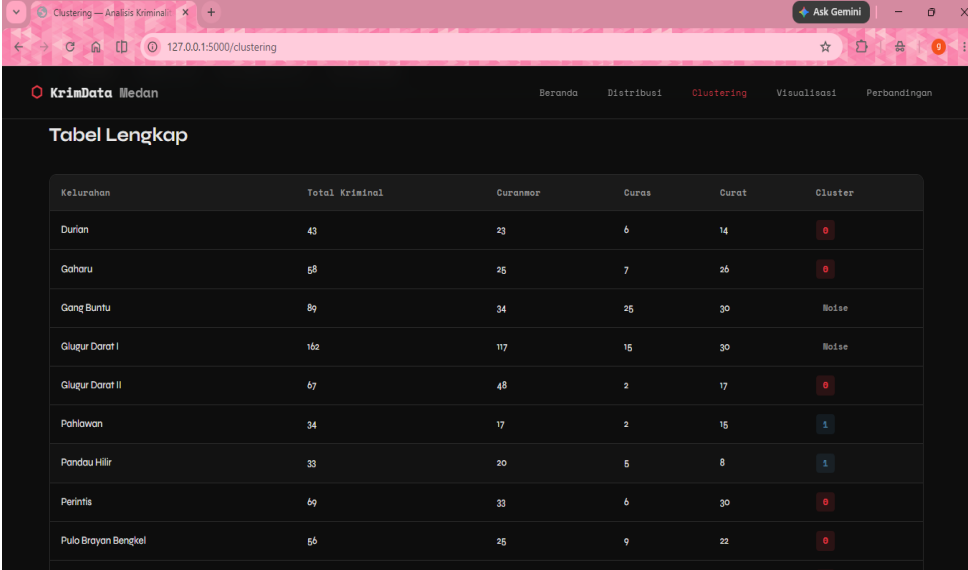
Pada tab DBSCAN, website menampilkan hasil: Jumlah Cluster = 2, Noise/Outlier = 6 kelurahan, Silhouette Score = 0.5173, Davies-Bouldin Index = 0.5976, dan parameter eps = 0.2, minPts = 5. Hasil cluster DBSCAN dari website: Noise/Outlier (6 kelurahan): Gang Buntu, Glugur Darat I, Pulo Brayan Darat I, Pulo Brayan Darat II, Sei Kera Hilir I, dan Tegal Rejo, dengan rata-rata kriminal 114.5; Cluster 0 (Rawan Sedang) terdiri dari 10 kelurahan (Durian, Gaharu, Glugur Darat II, Perintis, Pulo Brayan Bengkel, Pulo Brayan Bengkel Baru, Sidodadi, Sidorame Barat I, Sidorame Barat II, Sidorame Timur) dengan rata-rata kriminal 59.8; Cluster 1 (Rawan Rendah) terdiri dari 4 kelurahan (Pahlawan, Pandau Hilir, Sei Kera Hilir II, Sei Kera Hulu) dengan rata-rata kriminal 26.25.



Gambar 4.44 Hasil DBSCAN pada Halaman Clustering



Gambar 4.45 Hasil DBSCAN pada Halaman Clustering lanjutan



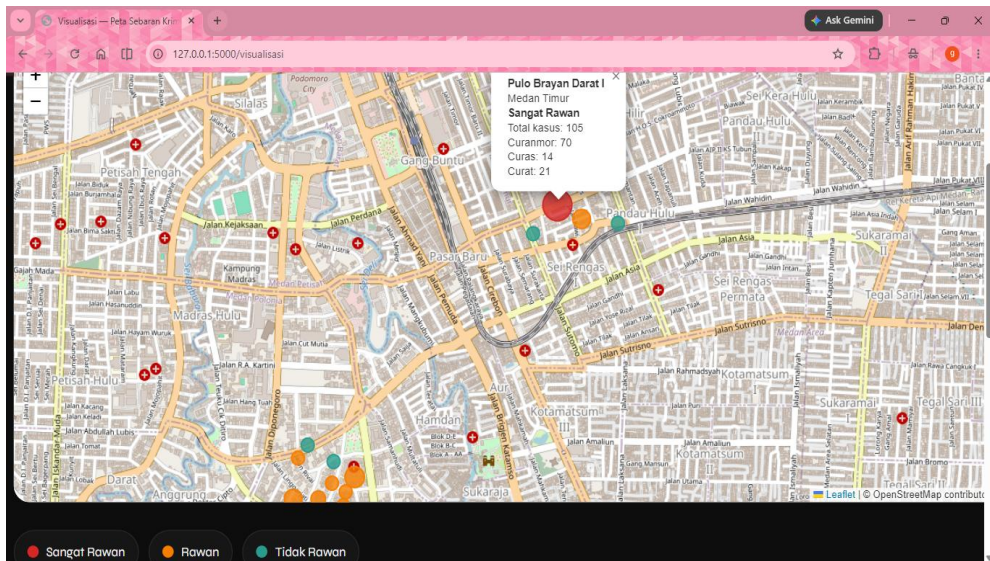
The screenshot shows a web interface for 'KrimData Medan' with a 'Clustering' tab selected. Below the navigation bar, there is a section titled 'Tabel Lengkap' containing a table with the following data:

Kelurahan	Total Kriminal	Curanmor	Curas	Curat	Cluster
Durian	43	23	6	14	0
Gaharu	58	28	7	26	0
Gang Buntu	89	34	28	30	Noise
Glugur Darat I	162	117	15	30	Noise
Glugur Darat II	67	48	2	17	0
Pahlawan	34	17	2	15	1
Pandau Hillir	33	20	5	8	1
Perintis	69	33	6	30	0
Pulo Brayan Bengkel	56	28	9	22	0

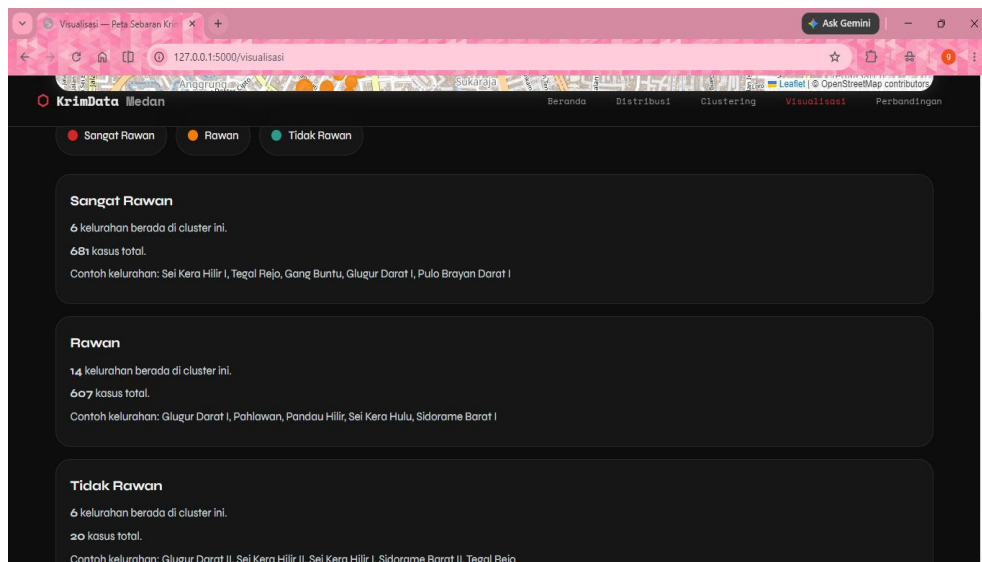
Gambar 4.46 Hasil DBSCAN pada Halaman Clustering lanjutan

4. Visualisasi

Halaman Visualisasi menampilkan peta interaktif berbasis Leaflet/OpenStreetMap yang memvisualisasikan kelurahan-kelurahan di Kecamatan Medan Timur beserta kategori kerawanannya: Sangat Rawan (6 kelurahan, 681 kasus total), Rawan (14 kelurahan, 607 kasus total), dan Tidak Rawan (6 kelurahan, 20 kasus total). Peta juga menampilkan detail per kelurahan saat diklik, seperti contoh Pulo Brayan Darat I yang termasuk kategori Sangat Rawan dengan 105 total kasus, Curanmor: 70, Curas: 14, Curat: 21.



Gambar 4.47 Hasil visualisasi

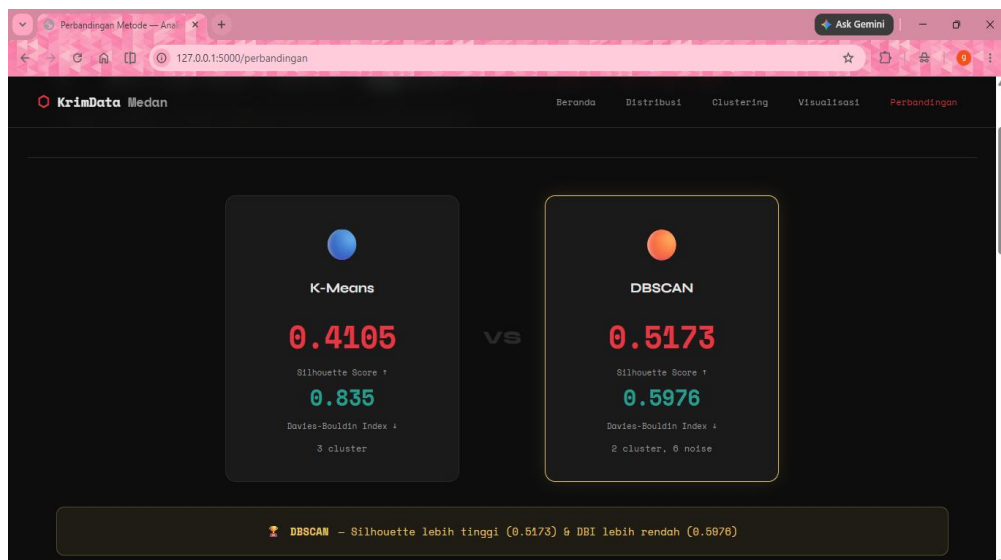


Gambar 4.48 Hasil visualisasi lanjutan

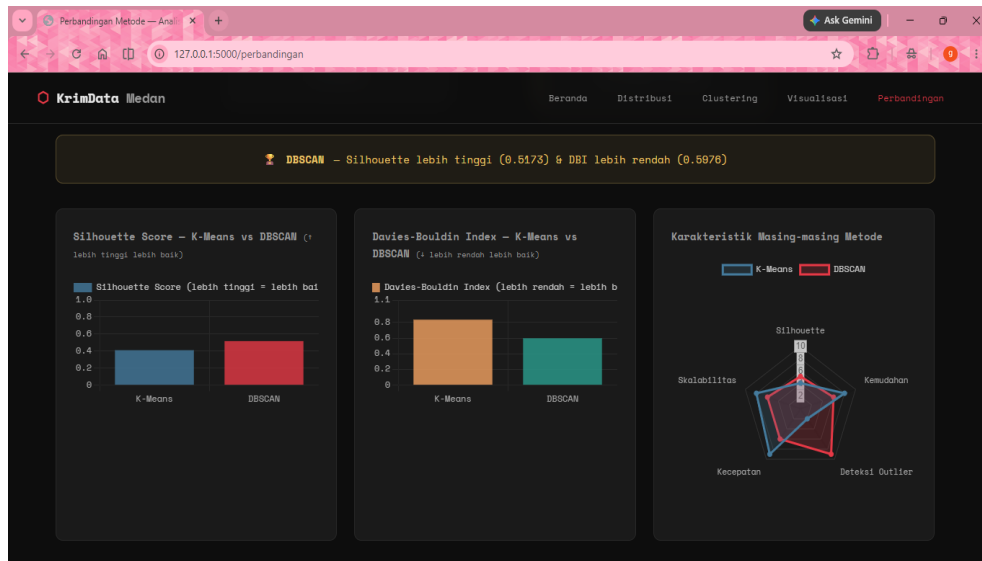
5. Perbandingan

Halaman Perbandingan menampilkan perbandingan komprehensif antara K-Means dan DBSCAN melalui tiga visualisasi: (1) grafik batang Silhouette Score K-Means vs DBSCAN, (2) grafik batang Davies-Bouldin Index K-Means vs DBSCAN, dan (3) radar chart perbandingan karakteristik metode meliputi dimensi Silhouette, Fleksibilitas, Kecepatan, Sensitivitas, dan Deteksi Outlier. Kesimpulan dari halaman perbandingan: “DBSCAN – Silhouette lebih

tinggi (0.5173) & DBI lebih rendah (0.5976).” Tabel perbandingan pada website menunjukkan: K-Means memerlukan penentuan K (Ya, Elbow Method), tidak mendeteksi outlier (Tidak), bentuk kluster bulat/convex, sensitif terhadap noise (Ya), dengan Silhouette Score = 0.4105 dan DBI = 0.835, jumlah kluster 3; sedangkan DBSCAN tidak memerlukan penentuan K (Tidak), mendeteksi outlier berupa noise point (Ya), bentuk kluster bebas (arbitrary shapes), tidak sensitif terhadap noise (Tidak), dengan Silhouette Score = 0.5173 dan DBI = 0.5976, jumlah kluster 2 dengan 6 kelurahan sebagai noise.



Gambar 4.49 Tampilan Halaman Perbandingan



Gambar 4.50 Tampilan Halaman Perbandingan lanjutan

Catatan Metode: Silhouette Score DBSCAN dihitung hanya dari titik non-noise (mengecualikan outlier), sesuai dengan metode yang digunakan dalam penelitian. Parameter DBSCAN dipilih secara otomatis untuk meningkatkan kualitas cluster sambil menjaga noise tetap rendah.

Kriteria	K-Means	DBSCAN
Perlu tentukan K	Ya (Elbow Method)	Tidak
Deteksi Outlier	Tidak	Ya (noise point)
Bentuk Cluster	Bulat / Convex	Bebas (arbitrary shape)
Sensitif terhadap noise	Ya	Tidak
Parameter	K3 (Elbow optimal)	eps=0.38, minPts=5
Silhouette Score ↑ (lebih tinggi lebih baik)	0.4106	0.5173
Davies-Bouldin Index ↓ (lebih rendah lebih baik)	0.835	0.5976
Jumlah Cluster	3	2
Noise / Outlier	-	6 kelurahan

Gambar 4.51 Tampilan Halaman Perbandingan lanjutan

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai klasterisasi daerah rawan kriminalitas di Medan Timur menggunakan algoritma K-Means dan DBSCAN, dapat disimpulkan bahwa proses klasterisasi berhasil dilakukan melalui tahapan preprocessing data, feature engineering, normalisasi, serta implementasi kedua algoritma tersebut.

Hasil penelitian menunjukkan bahwa algoritma K-Means dengan jumlah kluster ($K=3$) mampu mengelompokkan seluruh 20 kelurahan di Medan Timur, namun menghasilkan nilai Silhouette Index sebesar 0,4105 yang termasuk dalam kategori lemah dan Davies-Bouldin Index sebesar 0,835. Sementara itu, algoritma DBSCAN dengan parameter $\text{eps} = 0,20$ dan $\text{minPts} = 5$ menghasilkan dua kluster dari 14 kelurahan dengan nilai Silhouette Index sebesar 0,5173 yang berada pada kategori sedang dan Davies-Bouldin Index sebesar 0,5976. Hal ini menunjukkan bahwa DBSCAN memiliki kualitas kluster yang lebih baik dibandingkan K-Means dari kedua metrik evaluasi, terutama dalam mendeteksi pola kepadatan data dan mengidentifikasi outlier, meskipun K-Means lebih unggul dalam mengelompokkan seluruh data secara menyeluruh.

Dengan demikian, hasil klasterisasi yang diperoleh dapat dimanfaatkan sebagai dasar rekomendasi bagi pihak kepolisian dalam menentukan wilayah prioritas pengawasan serta merancang strategi pencegahan kriminalitas yang lebih

efektif dan efisien di wilayah Medan Timur, serta dapat meningkatkan kewaspadaan masyarakat khususnya medan timur terhadap tindak kriminal.

5.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, disarankan agar penelitian selanjutnya dapat menggunakan dataset yang lebih besar dan lebih lengkap dengan menambahkan variabel lain seperti waktu kejadian, faktor sosial, dan kepadatan penduduk agar hasil analisis menjadi lebih akurat dan komprehensif.

Selain itu, penelitian selanjutnya juga dapat menguji algoritma clustering lainnya seperti Hierarchical Clustering atau OPTICS untuk memperoleh perbandingan metode yang lebih luas dan mendalam. Penentuan parameter pada algoritma DBSCAN seperti eps dan minPts juga perlu dioptimalkan menggunakan metode tertentu agar menghasilkan klaster yang lebih maksimal. Selanjutnya, pengembangan sistem visualisasi berbasis peta atau Geographic Information System (GIS) sangat disarankan agar hasil klasterisasi dapat ditampilkan secara lebih interaktif dan mudah dipahami. Terakhir, penelitian berikutnya dapat mengkombinasikan metode clustering dengan teknik prediksi sehingga tidak hanya mampu mengelompokkan data, tetapi juga dapat memperkirakan potensi kriminalitas di masa yang akan datang.

DAFTAR PUSTAKA

- Anfelia, Y., Ula, M., & Retno, S. (2024). *Identification Of Environmental Security In Relation To The Crime Case Rate In Simeulue Regency Using Density Based Spatial Clustering Method With Noise (DBSCAN)*. 2, 2–12.
- Aswan, Y., Defit, S., & Nurcahyo, G. W. (2021). *Jurnal Sistim Informasi dan Teknologi Algoritma K-Means Clustering dalam Mengklasifikasi Data Daerah Rawan Tindak Kriminalitas (Polres Kepulauan Mentawai)*. 3, 245–250. <https://doi.org/10.37034/jsisfotek.v3i4.73>
- Azhari, D. A., Maulita, Y., & Ramadani, S. (2024). *Pengelompokan Data Kriminal untuk Menentukan Pola Rawan Tindak Kriminal Menggunakan Algoritma K-Means (Studi Kasus : Polsek Hampan Perak) judul penelitian yaitu “ Pengelompokan Data Kriminal Untuk Menentukan Pola Rawan Tindak Kriminal Menggunakan Algoritma K- Means ”. Faktor lingkungan Pergaulan Pada dasarnya manusia tidak dapat hidup tanpa kehadiran dan bantuan orang lain di*. 2(5).
- Dalam, P. K. D. A. N. K. (2025). *Dalam pengelompokkan tingkat kejahatan pada provinsi jawa tengah*. 8, 295–303.
- Elda, Y., Defit, S., Yunus, Y., & Syaljumairi, R. (2021). *Klasterisasi Penempatan Siswa yang Optimal untuk Meningkatkan Nilai Rata-Rata Kelas Menggunakan K-Means*. *Jurnal Informasi Dan Teknologi*, 3, 103–108. <https://doi.org/10.37034/jidt.v3i3.130>
- Foundations, T. (2021). *Clustering*. 129–130.
- Furqan, M., Aulia, A., & Sriani. (2022). *Algoritma K-Means Untuk Segmentasi*

- Kematangan Buah Jeruk Berdasarkan Kemiripan Warna. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 6(1), 199–208.
[http://repository.uinsu.ac.id/id/eprint/13134%0Ahttp://repository.uinsu.ac.id/13134/1/SKRIPSI ATIQUAH AULIA-1.pdf](http://repository.uinsu.ac.id/id/eprint/13134%0Ahttp://repository.uinsu.ac.id/13134/1/SKRIPSI%20ATIQUAH%20AULIA-1.pdf)
- Herlina, & Ernaningsih, Z. (2023). Implementasi K-Means Clustering untuk Analisis Tingkat Pemahaman Computational Thinking Siswa Sekolah Dasar. *Jurnal Media Informatika Budidarma*, 7(3), 1405–1413.
<https://doi.org/10.30865/mib.v7i3.6132>
- Kartikasari, M. D. (2021). Self-Organizing Map Menggunakan Davies-Bouldin Index dalam Pengelompokan Wilayah Indonesia Berdasarkan Konsumsi Pangan. *Jambura Journal of Mathematics*, 3(2), 187–196.
<https://doi.org/https://doi.org/10.34312/jjom.v3i2.10942>
- Kurniawan, R. A., Hasibuan, M. S., Piramida, P., & Ramadhan, R. S. (2022). Penerapan Algoritma K-Means Untuk Clustering Tempat Makan Di Batubara. *Journal of Computer Science and Informatics Engineering (CoSIE)*, 01(1), 10–18. <https://doi.org/10.55537/cosie.v1i1.27>
- Mochurad, L., Sydor, A., & Ratinskiy, O. (2023). *A fast parallelized DBSCAN algorithm based on OpenMp for detection of criminals on streaming services.*
- Rahmadayanti, F., Rahayu, R., Informatika, P. S., & Alam, P. (2023). *PENERAPAN METODE DATA MINING PADA KASUS KRIMINALITAS.* 15(1), 52–61.
- Ramadani, A. D., Ibrahim, F. H., Hidayat, M., & Habibullah, A. (2025). *Klasterisasi Data Produksi Daging Sapi Menggunakan Algoritma K-Means Orange Data Mining.* 5(1), 109–114.
- Rosiana, P. S., Mohsa, A. A., Fadila, M. A., & Haerul, J. (2023). *DATA*

KARAWANG DENGAN MENGGUNAKAN ALGORITMA CLUSTERING K-MEANS. 11(3).

Suriani, L. (2020). *Pengelompokan Data Kriminal Pada Poldasu Menentukan Pola Daerah Rawan Tindak Kriminal Menggunakan Data Mining Algoritma K-Means Clustering. 1*, 151–157. <https://doi.org/10.30865/json.v1i2.1955>

Syawali, Y., Haikal, M., Rangkuti, H., & Mayadi, K. A. (n.d.). *PENERAPAN ALGORITMA DBSCAN UNTUK ANALISIS DEMOGRAFIS dan PENGELUARAN PELANGGAN MALL.*

Tambunan, A. M. A., Khaira, U., & Hutabarat, B. F. (2025). *Implementasi algoritma dbscan untuk klasterisasi koperasi berdasarkan indikator keuangan. 07(01)*, 38–55.

Technology, I. (2025). *Clustering Women Violence Cases Based on Number in Central Java Province Using K-Means Algorithm. 6(1).* <https://doi.org/10.30596/jcositte.v6i1.21671>

Thakur, R. P. (2020). *CRIME ANALYSIS AND PREDICTION - BY USING DBSCAN ALGORITHM. 4262–4266.*