**ANALISIS DETEKSI DINI PENYAKIT JANTUNG DENGAN PENDEKATAN REGRESI LOGISTIK PADA DATA PASIEN**

## **IKHLASHUL AMAL**

UNIVERSITAS MUHAMMADIYAH MAKASSAR

e-mail: xxxx@xxxx.xxx\*

***Abstract***

*This study aims to classify heart disease risk in patients using clustering and logistic regression methods. The data includes 640 patients from RSUD Haji Makassar, where each patient is classified into three risk categories: low, high, and very high. The Elbow method was used to determine the optimal number of clusters, and the analysis results indicate that three clusters are the most suitable for this data. Cluster 0 includes patients with low risk, Cluster 1 includes patients with high risk, and Cluster 2 includes patients with very high risk. Subsequently, logistic regression was applied to predict heart disease risk based on available health variables such as age, blood pressure, and cholesterol levels. The accuracy of the logistic regression method was 98.44%. The results of this study show that the combination of clustering and logistic regression methods is effective in grouping patients and predicting heart disease risk levels, which can assist in medical decision-making and more precise healthcare planning.*

*Keywords: Clustering, Logistic Regression, Elbow, K-Means, Heart Disease Risk*

**Abstrak**

Penelitian ini bertujuan untuk mengklasifikasikan risiko penyakit jantung pada pasien dengan menggunakan metode clustering dan regresi logistik. Data yang digunakan mencakup 640 pasien di RSUD Haji Makassar, di mana setiap pasien diklasifikasikan ke dalam tiga kategori risiko: rendah, tinggi, dan sangat tinggi. Metode Elbow digunakan untuk menentukan jumlah cluster optimal, dan hasil analisis menunjukkan bahwa tiga cluster merupakan jumlah yang paling sesuai untuk data ini. Cluster 0 mencakup pasien dengan risiko rendah, Cluster 1 mencakup pasien dengan risiko tinggi, dan Cluster 2 mencakup pasien dengan risiko sangat tinggi. Selanjutnya, regresi logistik diterapkan untuk memprediksi risiko penyakit jantung berdasarkan variabel kesehatan yang tersedia, seperti usia, tekanan darah, dan kadar kolesterol.Hasil akurasi dari metode regresi logistic tersebut sebesar 98.44%. Hasil penelitian ini menunjukkan bahwa kombinasi metode clustering dan regresi logistik efektif dalam mengelompokkan pasien dan memprediksi tingkat risiko penyakit jantung, yang dapat membantu dalam pengambilan keputusan medis dan perencanaan perawatan kesehatan yang lebih tepat.

Kata Kunci: Clustering, Regresi Logistik, Elbow, K-Means, Risiko Penyakit Jantung

**1. Pendahuluan**

Penyakit kronis adalah kondisi kesehatan yang berlangsung lama, biasanya lebih dari tiga bulan, dan sering memerlukan penanganan serta perawatan berkepanjangan. Penyakit ini dapat berkembang secara perlahan dan mempengaruhi kemampuan penderitanya untuk melakukan aktivitas sehari-hari. Penyakit kronis merupakan penyebab kematian utama di Indonesia. Menurut Kementerian Kesehatan Republik Indonesia (2016), proporsi kematian akibat penyakit tidak menular meningkat dari 41,7% pada tahun 1995 menjadi 49,9% pada tahun 2001 dan sebesar 59,5% pada tahun 2007. Penyebab kematian utama adalah stroke ( 15,4%), disusul hipertensi, diabetes, kanker, dan PPOK. Penyakit kronis biasanya berlangsung dalam jangka waktu yang lama, bisa berhari-hari, berbulan-bulan, bahkan bertahun-tahun (Sri et al., 2021).

Penyakit jantung adalah istilah umum yang mencakup sejumlah kondisi yang mempengaruhi jantung dan pembuluh darah. Penyakit ini seringkali disebabkan oleh penumpukan plak di arteri (aterosklerosis), yang dapat menghambat aliran darah dan memicu masalah serius seperti serangan jantung atau stroke. Menurut WHO dan CDC, penyakit jantung adalah penyebab utama kematian di Inggris, Amerika Serikat, Kanada dan Australia. Jumlah orang dewasa yang terdiagnosis penyakit jantung adalah 26,6 juta (11,3%) dari populasi orang dewasa (Utomo & Mesran, 2020).

Deteksi dini penyakit jantung kronis sangat penting karena memungkinkan intervensi lebih awal yang dapat mencegah perkembangan penyakit dan komplikasi serius seperti serangan jantung dan stroke. Pengobatan pada tahap awal biasanya lebih efektif, mengurangi biaya perawatan kesehatan jangka panjang, dan meningkatkan kualitas hidup pasien. Dengan deteksi dini, faktor risiko seperti hipertensi dan kolesterol tinggi dapat dikelola lebih baik, dan pasien dapat lebih cepat mengadopsi gaya hidup sehat yang mendukung kesehatan jantung.

Deteksi dini penyakit jantung sangat penting untuk mengurangi risiko penyakit jantung. Namun, masih banyak orang yang belum tahu bagaimana cara mengurangi risiko tersebut. Penelitian ini menggunakan algoritma regresi logistik, yang merupakan metode analisis statistik untuk memprediksi kemungkinan terjadinya suatu peristiwa dengan hasil biner (dua pilihan) berdasarkan satu atau lebih variabel bebas. Metode ini sangat berguna untuk klasifikasi, seperti menentukan apakah seorang pasien berisiko terkena penyakit jantung (ya/tidak). Regresi logistik adalah teknik machine learning yang efektif dalam memprediksi probabilitas kejadian, termasuk dalam analisis data pasien untuk deteksi dini penyakit jantung. Dengan regresi logistik, faktor risiko utama dapat diidentifikasi dan individu dapat dikelompokkan berdasarkan tingkat risiko mereka, sehingga intervensi pencegahan bisa lebih tepat sasaran dan efisien. Penelitian ini bertujuan untuk meningkatkan kesadaran masyarakat tentang faktor risiko penyakit jantung dan pentingnya deteksi dini melalui teknologi machine learning .

Penelitian ini mengembangkan sistem analisis interaktif yang menyediakan pemodelan dan manipulasi data. Sistem ini dapat mendukung berbagai layanan kesehatan masyarakat untuk memberikan informasi kepada masyarakat mengenai faktor risiko penyakit jantung melalui deteksi dini. Setelah mengkaji literatur dari penelitian sebelumnya berjudul "GAMBARAN PENGETAHUAN BANTUAN HIDUP DASAR (BHD) PADA ANGGOTA KELUARGA YANG MEMILIKI FAKTOR RISIKO PENYAKIT JANTUNG DI DENPASAR TIMUR" (Pertiwi et al., 2021), penelitian ini berbeda karena berfungsi sebagai pendukung bagi penelitian yang sedang dilakukan. Tujuan utama penelitian ini adalah melakukan klasifikasi data secara efektif, dengan judul "ANALISIS DETEKSI DINI PENYAKIT KRONIS DENGAN PENDEKATAN REGRESI LOGISTIK PADA DATA PASIEN".

Berdasarkan pada latar belakang masalah di atas, maka rumusan masalah pada penelitian ini yaitu “

1. Bagaimana penerapan metode clustering dan regresi logistik dalam mengklasifikasikan risiko penyakit jantung pada pasien di RSUD Haji Makassar?

2. Bagaimana tingkat akurasi nilai dari metode regresi logistik untuk melakukan klasifikasi deteksi dini penyakit jantung diRSUD Haji makassar?

Berdasarkan pada latar belakang masalah, rumusan masalah di atas, maka tujuan yang ingin dicapai dari penelitian ini yaitu :

1. Menerapkan metode clustering dan regresi logistik untuk mengklasifikasikan risiko penyakit jantung pada pasien di RSUD Haji Makassar.

2. untuk mendapatkan nilai akurasi yang baik penggunaan metode regresi logistic dalam mengelompokkan dan mengklasifikasikan pasien dengan risiko penyakit jantung di RSUD Haji makassar

Berdasarkan pada uraian latar belakang , rumusan masalah dan tujuan penelitian di atas, maka manfaat penelitian ini sebagi berikut

1. Bagi penulis, Penelitian ini memberikan kesempatan bagi penulis untuk mengembangkan keterampilan dalam analisis data dan penerapan metode machine learning, seperti clustering dan regresi logistik, yang sangat relevan dalam bidang ilmu data dan kesehatan.

2. Bagi mahasiswa, Penelitian ini dapat menjadi referensi akademis yang berguna bagi mahasiswa lain yang sedang mengerjakan tugas akhir atau penelitian serupa di bidang kesehatan dan ilmu data.

3. Bagi universitas, Penelitian ini dapat meningkatkan reputasi universitas sebagai institusi yang mendukung penelitian berkualitas tinggi di bidang ilmu data dan kesehatan.

4. Pengambilan data pasien dari Rumah Sakit Haji Makassar sebagai sumber data utama.

5. Investigasi faktor-faktor yang berpengaruh terhadap akurasi deteksi, dengan fokus pada data pasien yang berasal dari Rumah Sakit Haji Makassar.

6. Eksplorasi kontribusi variabel klinis terhadap keberhasilan deteksi dini penyakit kronis (jantung) menggunakan Regresi Logistik, menggunakan data pasien yang diperoleh dari Rumah Sakit Haji Makassar.

**2. Metode Penelitian**

Tempat dan Waktu Penelitian

1. Tempat Penelitian

Tempat penelitian merupakan lokasi dan objek yang akan digunakan pada suatu penelitian. Lokasi pengambilan data pasien pada penelitian ini dilakukan di Rumah Sakit Umum Daerah (RSUD) Haji Makassar di Pintu keluar rs haji, Jl. Dg. Ngeppe, Balang Baru, Kec. Tamalate, Kota Makassar, Sulawesi Selatan 90122.

1. Waktu Penelitian

Jadwal penelitian yang akan dilaksanakan dimulai pada bulan April sampai bulan agustus sampai semua proses pengumpulan data selesai.

Alat dan Bahan

1. Kebutuhan Hardware ( perangkat keras )

1. Laptop HP EliteBook x360 1030 G4

2. RAM 16 GB

2. Kebutuhan Software ( Perangkat Lunak)

1. Visual code

2. Jupiter

3. Microsoft Excel

4. python

Perancangan Sistem

Perancangan sistem merupakan tahap penting dalam pengembangan sistem yang melibatkan pemodelan dan konstruksi sistem untuk memenuhi kebutuhan khusus dan mengatasi masalah yang diidentifikasi, terutama dalam konteks penelitian ini. Flowchart membantu dalam mengorganisir informasi, memvisualisasikan hubungan antarproses, dan menyediakan panduan yang jelas bagi pengembang dan pengguna sistem. Dengan menggunakan flowchart, perancangan sistem dapat dilakukan secara terstruktur dan efisien.

Flowcart Perancangan Sistem

Gambar 2. Teknik Perancangan Sistem

Penjelasan gambar 2. Teknik Perancangan

1. Mulai: Memulai proses pembuatan model.

2. Pengumpulan Data: Mengumpulkan data pasien dan variabel yang diperlukan.Praproses Data: Membersihkan dan mempersiapkan data untuk analisis.

3. Melakukan clustering untuk mengelompokkan data yang lebih akurat

4. Pemilihan Variabel Independen: Memilih variabel yang paling berpengaruh terhadap penyakit jantung.

5. Pendefinisian Model Regresi Logistik: Menentukan model regresi logistik dengan variabel independen yang dipilih.

6. Evaluasi Model: Mengevaluasi kinerja model menggunakan data uji

7. Interpretasi Hasil: Menginterpretasikan koefisien model untuk memahami faktor risiko penyakit jantung.

8. Selesai: Mengakhiri proses pembuatan model dan implementasi

Adapun flowchart dari algoritma Regresi Logistik Sebagai Berikut:

Berikut penjelasan lebih detail untuk setiap bagian dari flowchart regresi logistik untuk deteksi penyakit jantung:

1. Start

Deskripsi: Langkah awal memulai proses deteksi penyakit jantung menggunakan model regresi logistik. Pada tahap ini, kita merencanakan proses dan mengidentifikasi tujuan analisis.

2. Pengumpulan Data

Deskripsi: Data pasien dikumpulkan dari berbagai sumber, seperti rekam medis rumah sakit, survei kesehatan, atau dataset publik yang berisi fitur relevan terkait dengan penyakit jantung. Fitur yang sering digunakan termasuk usia, jenis kelamin, tekanan darah, kolesterol, dll.

Tujuan: Mengumpulkan data yang cukup untuk melatih dan menguji model.

3. Preprocessing Data

Data Cleaning: Pada langkah ini, data dibersihkan dari nilai yang hilang (missing values), outliers, atau data yang tidak relevan. Data yang tidak lengkap atau rusak perlu diatasi agar tidak memengaruhi hasil analisis.

Normalisasi/Standardisasi: Fitur numerik yang memiliki skala berbeda dinormalisasi atau distandarisasi untuk membuat model lebih stabil dan efisien. Misalnya, usia dan tekanan darah mungkin memiliki rentang yang berbeda, sehingga mereka perlu dinormalisasi.

4. Pembagian Dataset

Deskripsi: Data yang telah diproses dibagi menjadi dua subset utama:

Training Set (Data Pelatihan): Bagian data yang digunakan untuk melatih model regresi logistik.

Testing Set (Data Pengujian): Bagian data yang digunakan untuk menguji model setelah pelatihan, sehingga kita dapat mengevaluasi kinerjanya. Proporsi umum pembagian data adalah 80% untuk pelatihan dan 20% untuk pengujian, meskipun ini bisa bervariasi.

Tujuan: Memastikan bahwa model tidak hanya belajar dari data pelatihan, tetapi juga dapat menghasilkan prediksi yang baik pada data baru (data pengujian).

5. Pelatihan Model (Training)

Deskripsi: Pada tahap ini, model regresi logistik dilatih menggunakan data pelatihan. Regresi logistik bekerja dengan mencoba menemukan hubungan antara fitur input (X) dan output biner atau kategori (Y). Misalnya, model akan belajar bagaimana usia, tekanan darah, dan faktor lainnya memengaruhi kemungkinan seseorang terkena penyakit jantung.

Tujuan: Membangun model prediktif yang dapat mengidentifikasi risiko penyakit jantung berdasarkan data fitur.

6. Evaluasi Model

Deskripsi: Setelah model dilatih, kinerjanya diuji menggunakan data pengujian. Metode evaluasi meliputi:

Akurasi: Persentase prediksi yang benar dari keseluruhan prediksi.

Precision: Jumlah prediksi positif benar dari semua prediksi positif yang dibuat.

Recall (Sensitivity): Kemampuan model untuk mendeteksi semua kasus positif yang sebenarnya.

AUC (Area Under Curve): Mengukur kemampuan model untuk memisahkan kelas dengan baik, di mana nilai AUC mendekati 1 menunjukkan kinerja yang sangat baik.

Tujuan: Mengevaluasi seberapa baik model memprediksi risiko penyakit jantung pada data yang tidak terlihat sebelumnya (data pengujian).

7. Prediksi

Deskripsi: Setelah model dievaluasi, model digunakan untuk memprediksi risiko penyakit jantung pada pasien baru. Regresi logistik menghasilkan probabilitas antara 0 dan 1. Misalnya, jika probabilitasnya lebih besar dari ambang batas tertentu (misalnya 0.5), maka pasien dikategorikan memiliki risiko tinggi terkena penyakit jantung.

Tujuan: Membuat prediksi risiko penyakit jantung berdasarkan fitur pasien.

8. Keputusan

Deskripsi: Berdasarkan hasil prediksi, pasien diklasifikasikan ke dalam kategori risiko tertentu (misalnya risiko rendah, tinggi, atau sangat tinggi). Keputusan ini bisa digunakan oleh dokter untuk mengambil langkah lebih lanjut dalam diagnosis dan pengobatan.

Tujuan: Memberikan hasil yang bisa digunakan untuk pengambilan keputusan klinis.

9. End

Deskripsi: Tahap akhir di mana proses deteksi selesai. Hasilnya kemudian bisa diserahkan kepada dokter atau digunakan dalam sistem untuk tindakan lanjutan seperti pengobatan atau perencanaan perawatan.

Tujuan: Menyelesaikan alur proses dan menyediakan hasil yang bermanfaat bagi pengambil keputusan.

Teknik Pengujian

Gambar 4. Flowchart Teknik pengujian

Penjelasan gambar 4. Flowchart Teknik Pengujian

Tahap Awal: Langkah pertama dalam pengujian sistem, menandakan dimulainya proses evaluasi.

Masukan Data: Informasi pasien yang diperlukan untuk analisis dimasukkan ke dalam sistem.

Pembagian Data Latih dan Uji: Informasi pasien dipisahkan menjadi dua kelompok terpisah, yaitu data latih untuk melatih model dan data uji untuk menguji performanya.

Pelatihan Model Regresi Logistik dengan Data Latih: Model regresi logistik dilatih menggunakan data latih untuk mengajarkan model cara mengklasifikasikan data dengan akurat.

Validasi Model dengan Data Uji: Model regresi logistik yang telah dilatih divalidasi menggunakan data uji untuk mengevaluasi seberapa baik model dapat memprediksi hasil yang benar pada data yang belum pernah dilihat sebelumnya.

Evaluasi Hasil: Setelah validasi, performa model regresi logistik dievaluasi termasuk akurasi, presisi, recall, dan metrik evaluasi lainnya untuk menilai kemampuan model dalam mendeteksi penyakit jantung pada tahap awal.

Penilaian Performa Model Regresi Logistik dalam Deteksi Penyakit Jantung pada Tahap Awal: Langkah terakhir adalah menentukan secara akurat kinerja model regresi logistik dalam mendeteksi penyakit jantung pada tahap awal berdasarkan hasil evaluasi menggunakan data uji.

Teknik Analisis Data

deteksi dini penyakit jantung menggunakan regresi logistik pada dataset pasien, proses analisis data melibatkan beberapa langkah. Langkah-langkah tersebut meliputi pengumpulan informasi pasien, pra-pemrosesan data, pembagian data menjadi data latih dan data uji, pelatihan model regresi logistik, validasi dengan data uji, evaluasi performa model, hingga analisis hasil evaluasi untuk menentukan keandalan model dalam mendeteksi penyakit jantung pada tahap awal penelitian.

**3. Hasil dan diskusi**

1. Data yang diperoleh mencakup 640 data pasien penyakit jantung,yang diambil dari catatan rekam medis pasien RSUD Haji Makassar, yang bersumber dari poli jantung dan pembuluh darah .data tersebut konsultasi dari Dr.Husnul Khatimah,S.Ked dokter koas RS haji Makassar dan Dr.ANDI MUHAMMAD REIS R SAIBY, Sp.JP sebagai dokter spesialis jantung RS Haji Makassar, sehingga mendapatkan data sebagai berikut :

2. Tabel 1. Data mentah

1. Data Cleaning Dan Feature Selection: Menghapus atau memperbaiki data yang hilang, tidak valid, atau anomali. Serta Memilih fitur-fitur yang paling relevan untuk analisis, seperti usia, glukosa, ureum, kreatinin , sgot , sgpt , Tekanan Darah.

Tabel 2. Data olahan

1. Membagi dua hasil tekanan darah pasien yaitu : systolic dan Diastolic, Tekanan sistolik adalah angka pertama atau yang lebih tinggi dalam pengukuran tekanan darah. Ini mengukur tekanan di arteri ketika jantung berkontraksi dan memompa darah ke seluruh tubuh. Tekanan diastolik adalah angka kedua atau yang lebih rendah dalam pengukuran tekanan darah. Ini mengukur tekanan di arteri ketika jantung sedang dalam fase relaksasi (diastole), yaitu ketika jantung mengisi kembali dengan darah.

2. Tabel 3. hasil systolic dan diastolic

Clustering adalah teknik analisis data yang digunakan untuk mengelompokkan sekumpulan objek atau data berdasarkan kemiripan mereka. Tujuan dari clustering adalah untuk memisahkan data ke dalam grup (atau kluster) yang anggotanya memiliki karakteristik yang mirip satu sama lain dan berbeda dari anggota di kluster lain.

Pada clustering ini mengguakan metode Elbow Adapun Langkah-langkahnya sebagi berikut:

1. Persiapan data

2. Pengumpulan Data: Kumpulkan semua data yang relevan dari pasien, termasuk atribut-atribut seperti tekanan darah, kadar kolesterol, usia, jenis kelamin, dll.

3. Preprocessing: Bersihkan data dari nilai yang hilang atau anomali dan lakukan normalisasi atau standarisasi data jika variabel memiliki skala yang berbeda-beda.

4. Standarisasi Fitur

5. Gunakan StandardScaler atau metode standarisasi lainnya untuk memastikan bahwa semua fitur memiliki rata-rata nol dan standar deviasi satu. Ini penting terutama jika skala fitur berbeda.

6. Menerapkan K-Means Clustering dengan Berbagai Jumlah Klaster

7. Rentang Klaster: Tentukan rentang jumlah klaster yang akan diuji, misalnya dari 1 hingga 10.

8. Penerapan K-Means: Terapkan algoritma K-Means untuk setiap jumlah klaster dalam rentang tersebut. Pada setiap iterasi, hitung inertia, yaitu jumlah kuadrat jarak dari setiap titik data ke centroid klaster terdekat.

9. Menghitung Inertia (Sum of Squared Distances)

10. Inertia adalah ukuran seberapa kompak klaster yang terbentuk. Semakin kecil inertia, semakin baik klaster yang terbentuk. Namun, seiring bertambahnya jumlah klaster, inertia selalu berkurang.

11. Memplot Inertia terhadap Jumlah Klaster

12. Buat plot dengan sumbu X menunjukkan jumlah klaster dan sumbu Y menunjukkan inertia. Cari titik di mana penurunan inertia mulai melambat secara signifikan, membentuk bentuk "siku" pada grafik.

13. Menetukan jumlah Klaster optimal

14. Identifikasi titik "siku" pada grafik. Jumlah klaster pada titik ini dianggap optimal karena menambah lebih banyak klaster setelah titik ini memberikan pengurangan inertia yang marginal.

Berikut cara clutering menggunakan metode Elbow pada pemograman phyton

# penggunaan standar skalar

scaler = StandardScaler()

data\_scaled = scaler.fit\_transform(data\_cluster)

# Determine the optimal number of clusters using the Elbow Method

sse = []

for k in range(1, 11):

kmeans = KMeans(n\_clusters=k, random\_state=42)

kmeans.fit(data\_scaled)

sse.append(kmeans.inertia\_)

# Plot the SSE for each number of clusters

plt.figure(figsize=(10, 6))

plt.plot(range(1, 11), sse, marker='o')

plt.xlabel('Jumlah Kluster')

plt.ylabel('Sum of squared distances')

plt.title('Metode Elbow untuk penetapan jumlah Kluster')

plt.show()

# Based on the Elbow Method plot, determine the optimal number of clusters (e.g., 3)

optimal\_clusters = 3

kmeans = KMeans(n\_clusters=optimal\_clusters, random\_state=42)

data['Kluster'] = kmeans.fit\_predict(data\_scaled)

# Display the cluster assignments

data[['NO', 'NAMA PASIEN', 'Cluster']]

Pada gambar tersebut, diperlihatkan hasil plot dari Metode Elbow untuk menentukan jumlah klaster yang optimal pada data pasien jantung. Pada sumbu horizontal (X), terdapat jumlah klaster yang diuji, mulai dari 1 hingga 10. Sedangkan pada sumbu vertikal (Y), terdapat nilai inertia, yang merupakan jumlah kuadrat dari jarak antara titik-titik data dengan centroid klasternya.

Dalam penetapan cluster setiap pasien penyakit jantung setelah melakukan analisis sebelumnya menggunakan metode elbow dapat diperolah hasil cluster ada 3 jumlah clustering yang optimal. Setelah itu file hasil clustering disimpan kedalam file excel.Hasil klustering tersebut akan diinisialisasi menggunakan algoritma k-means.

Implementasi dalam inisialisasi algorima k-means

1. Menentukan Jumlah Klaster Optimal

2. optimal\_clusters = 3

3. Pada bagian ini, variabel optimal\_clusters diatur dengan nilai 3. Nilai ini ditentukan berdasarkan analisis yang telah dilakukan sebelumnya menggunakan Metode Elbow, yang menunjukkan bahwa jumlah klaster optimal adalah 3.

4. Inisialisasi dan Pelaksanaan Algoritma K-Means

kmeans = KMeans(n\_clusters=optimal\_clusters, random\_state=42)

data['Cluster'] = kmeans.fit\_predict(data\_scaled)

1. KMeans(n\_clusters=optimal\_clusters, random\_state=42): Algoritma K-Means diinisialisasi dengan parameter n\_clusters=3, yang artinya data akan dikelompokkan menjadi tiga klaster. Parameter random\_state=42 digunakan untuk memastikan bahwa hasil klasterisasi konsisten setiap kali algoritma dijalankan.

2. fit\_predict(data\_scaled): Metode fit\_predict diterapkan pada data yang telah diskalakan (data\_scaled). Metode ini melakukan dua hal: menyesuaikan (fit) model K-Means dengan data dan memprediksi klaster untuk setiap sampel data. Hasilnya adalah array berisi label klaster untuk setiap titik data.

3. data['Cluster'] = ...: Label klaster yang dihasilkan kemudian disimpan dalam kolom baru bernama 'Cluster' di dalam dataframe data. Ini menunjukkan klaster mana yang telah ditetapkan untuk setiap titik data.

4. Menyimpan Hasil Klasterisasi ke File Excel

# Save the results to an Excel file

output\_path = 'Hasil Kluster.xlsx'

data.to\_excel(output\_path, index=False)

1. output\_path = 'Hasil Kluster.xlsx': Menentukan jalur dan nama file output untuk menyimpan hasil klasterisasi.

2. data.to\_excel(output\_path, index=False): Metode to\_excel digunakan untuk menyimpan dataframe data ke dalam file Excel dengan nama 'Hasil Kluster.xlsx'. Parameter index=False digunakan untuk menghindari penyertaan indeks dataframe sebagai kolom dalam file Excel.

Setelah hasil clustering disimpan dalam file excel data yang telah diklasterisasi dengan terlebih dahulu mengidentifikasi kolom yang non-numerik, kemudian memisahkan data numerik untuk analisis lebih lanjut. Langkah selanjutnya adalah menghitung rata-rata untuk setiap klaster berdasarkan fitur numerik yang tersedia, yang menghasilkan centroid untuk masing-masing klaster.

Menampilkan hasil clustering pada data pasien penyakit jantung

Dari hasil klasterisasi ini, kita dapat melihat bagaimana data dalam setiap klaster memiliki karakteristik yang berbeda berdasarkan fitur yang diukur (misalnya, Systolic, Diastolic).

Klaster 0: Mempunyai rata-rata tekanan darah sistolik dan diastolik yang lebih rendah dibandingkan dengan klaster lain, yang mungkin menunjukkan pasien dengan tekanan darah lebih normal atau rendah.

Klaster 1: Menunjukkan rata-rata tekanan darah yang lebih tinggi, mungkin menunjukkan pasien dengan hipertensi.

Klaster 2: Memiliki nilai rata-rata tertentu untuk beberapa fitur, dan juga tekanan darah yang berbeda dengan klaster lainnya.

Adapun hasil dari cluster pada pasie jantung yang telah dimasukkan kedalam file excel sebagai berikut

Tabel 4. Hasil clustering

Regresi logistik adalah teknik analisis statistik yang digunakan untuk memodelkan hubungan antara variabel independen (prediktor) atau variabel biner. Secara umum, regresi logistik digunakan ketika variabel dependen memiliki dua komponen (seperti “ya” atau “tidak”, “lulus” atau “gagal”).

Dalam melakukan permodelan regresi logistic menggunakan scikirt learn ada beberapa tahapan yang akan dilakukan sehingga mendapatkan hasil yang maksimal yaitu

1. Normalisasi data : memastikan bahwa semua fitur berkontribusi secara seimbang pada model dan tidak mendominasi model karena skala yang berbeda.

2. Pelatihan model regresi logistic : Model ini akan memprediksi apakah seorang pasien memiliki penyakit jantung berdasarkan fitur-fitur seperti usia, tekanan darah, kadar kolesterol, dll.

3. Evaluasi model : mengevaluasi kinerjanya menggunakan data pengujian.

IMPLEMENTASI

1. Normalisasi data : StandardScaler adalah alat untuk melakukan standardisasi fitur, yang mengubah nilai-nilai fitur dalam dataset sehingga memiliki rata-rata nol dan deviasi standar sa tu.

# Normalisasi data

scaler = StandardScaler()

X = scaler.fit\_transform(X)

1. Pelatihan model regresi logistic : menginisialisasi dan melatih model regresi logistik menggunakan pustaka scikit-learn

# Menginisialisasi dan melatih model regresi logistik

model = LogisticRegression(max\_iter=2000)  # Meningkatkan max\_iter

model.fit(X\_train, y\_train)

1. Evaluasi model : untuk mengevaluasi performa model yang telah dilatih dengan menggunakan metrik akurasi dan laporan klasifikasi.

# Mengevaluasi model

akurasi = accuracy\_score(y\_test, y\_pred)

laporan = classification\_report(y\_test, y\_pred)

HASIL PREDIKSI

Hasil prediksi menunjukkan bahwa model regresi logistik dapat memisahkan kelas dengan baik. Laporan langsung dan klasifikasi memberikan gambaran umum tentang kinerja model. Hasil ini penting dalam mengidentifikasi pasien dengan gagal jantung yang signifikan sehingga intervensi yang tepat dapat diberikan.

Tabel 5. Hasil Akurasi

Penjelasan mengenai metrik-metrik dalam laporan klasifikasi:

Model Regresi logistic: memiliki akurasi sebesar 98.44%. Artinya, dari 128 sampel data pengujian, model berhasil memprediksi 126 sampel dengan benar.

Laporan Klasifikasi

menunjukkan hasil evaluasi model regresi logistik dengan metrik precision, recall, f1-score, dan support untuk tiga kelas berbeda (0, 1, dan 2). Berikut penjelasan mengenai hasil tersebut:

Akurasi

Akurasi: 0.984375 (98.44%)

Akurasi ini menunjukkan bahwa model mampu memprediksi dengan benar 98.44% dari total sampel. Dari 128 sampel, model membuat sekitar 126 prediksi yang benar. Ini adalah tingkat akurasi yang sangat tinggi, menunjukkan bahwa model regresi logistik bekerja sangat baik pada data ini.

Evaluasi Per Kelas

Kelas 0

Precision: 0.974684

Precision sebesar 97.47% berarti dari semua prediksi yang dilakukan untuk kelas 0, sekitar 97.47% benar-benar merupakan kelas 0.

Recall: 1.00

Recall sebesar 100% berarti model berhasil mendeteksi semua sampel yang benar-benar termasuk dalam kelas 0.

F1-score: 0.987179

F1-score, harmonik rata-rata dari precision dan recall, sebesar 98.72% menunjukkan bahwa model bekerja sangat baik dalam mendeteksi kelas 0.

Support: 77

Ini menunjukkan bahwa terdapat 77 sampel sebenarnya yang termasuk dalam kelas 0.

Kelas 1

Precision: 1.00

Precision sebesar 100% berarti semua prediksi yang dilakukan untuk kelas 1 benar-benar merupakan kelas 1.

Recall: 0.979592

Recall sebesar 97.96% berarti model mampu mendeteksi hampir semua sampel yang benar-benar termasuk dalam kelas 1, dengan hanya sedikit yang terlewatkan.

F1-score: 0.989691

F1-score sebesar 98.97% menunjukkan bahwa model memiliki kinerja yang sangat baik dalam mendeteksi kelas 1.

Support: 49

Ini menunjukkan bahwa terdapat 49 sampel sebenarnya yang termasuk dalam kelas 1.

Kelas 2

Precision: 1.00

Precision sebesar 100% berarti semua prediksi yang dilakukan untuk kelas 2 benar-benar merupakan kelas 2.

Recall: 0.50

Recall sebesar 50% berarti model hanya mampu mendeteksi setengah dari sampel yang benar-benar termasuk dalam kelas 2. Ini menunjukkan adanya kelemahan model dalam mendeteksi kelas ini.

F1-score: 0.666667

F1-score sebesar 66.67% menunjukkan bahwa meskipun precision tinggi, recall yang rendah menyebabkan F1-score yang lebih rendah untuk kelas ini.

Support: 2

Ini menunjukkan bahwa hanya ada 2 sampel sebenarnya yang termasuk dalam kelas 2, yang mungkin menyebabkan ketidakstabilan dalam metrik evaluasi untuk kelas ini.

Rata-Rata

Macro avg (Rata-rata makro):

Precision: 0.991561

Recall: 0.826531

F1-score: 0.881179

Ini adalah rata-rata dari precision, recall, dan F1-score dari semua kelas tanpa mempertimbangkan jumlah sampel per kelas. Rata-rata ini memberikan gambaran umum tentang kinerja model di setiap kelas, tetapi tidak mempertimbangkan ketidakseimbangan kelas.

Weighted avg (Rata-rata berbobot):

Precision: 0.984771

Recall: 0.984375

F1-score: 0.983133

Ini adalah rata-rata dari precision, recall, dan F1-score yang mempertimbangkan jumlah sampel di setiap kelas. Ini lebih representatif dari performa keseluruhan model ketika ada ketidakseimbangan kelas.

Selanjutnya adalah Penginputan Hasil, Model diinput kedalam file excel

# Mengevaluasi model

akurasi = accuracy\_score(y\_test, y\_pred)

laporan = classification\_report(y\_test, y\_pred, output\_dict=True)

# Konversi laporan klasifikasi menjadi DataFrame

laporan\_df = pd.DataFrame(laporan).transpose()

# Membuat DataFrame untuk data uji beserta prediksinya

df\_test = df.loc[y\_test.index].copy()

df\_test['Prediksi Kluster'] = y\_pred

# Ekspor hasil ke file Excel

with pd.ExcelWriter('Hasil\_Validasi\_Kluster\_Test.xlsx', engine='openpyxl') as writer:

df\_test.to\_excel(writer, sheet\_name='Data Uji dan Prediksi', index=False)

laporan\_df.to\_excel(writer, sheet\_name='Laporan Klasifikasi')

# Menambahkan akurasi sebagai lembar baru

pd.DataFrame({'Akurasi': [akurasi]}).to\_excel(writer, sheet\_name='Akurasi', index=False)

print("Hasil validasi dan data uji telah diekspor ke file Excel 'Hasil\_Validasi\_Kluster\_Test.xlsx'.")

hasil prediksi telah diekspor ke file

Tabel 6. Prediksi Cluster

Model regresi logistik yang diterapkan menunjukkan skor recall dan f1 terbaik dan terakurat (98,44%), hampir sempurna untuk ketiga kelas. Hal ini menunjukkan bahwa model sangat efektif dalam membedakan kelas-kelas pada kumpulan data ini, dengan kesalahan prediksi yang sangat rendah. Model ini dapat dianggap andal untuk digunakan dalam konteks yang mirip dengan data eksperimen, seperti prediksi berdasarkan kondisi yang tersedia..

1. Pengujian Sistem Clustering Elbow

2. Pengujian sistem clustering Elbow dengan melihat hasil kluster pada kolom kluster dimana nilai 0 itu mempresentasikan resiko rendah penyakit jantung sedangkan nilai 1 mempresentasikan resiko tinggi penyakit jantung,dan nilai 2 mempresentasikan resiko sangat tinggi penyakit jantung.

Tabel 7. clustering

Selanjutnya mencari jumlah data kluster beresiko tinggi dan beresiko rendah dari total 640 data pasien deteksi dini.

Tabel 8. Jumlah prediksi

Berdasarkan data prediksi klaster, dapat disimpulkan bahwa mayoritas individu, sebanyak 358 orang, masuk ke dalam Klaster 0, yang menunjukkan kondisi kesehatan yang relatif stabil atau normal. Sebanyak 268 individu masuk ke dalam Klaster 1, yang kemungkinan memiliki peningkatan risiko kesehatan dibandingkan dengan Klaster 0. Sementara itu, hanya 14 individu yang termasuk dalam Klaster 2, Tahap Analisis ini berguna dalam mengidentifikasi terkait pasien penyakit jantung. Hal ini telah sesuai dengan yang dijelaskan oleh narasumber yaitu dr.Husnul Khatimah S,ked Selaku dokter Rumah Sakit Umum Daerah ( RSUD) Haji Makassar mengatakan bahwa dapat dilihat dari hasil pemeriksaan laboratorium yang menunjukkan bahwa pasien yang mengalami peningkatan kadar ureum dan kreatinin,mengindikasikan adanya kerusakan fungsi ginjal.serta hipertensi yang dapat menyebabkan penyumbatan aterosklerosis dan glukosa dapat menyebabkan diabetes yang juga dapat meningkatkan aterosklerosis. Selain itu ,kadar SGOT dan SGPT yang tinggi mengindikasikan gangguan pada hati dan jantung pasien.

1. Pengujian sistem klasifikasi dan Prediksi Regresi Logistik

2. Pengujian Sistem klasifikasi dan prediksi Regresi Logistik Dilakukan dengan cara manual Menghitung regresi logistik secara manual memerlukan beberapa langkah matematis yang melibatkan penggunaan fungsi logistik

Menghitung manual dalam metode regresi logistic

Asumsi Baru:

Dari 77 pasien dengan prediksi kategori rendah, 76 pasien sebenarnya rendah (True Negative, TN) dan 1 pasien sebenarnya tinggi (False Negative, FN).

Dari 49 pasien dengan prediksi kategori tinggi, 48 pasien sebenarnya tinggi (True Positive, TP) dan 1 pasien sebenarnya sangat tinggi (False Positive, FP).

Dari 2 pasien dengan prediksi kategori sangat tinggi, 2 pasien benar-benar berada di kategori ini (True Positive, TP).

Matriks Kontingensi Berdasarkan Asumsi:

Tabel 9.laporan klasifikasi

Menghitung Metode Evaluasi

Akurasi:

Akurasi

1. Presisi (precision) untuk kategori tinggi

2. presisi = 0.98 atau 98%

3. Recall (sensivity) untuk kategori tinggi

4. Recall =

5. F1-score untuk kategori tinggi

6. F1-Score = 2 x

7. Presisi dan Recall untuk kategori sangat tinggi:

8. Karena hanya ada dua prediksi dan semuanya benar:

Dengan akurasi 99.22%, model ini menunjukkan kinerja yang sangat baik:

Akurasi: 98.44%

Presisi: 98%

Recall: 98.%

F1-Score: 98%

Hasil prediksi model klasifikasi menunjukkan akurasi yang tinggi dengan 126 prediksi benar dari seluruh peserta uji. Ini menghasilkan persentase prediksi akurat sebesar 98.44%. Data tersebut menunjukkan bahwa model klasifikasi yang digunakan mampu memprediksi hasil paling akurat dengan akurasi hampir 100%. Bagian prediksi dibagi menjadi tiga bagian yaitu “Rendah”,“Tinggi”, dan “Sangat Tinggi yang menampilkan hasil evaluasi masing-masing peserta sesuai kriteria yang telah ditetapkan. Keberhasilan ini menunjukkan potensi besar model klasifikasi untuk digunakan dalam berbagai situasi yang memerlukan akurasi prediksi tinggi

**4. Kesimpulan**

1. Penelitian ini menggunakan metode clustering dan regresi logistik untuk menganalisis data 640 pasien jantung di RSUD Haji Makassar. Dari hasil analisis cluster menggunakan metode Elbow dan K-Means ditemukan tiga cluster optimal yang dapat mengklasifikasikan data pasien. Kelompok-kelompok ini memiliki karakteristik yang berbeda-beda: kelompok 0 mencakup pasien dengan risiko penyakit jantung rendah, kelompok 1 mencakup pasien dengan risiko penyakit jantung tinggi. Dan kelompok 2 mencakup pasien dengan risiko penyakit jantung sangat tinggi

2. Dalam model regresi logistik menunjukkan tingkat akurasi yang tinggi yaitu 98.44% . hal ini menunjukkan bahwa metode regresi logitik sangat akurat dalam mendeteksi penyakit jantung

Dari kesimpulan yang telah dijelaskan Adapun saran dari penelitian ini yaitu:

Penelitian ini dapat diperluas dengan menggunakan metode machine learning lain, seperti Random Forest atau Neural Networks, untuk membandingkan hasil prediksi dan akurasi yang lebih baik dalam klasifikasi risiko penyakit jantung.

**Referensi**

[1] L

[2] A

[3] M

[4] P

[5] I

[6] R

[7] A

[8] N

[9] Lampiram 1. Source code

[10] import pandas as pd

[11] import numpy as np

[12] from sklearn.preprocessing import StandardScaler

[13] from sklearn.cluster import KMeans

[14] import matplotlib.pyplot as plt

[15] # Memuat Data

[16] data = pd.read\_excel('DATA JANTUNG RM(3).xlsx')

[17] data.info()

[18] data.head()

[19] # Kolom Patokan Kluster

[20] columns = ['Glukosa', 'Ureum', 'Kreatinin', 'SGOT', 'SGPT', 'Tekanan darah']

[21] # Pembersihan kolom tekanan darah dan bagi dua ke dalam dua kolom

[22] data['Tekanan darah'] = data['Tekanan darah'].str.replace(r'\D', '/')

[23] data[['Systolic', 'Diastolic']] = data['Tekanan darah'].str.split('/', expand=True).astype(float)

[24] columns.extend(['Systolic', 'Diastolic'])

[25] # masukkan Systolic dan Diastolic ke dalam kolom inputan

[26] columns = ['Ureum', 'Kreatinin', 'SGOT', 'SGPT', 'Systolic', 'Diastolic']

[27] data\_cluster = data[columns]

[28] # penanganan nilai kosong

[29] data\_cluster = data\_cluster.fillna(data\_cluster.mean())

[30] # penggunaan standar skalar

[31] scaler = StandardScaler()

[32] data\_scaled = scaler.fit\_transform(data\_cluster)

[33] # Determine the optimal number of clusters using the Elbow Method

[34] sse = []

[35] for k in range(1, 11):

[36] kmeans = KMeans(n\_clusters=k, random\_state=42)

[37] kmeans.fit(data\_scaled)

[38] sse.append(kmeans.inertia\_)

[39] # Plot the SSE for each number of clusters

[40] plt.figure(figsize=(10, 6))

[41] plt.plot(range(1, 11), sse, marker='o')

[42] plt.xlabel('Jumlah Kluster')

[43] plt.ylabel('Sum of squared distances')

[44] plt.title('Metode Elbow untuk penetapan jumlah Kluster')

[45] plt.show()

[46] # Based on the Elbow Method plot, determine the optimal number of clusters (e.g., 3)

[47] optimal\_clusters = 3

[48] kmeans = KMeans(n\_clusters=optimal\_clusters, random\_state=42)

[49] data['Cluster'] = kmeans.fit\_predict(data\_scaled)

[50] # Display the cluster assignments

[51] data[['NO', 'NAMA PASIEN', 'Cluster']]

[52] # Based on the Elbow Method plot, determine the optimal number of clusters (e.g., 3)

[53] optimal\_clusters = 3

[54] kmeans = KMeans(n\_clusters=optimal\_clusters, random\_state=42)

[55] data['Cluster'] = kmeans.fit\_predict(data\_scaled)

[56] # Save the results to an Excel file

[57] output\_path = 'Hasil Kluster.xlsx'

[58] data.to\_excel(output\_path, index=False)

[59] # Load the data

[60] file\_path = 'Hasil Kluster.xlsx'

[61] df = pd.read\_excel(file\_path)

[62] # Print data types to identify non-numeric columns

[63] print(df.dtypes)

[64] # Convert columns to numeric if needed (example column names)

[65] # df['Column1'] = pd.to\_numeric(df['Column1'], errors='coerce')

[66] # df['Column2'] = pd.to\_numeric(df['Column2'], errors='coerce')

[67] # Filter out non-numeric columns

[68] numeric\_data = df.select\_dtypes(include=[np.number])

[69] # Perform groupby and calculate the mean

[70] centroids = numeric\_data.groupby(df['Cluster']).mean()

[71] # Display the centroids

[72] print(centroids)

[73] # Load the data

[74] file\_path = 'Hasil Kluster.xlsx'

[75] df = pd.read\_excel(file\_path, sheet\_name='Sheet1')

[76] from sklearn.model\_selection import train\_test\_split

[77] from sklearn.linear\_model import LogisticRegression

[78] from sklearn.metrics import classification\_report, accuracy\_score

[79] from sklearn.preprocessing import StandardScaler

[80] import openpyxl

[81] # Memastikan bahwa kolom memiliki tipe data numerik

[82] df['Usia'] = pd.to\_numeric(df['Usia'], errors='coerce')

[83] df['Ureum'] = pd.to\_numeric(df['Ureum'], errors='coerce')

[84] df['Kreatinin'] = pd.to\_numeric(df['Kreatinin'], errors='coerce')

[85] df['SGOT'] = pd.to\_numeric(df['SGOT'], errors='coerce')

[86] df['SGPT'] = pd.to\_numeric(df['SGPT'], errors='coerce')

[87] df['Glukosa'] = pd.to\_numeric(df['Glukosa'], errors='coerce')

[88] df['Systolic'] = pd.to\_numeric(df['Systolic'], errors='coerce')

[89] df['Diastolic'] = pd.to\_numeric(df['Diastolic'], errors='coerce')

[90] df['Cluster'] = pd.to\_numeric(df['Cluster'], errors='coerce')

[91] # Mendefinisikan fitur input dan target

[92] fitur = ['Usia', 'Ureum', 'Kreatinin', 'SGOT', 'SGPT', 'Glukosa', 'Systolic', 'Diastolic']

[93] target = 'Cluster'

[94] # Mengekstraksi fitur input dan variabel target

[95] X = df[fitur]

[96] y = df[target]

[97] # Menangani nilai yang hilang (jika ada) dengan mengisinya dengan rata-rata kolom

[98] X = X.fillna(X.mean())

[99] # Normalisasi data

[100] scaler = StandardScaler()

[101] X = scaler.fit\_transform(X)

[102] # Membagi data menjadi set pelatihan dan pengujian

[103] X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

[104] # Menginisialisasi dan melatih model regresi logistik

[105] model = LogisticRegression(max\_iter=2000)  # Meningkatkan max\_iter

[106] model.fit(X\_train, y\_train)

[107] # Membuat prediksi pada set pengujian

[108] y\_pred = model.predict(X\_test)

[109] # Mengevaluasi model

[110] akurasi = accuracy\_score(y\_test, y\_pred)

[111] laporan = classification\_report(y\_test, y\_pred)

[112] print("Akurasi:", akurasi)

[113] print("Laporan Klasifikasi:\n", laporan)

[114] # Mengevaluasi model

[115] akurasi = accuracy\_score(y\_test, y\_pred)

[116] laporan = classification\_report(y\_test, y\_pred, output\_dict=True)

[117] # Konversi laporan klasifikasi menjadi DataFrame

[118] laporan\_df = pd.DataFrame(laporan).transpose()

[119] # Membuat DataFrame untuk data uji beserta prediksinya

[120] df\_test = df.loc[y\_test.index].copy()

[121] df\_test['Prediksi Kluster'] = y\_pred

[122] # Mapping prediksi kluster ke label deskriptif

[123] label\_mapping = {0: 'rendah', 1: 'tinggi', 2: 'sangat tinggi'}

[124] df\_test['Prediksi Kluster'] = df\_test['Prediksi Kluster'].map(label\_mapping)

[125] # Ekspor hasil ke file Excel

[126] with pd.ExcelWriter('Hasil\_Validasi\_Kluster\_Test.xlsx', engine='openpyxl') as writer:

[127] df\_test.to\_excel(writer, sheet\_name='Data Uji dan Prediksi', index=False)

[128] laporan\_df.to\_excel(writer, sheet\_name='Laporan Klasifikasi')

[129] # Menambahkan akurasi sebagai lembar baru

[130] pd.DataFrame({'Akurasi': [akurasi]}).to\_excel(writer, sheet\_name='Akurasi', index=False)

[131] print("Hasil validasi dan data uji telah diekspor ke file Excel 'Hasil\_Validasi\_Kluster\_Test.xlsx'.")