

LAPORAN PENGOLAHAN DATA DAN PENCARIAN DEFINISI

PERTEMUAN 1- 7

(TUGAS UJIAN TENGAH SEMESTER)

TUGAS UTS



OLEH:

ASEP RIDWAN HIDAYAT

231012050036

PROGRAM STUDI MAGISTER TEKNIK INFORMATIKA

PROGRAM PASCA SARJANA

UNIVERSITAS PAMULANG

TANGERANG SELATAN

2024

TUGAS PERTEMUAN KE 1

Penugasan

A. Jelaskan istilah-istilah dalam slide

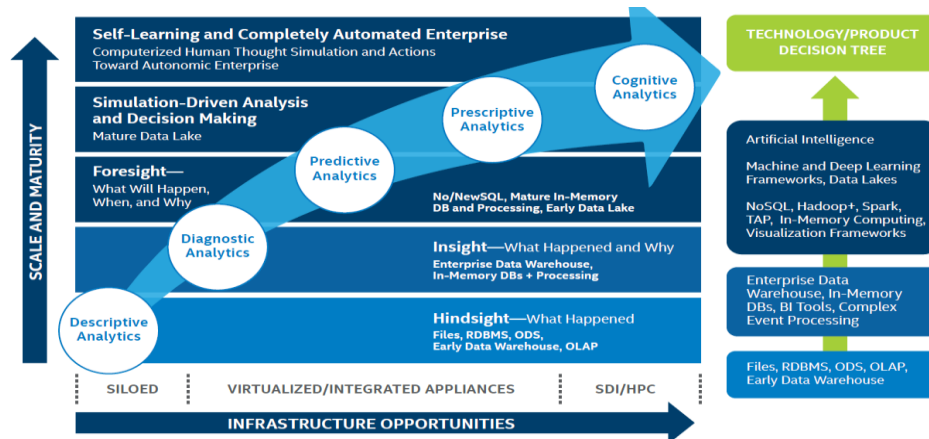


Figure 1. Advanced Analytics Maturity Path: Moving to Real-Time Enterprise
The analytics maturity curve moves from basic analytics that describe what happened in the past to cognitive analytics that automate decisions.

Gambar A.1 Slide ke-12 dari pertemuan 1 data mining

1.1 Diagnostic analytics

Diagnostik Analitik merujuk pada proses menganalisis data untuk memahami kondisi atau situasi saat ini dengan lebih baik.

Diagnostik analitik bertujuan untuk menjawab pertanyaan "mengapa" suatu peristiwa terjadi dengan menganalisis data yang ada.

Beberapa tugas yang terkait dengan diagnostic analytics:

1. Identifikasi Masalah: Mengidentifikasi masalah atau tantangan tertentu yang ingin dipecahkan atau dipahami melalui analisis data.
2. Pengumpulan Data: Mengumpulkan data historis atau saat ini yang relevan untuk memahami konteks masalah atau peristiwa yang sedang dihadapi.
3. Pemrosesan dan Pembersihan Data: Membersihkan dan memproses data untuk memastikan kualitasnya dan menghilangkan anomali atau kesalahan yang dapat memengaruhi analisis.
4. Analisis Korelasi: Menganalisis korelasi antara berbagai variabel atau faktor untuk menentukan hubungan yang mungkin antara mereka.
5. Analisis Root Cause: Mengidentifikasi akar penyebab dari masalah atau peristiwa tertentu dengan menganalisis data untuk menemukan faktor-faktor yang berkontribusi terhadap kejadian tersebut.

6. Pengembangan Solusi: Menggunakan wawasan yang diperoleh dari analisis data untuk merancang solusi atau strategi yang efektif untuk mengatasi masalah atau meningkatkan kinerja di masa mendatang.

1.2 Predictive analytics

Analitik prediktif merujuk pada proses penggunaan data, algoritma statistik, dan teknik machine learning untuk membuat prediksi tentang hasil masa depan atau perilaku berdasarkan pola dan tren yang teridentifikasi dalam data historis.

Tujuan utamanya adalah untuk memprediksi apa yang mungkin terjadi di masa mendatang berdasarkan pemahaman terhadap data yang ada.

Tugas dari Predictive Analytics diantaranya:

1. Pengumpulan Data: Mengumpulkan data historis yang relevan yang akan digunakan untuk membangun model prediktif.
2. Pembersihan dan Pemrosesan Data: Membersihkan dan memproses data untuk menghilangkan anomali, duplikasi, atau nilai yang hilang yang dapat mengganggu analisis yang akurat.
3. Eksplorasi Data: Mengeksplorasi data untuk memahami pola, tren, dan korelasi yang ada dalam data historis.
4. Pemilihan Fitur: Memilih variabel atau fitur yang paling relevan untuk dimasukkan ke dalam model prediktif.
5. Pembuatan Model: Membangun model prediktif menggunakan berbagai teknik seperti regresi, klasifikasi, atau klusterisasi berdasarkan data historis yang ada.
6. Validasi Model: Memvalidasi model prediktif untuk memastikan bahwa model tersebut dapat memberikan prediksi yang akurat dengan menggunakan metode seperti validasi silang atau pemisahan data.
7. Optimisasi Model: Mengoptimalkan parameter model untuk meningkatkan kinerja dan keakuratan prediksi.
8. Evaluasi Model: Mengukur kinerja model prediktif dengan menggunakan metrik evaluasi yang relevan seperti akurasi, presisi, recall, atau area di bawah kurva ROC.
9. Implementasi Model: Mengimplementasikan model prediktif ke dalam lingkungan produksi sehingga dapat digunakan untuk membuat prediksi pada data baru.
10. Pemantauan dan Penyesuaian: Memantau kinerja model secara berkala dan melakukan penyesuaian jika diperlukan berdasarkan perubahan dalam data atau kondisi bisnis.

11. Interpretasi Hasil: Menginterpretasi hasil prediksi dan mengambil tindakan yang sesuai berdasarkan wawasan yang diberikan oleh model prediktif.

1.3 Prespective analytics

Prespective analytics adalah pendekatan dalam data mining yang memungkinkan para analis untuk memahami data dari berbagai sudut pandang atau perspektif. Hal ini dapat mencakup analisis dari berbagai dimensi, termasuk waktu, lokasi, demografi, perilaku pelanggan, dan faktor lainnya yang relevan tergantung pada konteks bisnis atau masalah yang sedang diteliti.

Tujuan utama dari seorang perspective analitic ada pada mensimulasi model dari sebuah data sampai pada mengambil sebuah kesimpulan.

Tugas yang biasanya terkait dengan perspektif analitik dalam data mining.

1. Pengumpulan Data: Memastikan data yang relevan dan berkualitas tersedia untuk analisis. Ini mungkin melibatkan integrasi data dari berbagai sumber yang berbeda.
2. Pembersihan Data: Melakukan pembersihan data untuk menghilangkan anomali, duplikasi, atau nilai yang hilang yang dapat mengganggu analisis yang akurat.
3. Pemrosesan Data: Menyiapkan data untuk analisis dengan melakukan transformasi dan normalisasi yang diperlukan, serta membuat struktur data yang sesuai.
4. Analisis Exploratif: Melakukan analisis awal untuk mengeksplorasi pola, tren, dan korelasi dalam data. Ini bisa melibatkan penggunaan visualisasi data untuk memahami distribusi dan hubungan antar variabel.
5. Analisis Prediktif: Membangun model prediktif yang dapat memprediksi perilaku masa depan atau hasil berdasarkan data historis. Ini melibatkan penggunaan teknik seperti regresi, klasifikasi, atau klasterisasi.
6. Analisis Preskriptif: Membuat rekomendasi atau strategi berdasarkan hasil analisis untuk membimbing keputusan dan tindakan di masa depan. Ini bisa melibatkan penggunaan teknik optimisasi atau simulasi.
7. Evaluasi Model: Mengukur kinerja model analitik untuk memastikan keakuratannya dan mengidentifikasi area di mana model tersebut dapat ditingkatkan.
8. Implementasi Solusi: Mengimplementasikan wawasan yang diperoleh dari analisis data ke dalam strategi bisnis atau proses operasional untuk meningkatkan kinerja dan hasil.

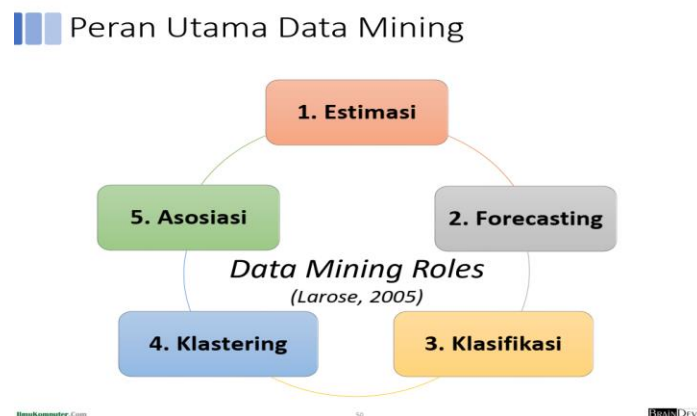
9. Monitoring dan Penyesuaian: Memantau kinerja solusi analitik secara terus-menerus dan melakukan penyesuaian jika diperlukan berdasarkan perubahan dalam data atau kondisi pasar.

1.4 Cognitive Analytics

Cognitive analytics adalah pendekatan analitik yang menggabungkan teknologi kecerdasan buatan (artificial intelligence) dengan pemahaman tentang cara kerja otak manusia, termasuk proses berpikir, belajar, dan pengambilan keputusan. Tujuannya adalah untuk memanfaatkan kemampuan mesin untuk memproses dan menganalisis data besar secara cepat, sementara juga mempertimbangkan konteks psikologis dan perilaku manusia.

Tugas-tugas dalam cognitive analytics mencakup serangkaian kegiatan yang bertujuan untuk memanfaatkan teknologi kecerdasan buatan (AI) dan pemahaman tentang proses kognitif manusia untuk menganalisis data dan menghasilkan wawasan yang bernilai.

B. Jelaskan peran utama data mining dari slide :



Gambar B.1 Slide ke-12 dari pertemuan 1 data mining

1. Estimasi

Estimasi merujuk pada proses memperkirakan atau mengevaluasi nilai yang tidak diketahui dari suatu variabel berdasarkan data yang ada. Ini melibatkan penggunaan teknik statistik atau matematika untuk membuat perkiraan tentang nilai yang mungkin dari suatu variabel berdasarkan data yang diamati. Estimasi sangat penting dalam analisis data karena memungkinkan kita untuk membuat asumsi yang masuk akal tentang populasi atau fenomena yang lebih besar berdasarkan sampel yang tersedia.

2. Forecasting

Forecasting (peramalan) merujuk pada proses menggunakan data historis untuk memprediksi nilai masa depan dari suatu variabel atau serangkaian variabel. Tujuan dari peramalan adalah untuk memproyeksikan pola atau tren yang ada dalam data historis ke masa depan, sehingga organisasi atau individu dapat membuat keputusan yang lebih baik dan merencanakan tindakan yang sesuai.

3. Klasifikasi

Klasifikasi adalah proses mengelompokkan atau memasangkan data ke dalam kategori atau kelas tertentu berdasarkan karakteristik atau atribut yang diberikan. Tujuan dari klasifikasi adalah untuk memprediksi label kelas dari instance-data baru berdasarkan pola yang ditemukan dari data yang telah diamati sebelumnya.

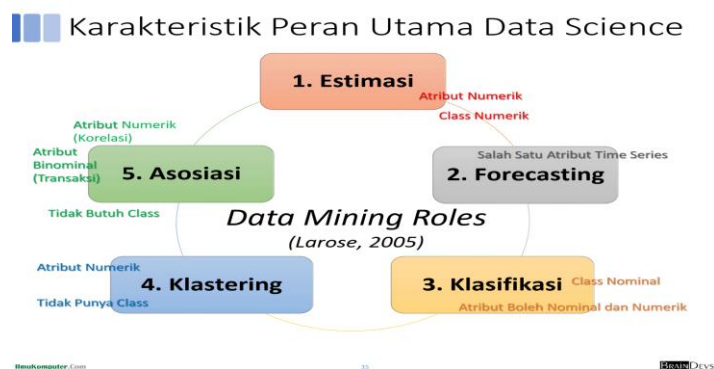
4. Klastering

Klastering (clustering) adalah proses pengelompokan data ke dalam kelompok-kelompok yang homogen berdasarkan kesamaan karakteristiknya. Tujuan dari klastering adalah untuk mengidentifikasi pola atau struktur yang ada dalam data tanpa memerlukan label kelas sebelumnya. Dengan kata lain, klastering membantu dalam mengelompokkan data tanpa adanya informasi tentang kelas-kelas yang mungkin.

5. Asosiasi

Asosiasi (association) merujuk pada proses menemukan hubungan atau korelasi antara item-item dalam dataset. Tujuan utama dari analisis asosiasi adalah untuk mengidentifikasi aturan asosiasi yang menggambarkan hubungan antara item-item yang sering muncul bersama dalam suatu transaksi atau kejadian.

C. Jelaskan karekteristik peran utama dari data science



Gambar C.1 Slide Ke-16 dari pertemuan 1 data mining

1. Atribut Numerik

Atribut numerik adalah jenis atribut atau fitur dalam data yang nilainya berupa angka atau bilangan. Atribut ini dapat memiliki nilai kontinu, diskrit, atau interval, yang artinya mereka dapat mewakili kuantitas yang terus

berubah atau jumlah yang terbatas dalam suatu rentang. Atribut numerik biasanya digunakan untuk menyimpan data yang memiliki skala pengukuran atau nilai yang dapat diukur secara kuantitatif.

Contoh dari atribut numerik

- a) Usia: Atribut ini mewakili usia seseorang dalam bentuk bilangan bulat, misalnya 25 tahun, 42 tahun, atau 60 tahun.
- b) Pendapatan: Atribut ini mewakili pendapatan seseorang dalam bentuk bilangan real atau desimal, misalnya \$35,000, \$50,000, atau \$100,000.
- c) Jumlah Barang: Atribut ini mewakili jumlah barang atau item tertentu dalam suatu transaksi atau stok, dalam bentuk bilangan bulat, misalnya 10 buah, 50 buah, atau 100 buah

2. Class Numerik

Class numerik adalah jenis variabel target dalam suatu dataset yang nilainya berupa bilangan atau angka kontinu. Dalam konteks klasifikasi, variabel target ini biasanya merupakan nilai yang ingin diprediksi atau diklasifikasikan berdasarkan atribut-atribut yang ada.

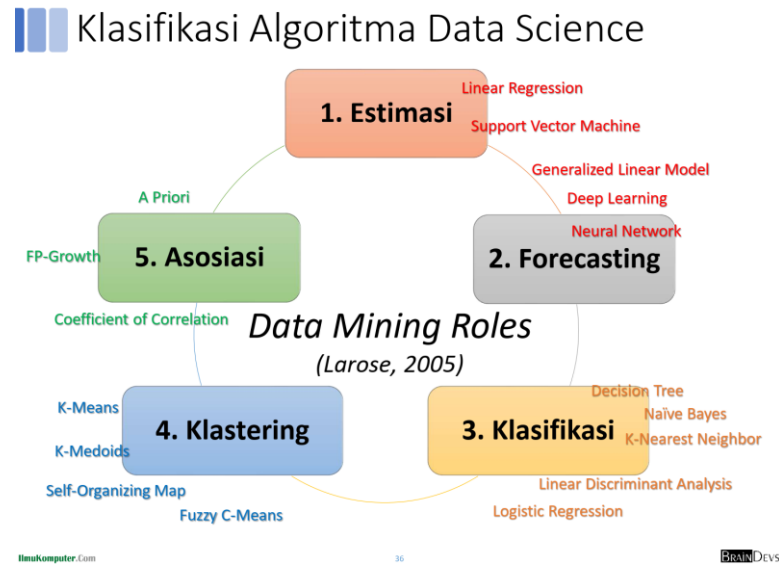
Contoh dari Class numerik , diantaranya :

- a) Prediksi Penjualan: Jika kita ingin memprediksi penjualan suatu produk berdasarkan atribut-atribut seperti harga, promosi, dan musim, maka class numeriknya akan menjadi jumlah penjualan produk tersebut dalam satuan angka kontinu, misalnya 100 unit, 500 unit, atau 1000 unit.
- b) Prediksi Harga Rumah: Jika kita ingin memprediksi harga rumah berdasarkan atribut-atribut seperti luas tanah, jumlah kamar tidur, dan lokasi geografis, maka class numeriknya akan menjadi harga rumah dalam bentuk angka kontinu, misalnya Rp. 100,000, Rp. 200,000, atau Rp. 500,000.

3. Class Nominal

Class Nominal adalah jenis variabel atau atribut dalam suatu dataset yang nilainya merepresentasikan kategori atau label yang tidak memiliki urutan atau tingkatan tertentu. Dalam konteks klasifikasi atau analisis data, class nominal digunakan untuk menunjukkan kelas atau kelompok yang berbeda tanpa adanya hubungan ordinal di antara mereka.

D. Jelaskan Klasifikasi Algoritma Data Science



Gambar C.1 Slide Ke-17 dari pertemuan 1 data mining

Estimasi :

- Linear Regresiion

Regresi linear adalah metode statistik yang digunakan untuk memodelkan hubungan linier antara satu atau lebih variabel independen (prediktor) dan satu variabel dependen (target). Tujuan utama dari regresi linear adalah untuk memahami dan memprediksi bagaimana nilai variabel dependen akan berubah ketika nilai variabel independen berubah.

Dalam regresi linear, hubungan antara variabel independen (x) dan variabel dependen (y) dijelaskan oleh persamaan garis lurus:

$$y = mx + b$$

di mana:

- ✓ y adalah variabel dependen atau target yang ingin diprediksi.
 - ✓ x adalah variabel independen atau prediktor.
 - ✓ m adalah kemiringan (slope) dari garis regresi, yang menunjukkan seberapa besar perubahan dalam
 - ✓ y yang diharapkan ketika x berubah.
 - ✓ b adalah perpotongan (intercept) dari garis regresi, yaitu nilai y ketika $x=0$.
- Support vector machine
Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang digunakan untuk masalah klasifikasi dan regresi. SVM bekerja dengan membangun hyperplane dalam ruang berdimensi tinggi yang memisahkan dua kelas data. Hyperplane ini dipilih sedemikian rupa sehingga jarak antara

hyperplane dan titik terdekat dari setiap kelas (yang disebut support vector) adalah maksimum.

- Generalized Linear modeling

Generalized Linear Model (GLM) adalah kerangka kerja statistik yang digunakan untuk memodelkan hubungan antara variabel dependen (response variable) dan satu atau lebih variabel independen (predictor variable) dalam berbagai situasi, termasuk situasi di mana distribusi kesalahan tidak memenuhi asumsi normalitas dalam regresi linear klasik.

- Deep learning

Deep learning adalah cabang dari pembelajaran mesin (machine learning) yang menggunakan arsitektur jaringan saraf tiruan (neural networks) yang sangat dalam (deep) untuk memodelkan dan mengekstraksi pola dari data yang kompleks. Deep learning memiliki kemampuan untuk belajar secara otomatis dari data yang tidak terstruktur atau tidak terlabel, tanpa memerlukan fitur atau pemrosesan manual yang rumit.

Deep learning telah menjadi sangat populer karena keberhasilannya dalam berbagai tugas pembelajaran mesin yang menantang, termasuk pengenalan gambar, pengenalan suara, penerjemahan bahasa, pengenalan wajah, dan masih banyak lagi, Contoh-contoh aplikasi deep learning termasuk:

- ✓ Pengenalan Gambar: Membedakan objek dan kelas dalam gambar, seperti klasifikasi gambar atau deteksi objek.
- ✓ Penerjemahan Bahasa: Menerjemahkan teks dari satu bahasa ke bahasa lain.
- ✓ Pengenalan Suara: Mengenali dan mentranskripsikan ucapan manusia menjadi teks tertulis.
- ✓ Otomatisasi Proses: Memodelkan dan memprediksi perilaku sistem kompleks, seperti kendaraan otonom dan sistem rekomendasi.
- ✓ Analisis Data: Menganalisis data besar untuk menemukan pola dan tren yang tidak terlihat oleh manusia.

Deep learning menggunakan teknik-teknik seperti backpropagation (propagasi balik), stochastic gradient descent (turunan gradien stokastik), dan optimasi berbasis gradien untuk melatih jaringan saraf tiruan dengan data. Keberhasilan deep learning sering kali bergantung pada memiliki jumlah data yang besar dan komputasi yang kuat, sehingga kemajuan dalam teknologi komputer dan ketersediaan data telah mendukung perkembangan pesat dalam bidang ini.

- Neural network

Jaringan saraf tiruan (neural network) adalah model komputasi yang terinspirasi dari struktur dan fungsi jaringan saraf biologis dalam otak manusia. Jaringan saraf tiruan terdiri dari serangkaian unit pemrosesan (neuron) yang saling

terhubung, membentuk suatu arsitektur yang memungkinkan untuk melakukan pemodelan matematika yang rumit dan pembelajaran dari data.

Jaringan saraf tiruan dapat memiliki berbagai arsitektur, termasuk:

- ✓ Feedforward Neural Network (FFNN): Jaringan ini memiliki aliran data yang satu arah, dari input ke output, tanpa siklus atau keterhubungan mundur. Ini termasuk jaringan saraf tiruan multilayer perceptron (MLP), yang terdiri dari lapisan input, lapisan tersembunyi (hidden layers), dan lapisan output.
- ✓ Recurrent Neural Network (RNN): Jaringan ini memiliki hubungan siklik antar-neuron, yang memungkinkan informasi untuk mengalir mundur dalam jaringan. RNN sering digunakan dalam tugas-tugas yang melibatkan data urutan, seperti pemrosesan bahasa alami dan pemodelan waktunya.
- ✓ Convolutional Neural Network (CNN): Jaringan ini memiliki arsitektur khusus yang berfokus pada pemrosesan data berbentuk grid, seperti gambar. CNN menggunakan lapisan konvolusi untuk mengekstraksi fitur-fitur spasial dari data.

Forecasting

Klasifikasi

- Decision tree

Decision tree (pohon keputusan) adalah model prediktif dalam analisis data yang menggambarkan struktur pilihan keputusan dan hasil yang mungkin. Model ini mengambil bentuk struktur pohon, di mana setiap simpul (node) dalam pohon mewakili keputusan atau pengujian atas suatu atribut, setiap cabang (branch) mewakili hasil dari keputusan tersebut, dan setiap daun (leaf) mewakili hasil atau label klasifikasi.

Proses pembuatan keputusan dalam decision tree dimulai dari simpul akar (root node), di mana atribut yang paling penting untuk pemisahan data ditempatkan. Data kemudian dipecah ke dalam subset berdasarkan nilai atribut tersebut.

Keuntungan dari decision tree meliputi kemampuan untuk dengan mudah diinterpretasikan, memahami hubungan antara variabel, dan dapat menangani data yang terstruktur dan tidak terstruktur. Decision tree juga dapat digunakan untuk klasifikasi dan regresi, tergantung pada tipe variabel target (kategori atau numerik).

Beberapa algoritma populer untuk membangun decision tree meliputi:

- ✓ ID3 (Iterative Dichotomiser 3): Algoritma ini menggabungkan konsep entropi dan gain informasi untuk memilih atribut terbaik untuk memisahkan data.
- ✓ C4.5 (Successor of ID3): Versi yang diperbarui dari ID3, C4.5 juga menggunakan gain informasi tetapi memperkenalkan strategi untuk menangani nilai yang hilang dan atribut yang kontinu.
- ✓ CART (Classification and Regression Trees): Algoritma ini dapat digunakan untuk klasifikasi dan regresi. Ini menggunakan kriteria pemisahan yang berbeda, seperti impurity (ketidakhomogenan) Gini untuk klasifikasi dan penurunan MSE (Mean Squared Error) untuk regresi.
- Naïve bayes

Naive Bayes adalah salah satu algoritma klasifikasi yang populer dalam pembelajaran mesin (machine learning). Algoritma ini didasarkan pada teorema Bayes dengan asumsi bahwa setiap fitur dalam data adalah independen satu sama lainnya. Meskipun asumsi ini sering kali tidak benar dalam dunia nyata, namun Naive Bayes tetap memberikan hasil yang baik dalam banyak kasus.
- k-Nearest Neighbors (K-NN)

K-Nearest Neighbors (K-NN) adalah salah satu algoritma klasifikasi yang sederhana dan populer dalam pembelajaran mesin. Algoritma ini digunakan untuk memprediksi kelas dari sebuah sampel data dengan cara mencari k-neighbors terdekat dari data tersebut dalam ruang fitur, lalu menggunakan mayoritas kelas dari tetangga terdekat tersebut untuk memprediksi kelas sampel data yang baru.

Proses klasifikasi dengan algoritma K-NN melibatkan langkah-langkah berikut:

 - ✓ Pemilihan Nilai K: Tentukan nilai k, yaitu jumlah tetangga terdekat yang akan digunakan untuk memprediksi kelas sampel data yang baru.
 - ✓ Pencarian Tetangga Terdekat: Hitung jarak antara sampel data yang baru dengan setiap sampel data dalam dataset. Jarak ini bisa dihitung menggunakan berbagai metrik, seperti jarak Euclidean, jarak Manhattan, atau jarak Minkowski. Kemudian, identifikasi k sampel data dengan jarak terdekat dengan sampel data yang baru.
 - ✓ Voting Mayoritas: Tentukan kelas mayoritas dari k tetangga terdekat tersebut. Karena kelas tersebut mungkin bukan unik, maka bisa digunakan metode pemilihan, seperti voting mayoritas atau weighted voting, untuk menentukan kelas yang paling mungkin.
 - ✓ Prediksi Kelas: Gunakan kelas mayoritas yang telah diidentifikasi untuk memprediksi kelas sampel data yang baru.

Algoritma K-NN tidak memerlukan proses pelatihan (training) yang kompleks, karena hanya menyimpan data latih dalam memori. Namun, algoritma ini memerlukan komputasi yang tinggi saat melakukan prediksi pada data yang

besar atau dalam dimensi yang tinggi, karena memerlukan perhitungan jarak dengan setiap titik dalam dataset.

- **Linear Discriminant Analysis**

Linear Discriminant Analysis (LDA) adalah metode statistik yang digunakan untuk mengklasifikasikan data dengan mengidentifikasi hubungan linear antara variabel independen (fitur) dan variabel dependen (kelas atau label kategori). LDA juga dapat digunakan untuk reduksi dimensi, yang merupakan teknik untuk mengurangi jumlah variabel independen dalam dataset sambil mempertahankan sebagian besar informasi.

Tujuan utama dari Linear Discriminant Analysis adalah untuk memaksimalkan jarak antara pusat-pusat kelas (mean) dan meminimalkan dispersi dalam setiap kelas, sehingga memungkinkan untuk membuat batas keputusan (decision boundary) yang optimal antara kelas-kelas.

- **Logistic Regression**

Logistic Regression adalah salah satu teknik dalam statistik yang digunakan untuk melakukan analisis regresi pada data biner atau kategori. Meskipun namanya menyiratkan "regresi", logistic regression sebenarnya lebih sering digunakan untuk masalah klasifikasi, khususnya dalam kasus di mana variabel dependen (target) adalah variabel biner, yaitu memiliki dua kemungkinan kelas atau label.

Keuntungan dari logistic regression adalah kemampuannya untuk mengatasi masalah klasifikasi biner dengan baik, kemudahan interpretasi hasil, serta tidak adanya asumsi tentang distribusi normalitas dari variabel independen. Logistic regression juga cukup fleksibel dan dapat diperluas untuk menangani data yang lebih kompleks dengan menggunakan teknik seperti regularisasi dan polinomial logistic regression.

Klastering

- **K-means**

K-Means adalah salah satu algoritma pengelompokan atau klastering yang paling umum digunakan dalam analisis data. Tujuannya adalah untuk membagi dataset menjadi beberapa kelompok (klaster) berdasarkan pola atau kesamaan diantara observasi.

Algoritma K-Means berusaha untuk meminimalkan jumlah kesalahan penugasan klaster, yang diukur dengan jarak antara observasi dan pusat klasternya. Algoritma ini berhenti ketika tidak ada perubahan dalam penugasan klaster atau ketika kriteria konvergensi telah tercapai.

K-Means memiliki beberapa keunggulan, termasuk efisiensi komputasi yang tinggi dan mudah diimplementasikan. Namun, algoritma ini juga memiliki beberapa kelemahan, seperti sensitif terhadap inisialisasi awal pusat klaster, rentan terhadap konvergensi ke minimum lokal, dan tidak cocok untuk data yang memiliki klaster dengan bentuk atau ukuran yang berbeda.

- K-Medoids

K-Medoids adalah varian dari algoritma klustering K-Means yang sering digunakan untuk membagi dataset menjadi beberapa kelompok (klaster) berdasarkan kesamaan antara observasi. Salah satu perbedaan utama antara K-Medoids dan K-Means adalah bahwa K-Medoids menggunakan titik-titik data aktual dalam dataset sebagai medoids (representasi pusat) dari setiap klaster, sementara K-Means menggunakan rata-rata dari titik-titik data.

K-Medoids memiliki beberapa keunggulan dibandingkan dengan K-Means, terutama dalam hal kestabilan terhadap inisialisasi medoids awal dan ketangguhan terhadap outlier. Selain itu, karena medoids adalah titik-titik aktual dalam dataset, hasil klustering dari K-Medoids lebih mudah diinterpretasikan daripada K-Means.

K-Medoids sering digunakan dalam berbagai aplikasi, seperti segmentasi pelanggan, pengelompokan geografis, dan analisis biologis. Ini adalah salah satu algoritma klustering yang berguna dalam analisis data. Namun, perlu dicatat bahwa K-Medoids juga memiliki beberapa kelemahan, seperti sensitivitas terhadap jumlah klaster awal yang dipilih dan kinerja yang buruk pada dataset besar.

- Self Organizing Map

Self-Organizing Map (SOM), juga dikenal sebagai Kohonen map, adalah salah satu jenis dari jaringan saraf tiruan (neural network) yang dikembangkan oleh Teuvo Kohonen pada tahun 1980-an. SOM adalah algoritma pembelajaran tanpa pengawasan yang digunakan untuk pemetaan dan visualisasi data multidimensi ke dalam representasi dua dimensi, sehingga memudahkan interpretasi dan analisis.

Prinsip dasar dari SOM adalah untuk mengorganisasi data ke dalam struktur topologi dua dimensi, seperti grid atau lattice, di mana setiap unit atau neuron dalam grid merepresentasikan suatu kelas atau konsep. Selama proses pelatihan, SOM secara iteratif memperbarui bobot setiap neuron untuk mencocokkan pola input yang diberikan. Neuron yang berdekatan dalam grid memiliki bobot yang serupa, sehingga membentuk klaster atau kelompok yang saling berhubungan.

SOM memiliki berbagai aplikasi dalam analisis data dan pemetaan, termasuk dalam bidang pengenalan pola, segmentasi pelanggan, analisis citra, analisis genomik, dan pemrosesan bahasa alami. SOM membantu mengurangi dimensi data kompleks ke dalam representasi yang lebih sederhana dan mudah dimengerti, sehingga memudahkan analisis dan interpretasi.

- Fuzzy C-Mean

Fuzzy C-Means (FCM) adalah salah satu algoritma klustering yang digunakan untuk mengelompokkan data ke dalam beberapa kelompok (klaster) berdasarkan kesamaan antara observasi. Namun, yang membedakan FCM dari algoritma

klustering lainnya, seperti K-Means, adalah bahwa FCM mengizinkan observasi untuk termasuk dalam beberapa kluster secara parsial, bukan secara eksklusif ke satu kluster saja. Pendekatan ini memungkinkan FCM untuk menangani tingkat ketidakpastian atau keambiguitasan dalam data.

Saat menggunakan FCM, setiap observasi diberi derajat keanggotaan untuk setiap kluster, yang merupakan bilangan pecahan antara 0 dan 1, yang menunjukkan seberapa kuat observasi tersebut terhubung dengan kluster tersebut. Observasi yang memiliki derajat keanggotaan yang tinggi terhadap satu kluster menunjukkan bahwa observasi tersebut cocok dengan kluster tersebut.

Asosiasi

- A Priori

A priori adalah istilah yang sering digunakan dalam konteks statistik, probabilitas, dan pemodelan data. Istilah ini berasal dari bahasa Latin yang secara harfiah berarti "sebelumnya". Dalam konteks statistik dan probabilitas, "a priori" mengacu pada pengetahuan, keyakinan, atau asumsi yang ada sebelum melihat atau menganalisis data. Ini adalah pengetahuan yang diperoleh sebelum pengamatan atau pengujian dilakukan.

Dalam analisis data dan pemodelan statistik, konsep a priori sering kali muncul dalam beberapa konteks:

- ✓ A priori probability: Probabilitas yang ditetapkan atau diprediksi sebelum pengumpulan data atau pengujian. Ini adalah probabilitas yang didasarkan pada pengetahuan atau asumsi sebelumnya, tanpa mempertimbangkan data yang sebenarnya.
- ✓ A priori distribution: Distribusi probabilitas yang dianggap sebelum melihat data. Ini adalah distribusi yang dijadikan dasar untuk memperkirakan parameter model atau probabilitas sebelum data dianalisis.
- ✓ A priori assumption: Asumsi yang dibuat sebelum melakukan analisis data atau pembuatan model. Asumsi ini bisa berupa keyakinan tentang struktur data, hubungan antar variabel, atau distribusi dari variabel acak.

Konsep a priori penting dalam statistik Bayes karena dalam pendekatan Bayes, pengetahuan atau asumsi sebelumnya (a priori) digabungkan dengan data yang diamati (a posteriori) untuk memperbarui estimasi atau keputusan statistik. Dengan kata lain, a priori digunakan sebagai prior distribution dalam proses pembaruan posterior distribution berdasarkan data yang diamati.

- FP-Growth

FP-Growth (Frequent Pattern Growth) adalah salah satu algoritma untuk menemukan pola sering (frequent patterns) dalam kumpulan data transaksional atau dataset yang mengandung itemset besar. Algoritma ini dikembangkan untuk mengatasi kelemahan algoritma Apriori yang memerlukan pencarian berulang-ulang dan membutuhkan banyak ruang memori untuk menyimpan itemset yang besar.

FP-Growth memiliki beberapa keunggulan dibandingkan algoritma Apriori:

- ✓ FP-Growth ini hanya memerlukan dua kali passing pada kumpulan data, sehingga lebih efisien, terutama untuk kumpulan data besar.
- ✓ FP-Growth menggunakan struktur data kompak (FP-tree) untuk menyimpan pola yang sering terjadi, sehingga mengurangi penggunaan memori.
- ✓ FP-Growth bisa jauh lebih cepat daripada Apriori untuk menambang pola yang sering terjadi, terutama ketika kumpulan data berisi banyak transaksi dan item.

FP-Growth banyak digunakan dalam tugas penambangan data seperti analisis keranjang pasar, sistem rekomendasi, dan penambangan aturan asosiasi, di mana menemukan kumpulan item dan pola yang sering digunakan sangat penting untuk memahami hubungan mendasar dalam data.

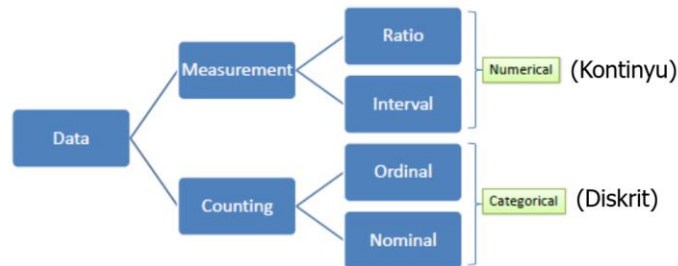
- Coeficien of Coleration

Coefficient of Correlation (Koefisien Korelasi) adalah suatu metrik statistik yang digunakan untuk mengukur kekuatan dan arah hubungan antara dua variabel. Koefisien korelasi mengindikasikan sejauh mana perubahan dalam satu variabel dapat diprediksi oleh perubahan dalam variabel lainnya. Koefisien korelasi berkisar dari -1 hingga 1, di mana:

- ✓ Koefisien korelasi +1 menunjukkan hubungan linier sempurna dan positif antara dua variabel, yang berarti bahwa ketika satu variabel meningkat, variabel lainnya juga cenderung meningkat secara proporsional.
- ✓ Koefisien korelasi -1 menunjukkan hubungan linier sempurna dan negatif antara dua variabel, yang berarti bahwa ketika satu variabel meningkat, variabel lainnya cenderung menurun secara proporsional.
- ✓ Koefisien korelasi 0 menunjukkan bahwa tidak ada hubungan linier antara dua variabel.

Jelaskan Tipe Data

Tipe Data



InfoKomputer.Com

32

DonalDeyS

Gambar C.1 Slide Ke-18 dari pertemuan 1 data mining

1. **Data Nominal:** Data nominal adalah tipe data yang hanya menunjukkan kategori atau label tanpa memiliki urutan atau tingkatan yang intrinsik. Contoh data nominal termasuk jenis kelamin (pria/wanita), warna (merah/biru/hijau), atau jenis kendaraan (mobil/motor/sepeda).
2. **Data Ordinal:** Data ordinal adalah tipe data yang memiliki urutan atau tingkatan, tetapi perbedaan antara nilai-nilai tidak dapat diukur secara kuantitatif dengan presisi. Contoh data ordinal termasuk tingkat kepuasan (sangat puas/puas/netral/tidak puas/sangat tidak puas) atau kelas sosial (tinggi/sedang/rendah).
3. **Data Interval:** Data interval adalah tipe data yang memiliki urutan, dan perbedaan antara nilai-nilai dapat diukur secara kuantitatif dengan presisi, tetapi tidak ada nol mutlak yang bermakna. Contoh data interval termasuk suhu dalam skala Celsius atau Fahrenheit.
4. **Data Rasio:** Data rasio adalah tipe data yang memiliki urutan, perbedaan antara nilai-nilai dapat diukur secara kuantitatif dengan presisi, dan terdapat nol mutlak yang bermakna. Contoh data rasio termasuk berat badan, tinggi badan, atau pendapatan.

TUGAS PADA PERTEMUAN KE-2

MATERI : REGRESI LINIER SEDERHANA

Catatan Tugas pertemuan 2

1. Buat Model untuk Regresi Linear sederhana
2. Buat Model untuk Regresi Linear berganda
3. Diskusikan dalam Forum
4. Tuliskan dalam laporan (dikumpulkan saat UTS)

1. Model Regresi Linear Sederhana

1.1. Deskripsi Masalah:

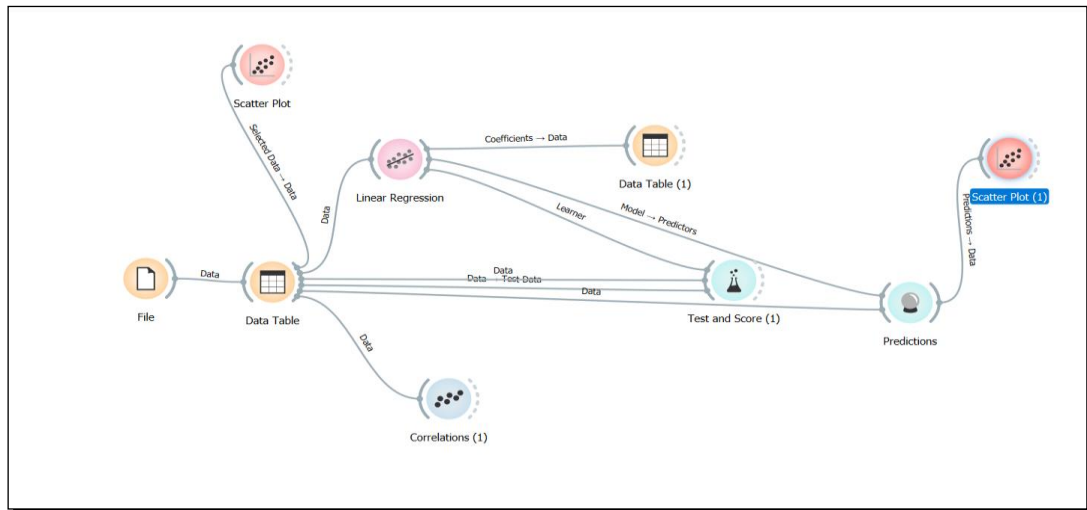
Data berat badan karyawan yang diprediksi dipengaruhi oleh konsumsi jumlah kalori/hari.

Nama Karyawan	Kalori/Hari	Berat Badan
Dian	530	89
Echa	300	48
Winda	358	56
Kelo	510	72
Intan	302	54
Putu	300	42
Aditya	387	60
Anita	527	85
Sefia	415	63
Rosa	512	74
Dani	362	75
Ridwan	325	70
Wahyu	425	65

Variabel :

- X (variable bebas/predictor) = jumlah kalori/hari
- Y (variable tak bebas/response) = berat badan

1.2. METODE PENGOLAHAN DATA



Didapat output

Data Table (1) - Orange

Info
2 instances (no missing data)
1 feature
No target variable.
1 meta attribute

Variables
☒ Show variable labels (if present)
☐ Visualize numeric values
☒ Color by instance classes

Selection
☒ Select full rows

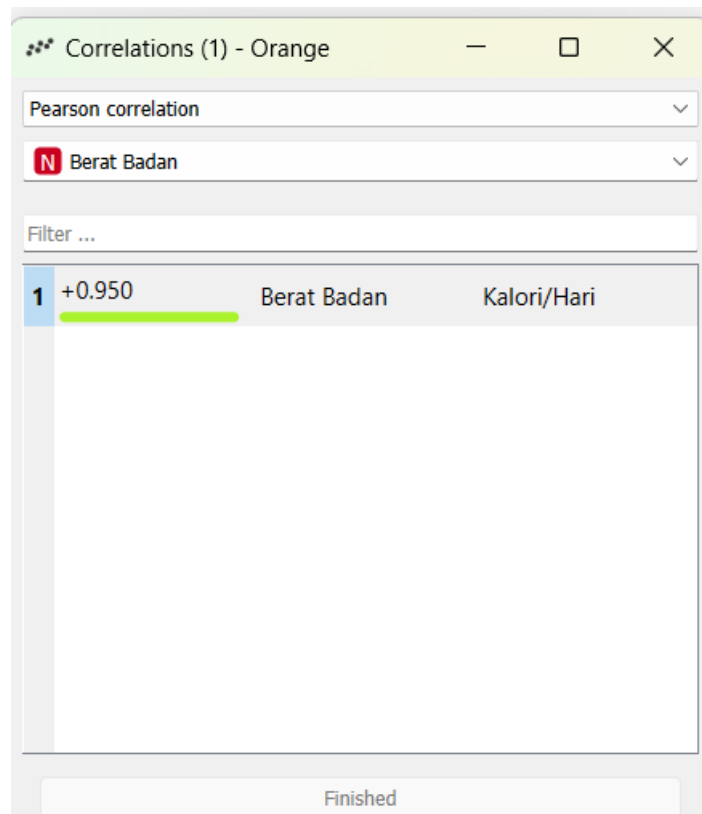
Restore Original Order

☒ Send Automatically

	name	coef
1	intercept	2.60804
2	Kalori/Hari	0.148978

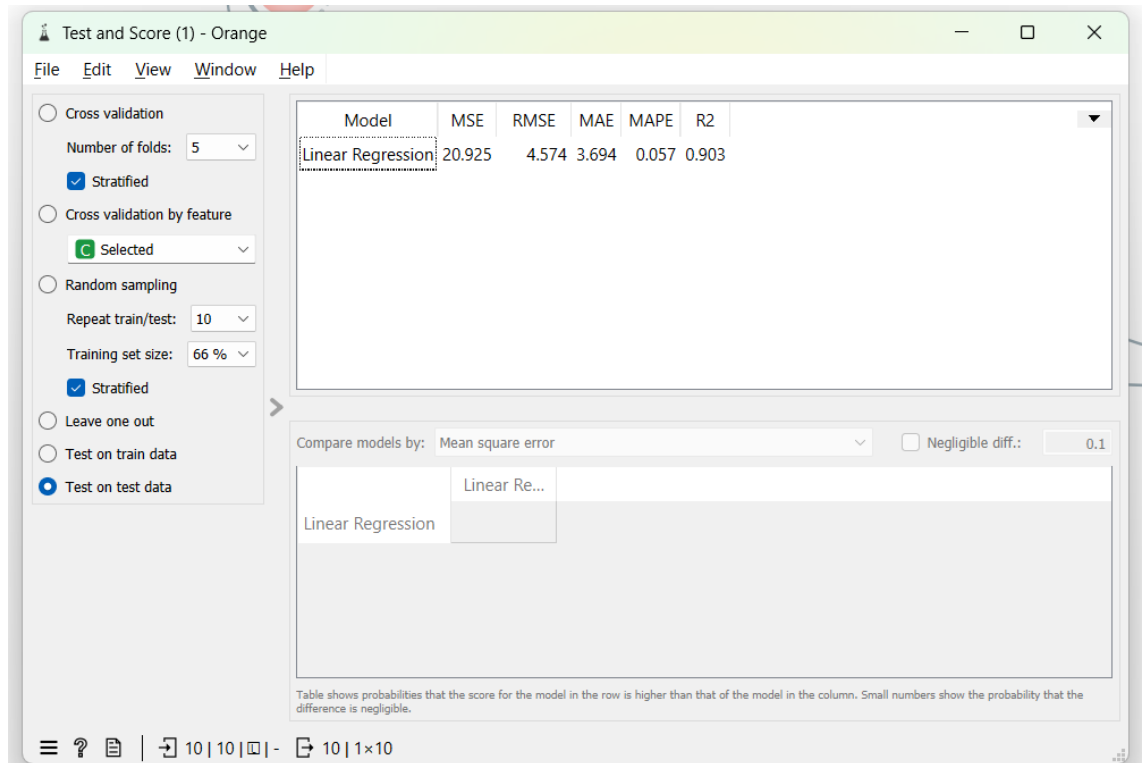
Dari gambar diatas didapatkan persamaan linear:

$$Y = 2.60 + 0.148X_1$$



1. Correlation

Dari capture diatas didapatkan nilai korelasi 0.95 koefisien korelasinya **Sangat Tinggi**, artinya berat badan memang sangat dipengaruhi oleh konsumsi jumlah kalori/hari.



Nilai Koefisien Determinasi (r^2), MRE

Didapat nilai Determinasi (R^2) 0.903 Nilai ini berarti bahwa, 90% variabel bebas/ predictor X dapat menerangkan/ menjelaskan variabel tak bebas/ response Y dan 10% dijelaskan oleh variabel lainnya.

2. Uji Signifikansi dan Hipotesis

Uji-t

a. Menentukan Hipotesis

H_0 = variabel X tidak berpengaruh signifikan/nyata terhadap Y
(Konsumsi jumlah kalori tidak berpengaruh terhadap berat badan karyawan)

H_1 = (Variabel X berpengaruh signifikan/nyata terhadap Y)

Konsumsi jumlah kalori berpengaruh terhadap berat badan karyawan

b. Menentukan tingkat signifikansi

(α) Tingkat signifikansi, α yang sering digunakan adalah $\alpha = 5\%$ ($\alpha = 0,05$)

c. menghitung nilai t hitung menggunakan rumus

$$t_{hit} = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}}$$

d. menghitung nilai tabel t

diketahui Koefisien Determinasi (r^2) = 0,90 dengan Koefisien Korelasi (r) = 0,95 , Jumlah data $n = 13$

dari rumus diatas :

$$t_{hit} = \frac{0.95\sqrt{13-2}}{\sqrt{1-0.90}} = 9.968$$

Derajat kebebasan, $df = n - k = 13 - 2 = 11$,

Dengan menggunakan tabel Uji - t untuk taraf signifikan $\alpha = 5\% = 0,05$ dan $df = 11$, maka diperoleh nilai t pada table, yaitu : $t_{tab} = 3,106$

e. Menentukan daerah penolakan

Menentukan daerah penolakan H_0 (daerah kritis) Bentuk pengujian dua arah, sehingga menggunakan uji-t dua arah :

H_0 akan ditolak jika $t_{hit} > t_{tab}$ atau $-(t_{hit}) < -(t_{tab})$, berarti H_1 diterima.

H_0 akan diterima jika $-(t_{hit}) < t_{tab} < t_{hit}$, berarti H_1 ditolak.

Dari nilai diatas didapatkan $t_{tab} = 3,106$ dengan $t_{hit} = 9.968$, sehinnnga didapat $t_{hit} > t_{tab}$ (H_0 ditolak) berarti bisa ditarik kesimpulan:

ada pengaruh nyata protein (variable X) terhadap berat badan dipertemuan pertama

2. Model Regresi Linear Berganda

2.1. Deskripsi Masalah:

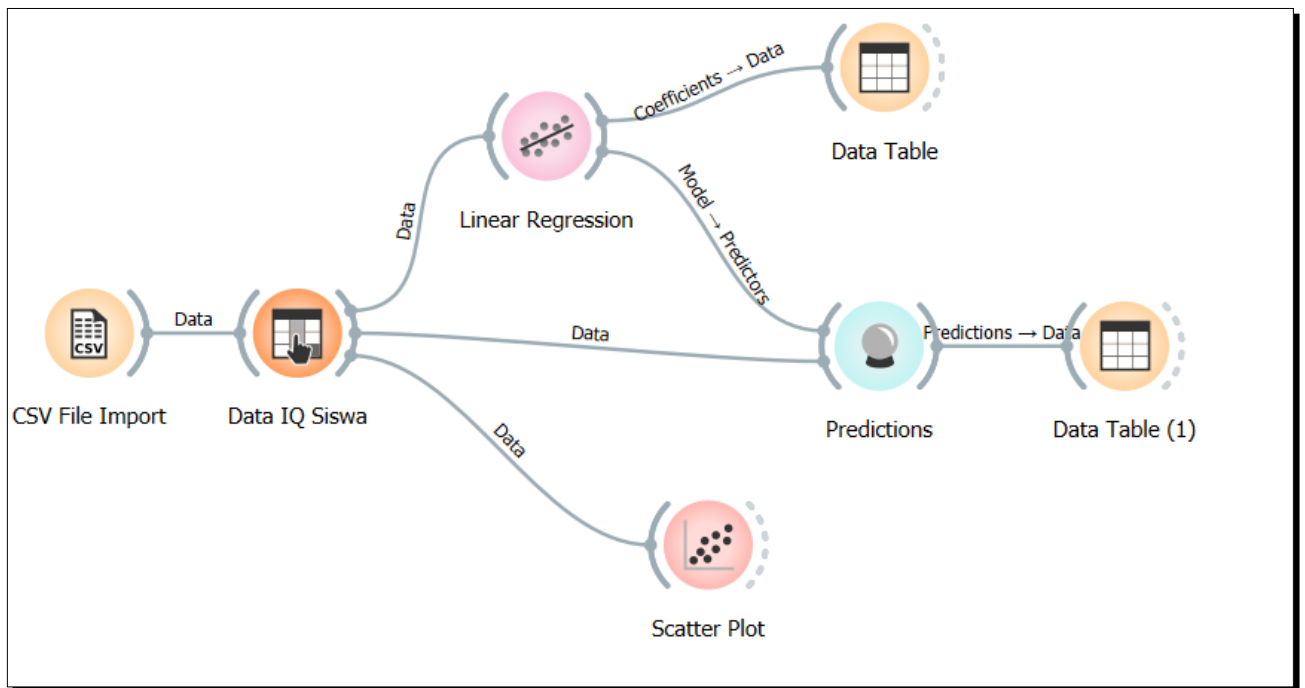
Diberikan data tentang IQ dan tingkat kehadiran sepuluh siswa di kelas yang diperkirakan mempengaruhi nilai UAS

Siswa	Tingkat kehadiran (%) (X_1)	IQ (X_2)	Nilai UAS (Y)
1	60	110	65
2	70	120	70
3	75	115	75
4	80	130	75
5	80	110	80
6	90	120	80
7	95	120	85
8	95	125	95
9	100	110	90
10	100	120	98

Variabel :

- X_1 (variable bebas/predictor) = Tingkat Kehadiran
- X_2 (variable bebas/predictor) = IQ
- Y (variable tak bebas/response) = Nilai UAS

2.2. Metode Pengolahan Data



Gambar 1.3 Pengolahan data menggunakan Orange Data Mining

Menggunakan software orange data mining, Output Screenshot dibawah ini:

The screenshot shows the 'Data Table - Orange' window. On the left, the 'Info' panel indicates 3 instances, 1 feature, and 1 meta attribute. The 'Variables' panel has 'Show variable labels (if present)' checked. The 'Selection' panel has 'Select full rows' checked. A 'Restore Original Order' button and a 'Send Automatically' checkbox are also visible. The main table displays the following data:

	name	coef
1	intercept	23.0545
2	IQ	-0.0343275
3	Tingkat kehadir...	0.737233

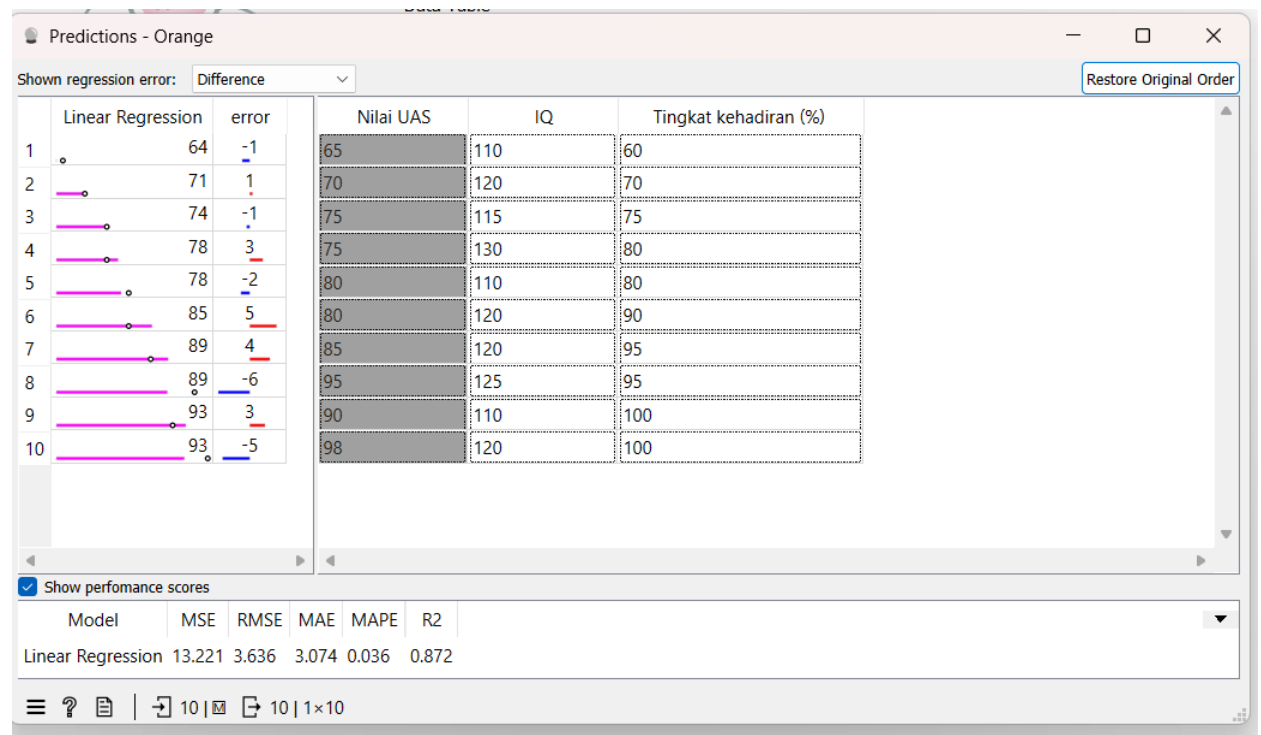
Gambar 2.1 Output dari Regresi Linear Berganda

Dari ouput diatas didapat persamaan regresi :

$$Y = 23.054 + 0.737X_1 - 0.00343X_2$$

Dilihat dari persamaan regresi, nilai X_1 lebih besar dibandingkan dengan nilai X_2 .

Dalam hal ini dapat disimpulkan bahwa presentase kehadiran dikelas lebih berpengaruh daripada IQ.



Dari output diatas Koefisien Determinasi : r2 didapat dari output 0.872, Nilai akhir (Y) yang dapat dijelaskan oleh tingkat kehadiran (X1) dan IQ (X2) pada persamaan regresi $Y = 23.054 + 0.737X_1 - 0.00343X_2$ adalah 87%, Sisanya, sebesar 13% dijelaskan oleh faktor lain diluar variable-variabel pada persamaan $Y = 23.054 + 0.737X_1 - 0.00343X_2$

TUGAS PADA PERTEMUAN KE-3 MATERI : REGRESI LOGISTIK

Catatan Tugas pertemuan 3

1. Buat Model untuk Regresi Logistik
2. Metrik Pengukuran Regresi Logistik + Confusion Matrix (Uraikan)
3. Kelebihan dan Kekurangan Regresi Logistik
4. Cari 10 Jurnal terkait pemanfaatan Regresi Logistik
5. Diskusikan dalam Forum
6. Tuliskan dalam laporan (dikumpulkan saat UTS)

1. Model Regresi Logistik

1.1 Data yang akan diolah dengan deskripsi sebagai berikut:

Seorang peneliti ingin mengetahui bagaimana pengaruh kualitas pelayanan publik terhadap kepuasan pengguna (masyarakat). Kualitas pelayanan publik diteliti melalui variabel Daya Tanggap (X1) dan Empati (X2). Kepuasan penggunaan layanan (Y) sebagai variabel dependent adalah variabel dummy dimana dimana jika responden menjawab puas maka kita beri skor 1 dan jika menjawab tidak puas kita beri skor 0.

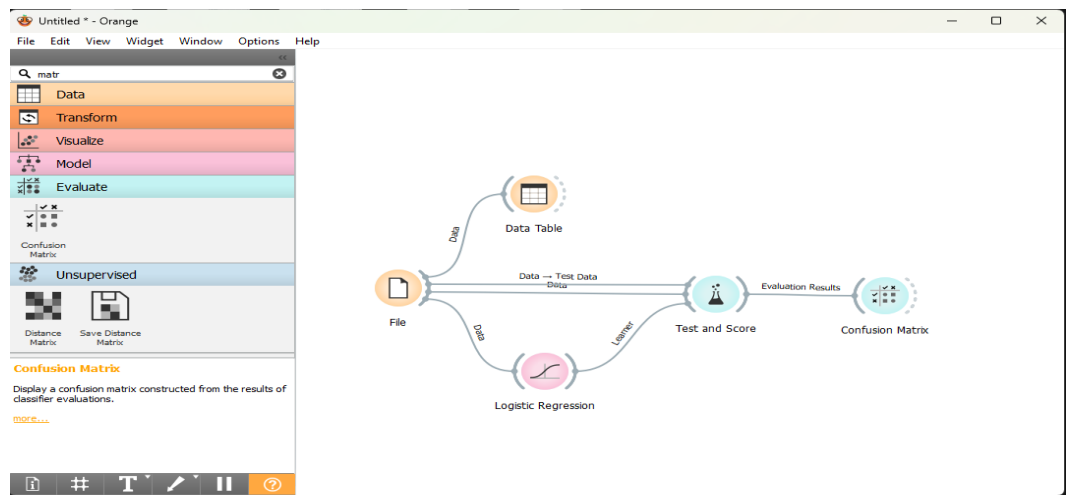
Person	Daya Tanggap (X1)	Empati (X2)	Kepuasan Pelanggan (Y)
1	36	45	1
2	34	39	0
3	30	38	0
4	32	38	1
5	36	45	1
6	33	42	0
7	36	45	1
8	36	45	1
9	31	36	0
10	31	37	0
11	36	45	1
12	33	41	0
13	32	40	0

Person	Daya Tanggap (X1)	Empati (X2)	Kepuasan Pelanggan (Y)
26	33	41	0
27	32	39	0
28	30	36	0
29	30	36	0
30	36	42	1
31	33	38	0
32	33	38	0
33	35	41	1
34	35	41	1
35	34	40	1
36	30	38	0
37	30	40	1
38	35	41	1

14	33	39	0	39	34	42	1
15	34	42	1	40	33	40	0
16	34	42	0	41	34	43	0
17	32	39	0	42	30	38	0
18	34	42	1	43	34	42	1
19	33	40	0	44	30	41	1
20	34	43	0	45	34	40	0
21	32	39	0	46	34	42	1
22	36	44	1	47	34	38	0
23	33	37	0	48	34	44	1
24	30	38	0	49	35	43	0
25	36	43	1	50	34	42	1

Tabel 1.1 Data Daya Tanggap dan Empati Pelayanan Publik

Berikut proses pengolahan data yang dilakukan menggunakan orange data mining.



Gambar 1.1 Pengolahan data menggunakan orangedata mining

Data Table - Orange

Info
50 instances (no missing data)
3 features
Target with 2 values
No meta attributes.

Variables
☒ Show variable labels (if present)
☐ Visualize numeric values
☒ Color by instance classes

Selection
☒ Select full rows

Restore Original Order

☒ Send Automatically

	Person	Kepuasan Pelanggan (Y)	Daya Tanggap (X1)	Empati (X2)
1	1	1	36	45
2	2	0	34	39
3	3	0	30	38
4	4	1	32	38
5	5	1	36	45
6	6	0	33	42
7	7	1	36	45
8	8	1	36	45
9	9	0	31	36
10	10	0	31	37
11	11	1	36	45
12	12	0	33	41
13	13	0	32	40
14	14	0	33	39
15	15	1	34	42
16	16	0	34	42
17	17	0	32	39
18	18	1	34	42
19	19	0	33	40
20	20	0	34	43
21	21	0	32	39

Gambar 1.2 Visualisasi Data Dengan Orange Data Mining

Data Table - Orange

Info
4 instances (no missing data)
1 feature
No target variable.
1 meta attribute

Variables
☒ Show variable labels (if present)
☐ Visualize numeric values
☒ Color by instance classes

Selection
☒ Select full rows

Restore Original Order

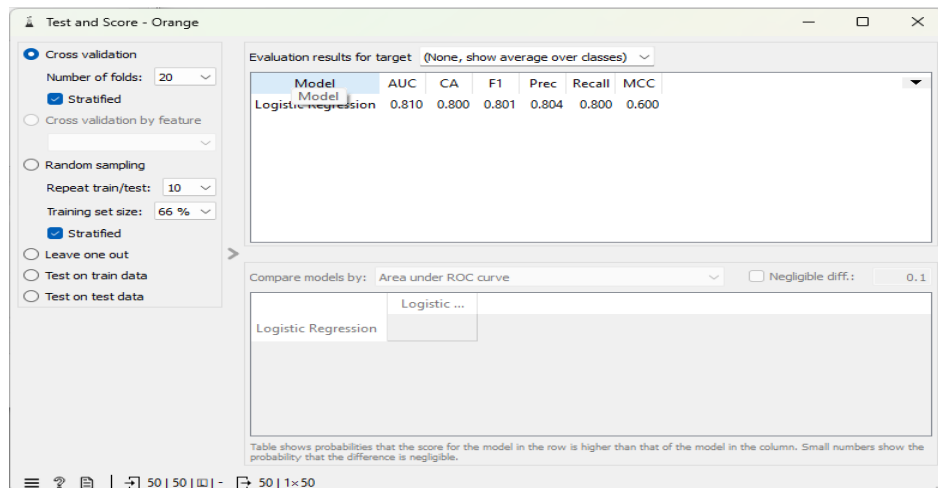
☒ Send Automatically

	name	1
1	intercept	-32.1887
2	Person	0.0265817
3	Daya Tanggap (X1)	0.194522
4	Empati (X2)	0.605506

Gambar 1.3 Coefisien Data

Dari gambar 1.3 bisa didapatkan nilai coefisien data, dengan model persamaan regresi logistic.

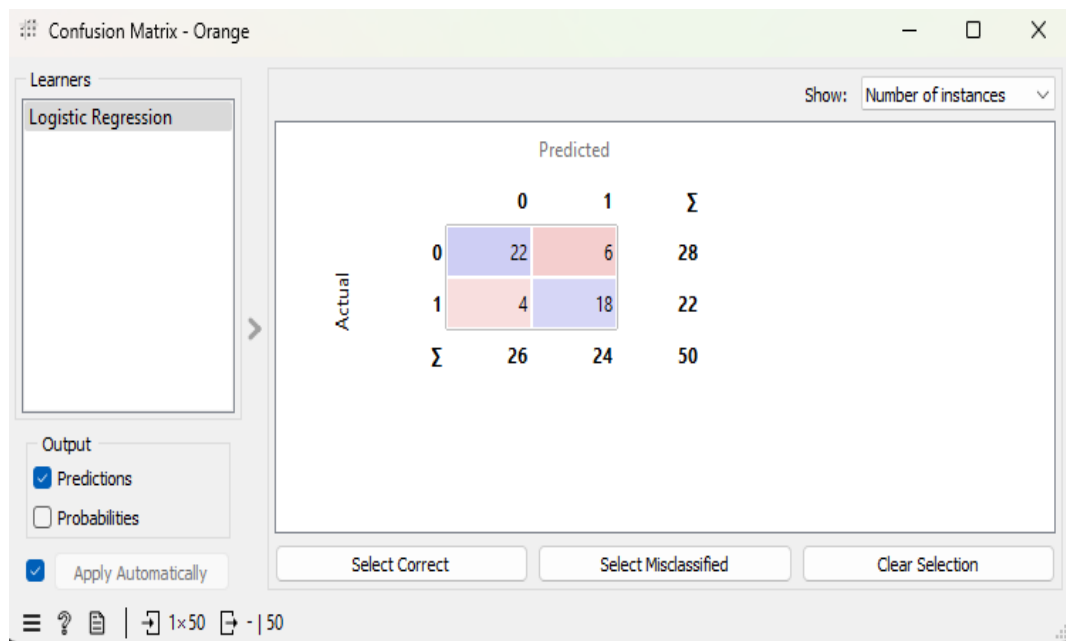
Dengan persamaan logistic



Gambar 1.4 Data Test and Score Pada Data Orange

2. Matrix Pengukuran Regresi logistic dan Confusion Matrix

Berikut output *Confusion Matrix*



Gambar 2.1 Hasil Confusion Matrik

Bisa dilihat dari gambar dibawah ini untuk menjelaskan *Confussion Matrix*

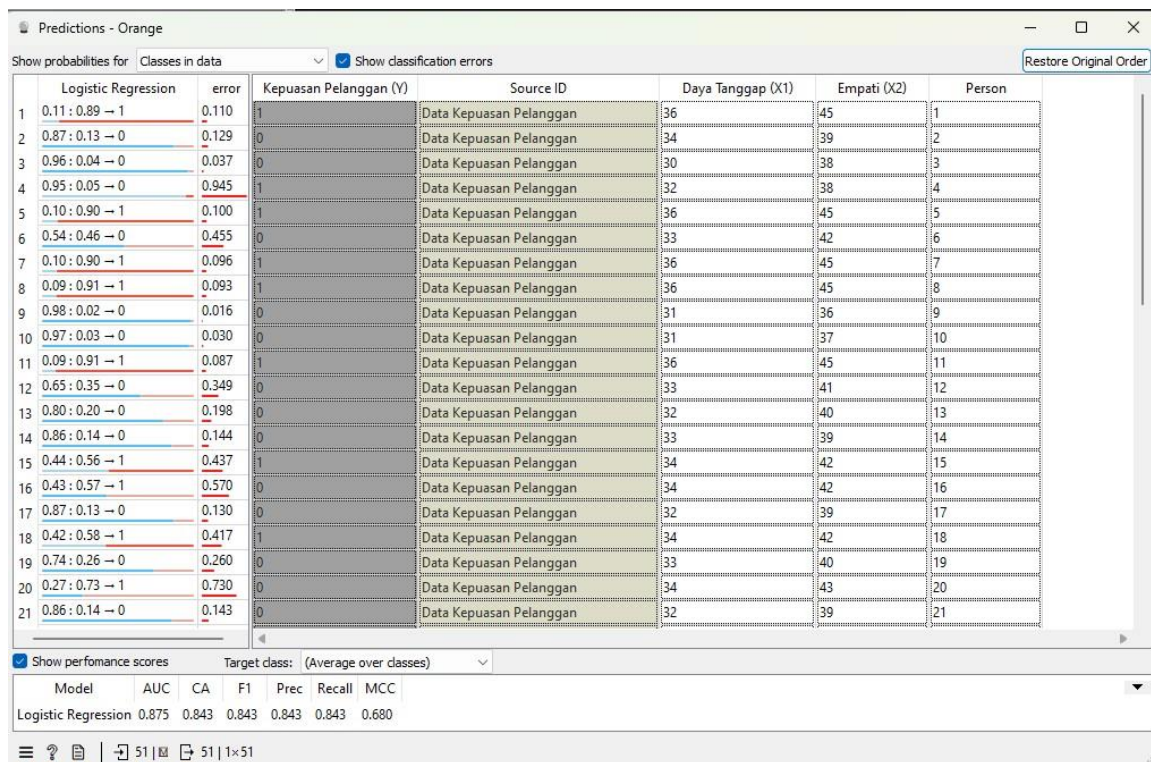
		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	TP (True Positive)	FP (False Positive) <i>Type I Error</i>
	0 (Negative)	FN (False Negative) <i>Type II Error</i>	TN (True Negative)

Gambar 2.2 Confusion Matrix

Dari output confusion matrix diatas dari 22 Pelanggan yang tidak puas, model memprediksi ada 4 pelanggan yang (FN), dan dari 11 pasien negatif kanker, model memprediksi ada 2 pasien yang diprediksi positif kanker (FP). Prediksi yang benar terletak pada tabel diagonal (garis bawah merah), sehingga secara visual sangat mudah untuk melihat kesalahan prediksi karena kesalahan prediksi berada di luar tabel diagonal confusion matrix.

#		Gain ratio	Gini	ANOVA	χ^2	Logi...sion
1	Daya Tanggap (X1)	0.184	0.218	18.899	12.793	0.479
2	Empati (X2)	0.151	0.186	29.652	13.074	1.222
3	Person	0.012	0.016	0.386	0.130	0.312

Gambar 1.6 Range data



Gambar 1.7 Data Hasil Prediksi

CASE II

Data kemampuan matematika peserta tes snmptn dan keberhasilan diterima di prodi pendidikan Matematika

No	Kemampuan Mtk (x)	Keberhasilan (Y)
1	82	1
2	87	1
3	83	0
4	80	0
5	82	0
6	82	1
7	80	1
8	85	1
9	80	0
10	82	0
11	79	0
12	81	0

No	Kemampuan Mtk (x)	Keberhasilan (Y)
19	81	0
20	82	0
21	80	0
22	79	0
23	80	0
24	79	0
25	82	1
26	79	0
27	80	0
28	79	0
29	80	0
30	80	0

13	82	0
14	80	0
15	83	1
16	82	0
17	80	1
18	80	0

31	79	0
32	80	0
33	80	0
34	79	0
35	85	1

3. Kelebihan dan Kekurangan Regresi Logistik

Kelebihan:

- Cocok untuk data biner: Regresi logistik sangat berguna saat Anda bekerja dengan variabel dependen yang bersifat biner atau kategorikal.
- Interpretasi yang Mudah: Hasil dari regresi logistik (koefisien) dapat diinterpretasikan secara langsung dalam hal log-odds atau peluang (odds ratio), membuatnya lebih mudah dipahami.
- Kemampuan untuk Menangani Variabel Independen: Regresi logistik dapat menangani berbagai jenis variabel independen, termasuk kategorikal dan kontinu.
- Mendeteksi Hubungan Non-linear: Meskipun regresi logistik adalah model linier, Anda bisa menambahkan fitur non-linear dengan menambahkan polinomial atau menggunakan transformasi variabel.
- Mengatasi Multikolinearitas: Regresi logistik lebih tahan terhadap masalah multikolinearitas (ketergantungan antar variabel independen) dibandingkan dengan regresi linier.

Kekurangan:

- Asumsi tentang Independensi: Regresi logistik mensyaratkan asumsi independensi antar pengamatan, artinya pengamatan tidak boleh saling bergantung. Ini bisa menjadi masalah jika data yang Anda miliki memiliki ketergantungan temporal atau spasial.
- Rentan terhadap Overfitting: Seperti model statistik lainnya, regresi logistik dapat menjadi rentan terhadap overfitting jika Anda tidak berhati-hati dalam pemilihan fitur atau penyesuaian model.
- Tidak Cocok untuk Variabel Dependensi Kontinu: Regresi logistik dirancang khusus untuk variabel dependen biner, sehingga tidak dapat digunakan langsung untuk variabel dependen yang kontinu.

- Keterbatasan dalam Kompleksitas Model: Regresi logistik merupakan model linier, sehingga tidak dapat menangani hubungan yang sangat kompleks antara variabel independen dan dependen. Ini bisa menjadi kendala jika hubungan tersebut bersifat non-linier.
- Dibutuhkan Data yang Besar: Regresi logistik sering memerlukan jumlah pengamatan yang cukup besar untuk menghasilkan hasil yang dapat diandalkan. Jika jumlah sampel terlalu kecil, estimasi parameter dapat menjadi tidak stabil

4. Summary Jurnal penerapan Regresi Logistik

No	Title Jurnal	Summary	Reference
1	Aplikasi Regresi Logistik Ordinal dalam Pemodelan Status Gizi Balita (Studi Kasus: Puskesmas Limapuluh Di Kota Pekanbaru)	<ul style="list-style-type: none"> • Tujuan dari penelitian di jurnal ini adalah factor paling pengaruh dari status gizi balita lebih tepatnya anak 1-5 th • Factor-faktor yang diambil diantaranya umur, berat badan, tinggi badan, Pendidikan ibu, pekerjaan ibu • Hasil dari kesimpulan penelitian ini adalah 190 orang balita sebanyak 76% mayoritas balita memiliki status gizi baik dan faktor yang paling mempengaruhi status gizi balita adalah berat badan dan pendidikan ibu. 	Pemodelan Status Gizi Balita Menggunakan Regresi Logistik Ordinal (Studi Kasus: Puskesmas Limapuluh Di Kota Pekanbaru) Rahmadeni Jurnal Sains Matematika dan Statistika (uin-suska.ac.id)
2	Deteksi Penyakit Jantung Menggunakan Metode Klasifikasi Decision Tree dan Regresi Logistik	<ul style="list-style-type: none"> • Penelitian ini bertujuan membandingkan kedua metode klasifikasi tersebut • untuk mendeteksi adanya penyakit jantung berdasarkan beberapa indikator • variable yang digunakan diantaranya usia (age), Jenis kelamin pasien, Cp Tipe nyeri dada yang diderita pasien. Dll dan hal yang paling berpengaruh adalah variabel thal (tipe detak jantung pasien) sebagai simpul akar • Dari akurasi dari kedua model tersebut, regresi logistik lebih akurat untuk mendeteksi adanya penyakit jantung dibandingkan model decision tree. 	Sains, Aplikasi, Komputasi dan Teknologi Informasi (unmul.ac.id)

3	Penerapan Teknik Deep Learning (Long Short Term Memory) dan Pendekatan Klasik (Regresi Linier) dalam Prediksi Pergerakan Saham BRI	<ul style="list-style-type: none"> • Penelitian ini bertujuan untuk membandingkan kinerja model algoritma LSTM dengan regresi linear dalam memprediksi harga saham BRI periode 2001-2022 • Penelitian ini membandingkan tingkat akurasi prediksi pada algoritma Long Short-Term Memory (LSTM) dan Regresi Linear berbasis Python maupun aplikasi Orange. • Hasil komparasi algoritma prediksi terhadap dua model dalam data mining, maka model yang lebih akurat adalah algoritma Regresi Linear pada Python. Hal ini dibuktikan dengan nilai RMSE yang lebih rendah 	Jurnal Informatika dan Bisnis Vol. 12 No. 2 Desember 2023 ISSN (p) 2301-9670 (e) 2477-5363
4	Klasifikasi Penderita Anemia Menggunakan Metode Regresi Logistik	<ul style="list-style-type: none"> • Penelitian ini bertujuan untuk mengetahui model klasifikasi penyakit anemia pada remaja putri • Atribut yang digunakan yaitu faktor-faktor yang mempengaruhi anemia diantaranya ferritin serum, STfR, dan riwayat penyakit kronis • Hasil penelitian menunjukkan bahwa terdapat dua variabel yang berpengaruh secara signifikan dalam mengklasifikasi anemia yaitu ferritin serum, dan STfR 	Jurnal Matematika dan Statistika serta Aplikasinya Vol.11 No. 2 Ed. Juli-Des 2023
5	Komparasi Model Pertambahan Tinggi Badan Balita Stunting Dengan Metode Regresi Kuantil dan Regresi Kuantil Bayesian	<ul style="list-style-type: none"> • Data penelitian yang digunakan adalah data 950 balita stunting di Kabupaten Solok pada bulan Agustus 2021 dan bulan Februari 2022. • Pada penelitian ini diperoleh bahwa metode regresi kuantil Bayesian • menghasilkan model dugaan yang lebih baik daripada metode regresi kuantil 	Komparasi Model Pertambahan Tinggi Badan Balita Stunting Dengan Metode Regresi Kuantil dan Regresi Kuantil Bayesian Yanuar Limits: Journal of Mathematics and Its Applications
6	Pemodelan Penerima Bantuan Sosial Masyarakat Kota Surabaya Tahun 2021 Menggunakan Regresi Logistik Multinomial	<ul style="list-style-type: none"> • Hasil analisis diketahui variabel yang memiliki pengaruh signifikan terhadap penerimaan bantuan sosial adalah variabel desil, usia, dan pekerjaan 	JURNAL SAINS DAN SENI ITS Vol. 12, No. 1 (2023), 2337-3520 (2301-928X Print)

7	Pendugaan Koefisien Regresi Logistik Biner Menggunakan Algoritma Least Angle Regression	<ul style="list-style-type: none"> • Pada jurnal penelitian ini, algoritma Least Angle Regression (LAR) digunakan dalam menyeleksi variabel yang signifikan agar mendapatkan model terbaik dari hasil pendugaan koefisien regresi logistik biner. • Algoritma LAR ini diterapkan pada data risiko kejadian stunting pada bayi usia dua tahun atau baduta di wilayah kerja Puskesmas Buntu Batu, Kabupaten Enrekang, Sulawesi Selatan pada tahun 2019. • Hasil yang diperoleh pada estimasi model dugaan regresi logistik biner menggunakan algoritma LAR yaitu nilai standar error sebesar 0.018 lebih kecil dibandingkan nilai standar error pada regresi logistik biner yaitu sebesar 0.025. • dari nilai diatas menunjukan bahwa model regresi logistik biner menggunakan algoritma LAR lebih baik dibandingkan model regresi logistik biner biasa pada data risiko kejadian stunting. • Berdasarkan hasil yang diperoleh maka variabel yang signifikan mempengaruhi risiko stunting pada baduta tahun 2019 di Kabupaten Enrekang adalah tinggi badan ayah, panjang badan lahir, ASI eksklusif, riwayat penyakit infeksi, dan riwayat imunisasi. 	DOI: https://doi.org/10.20956/ejsa.v5i1.12489
8	A predictive analytics model for forecasting outcomes in the National Football League games using decision tree and logistic regression	<ul style="list-style-type: none"> • Dalam studi ini, Dalam jurnal ini membangun model analisis prediktif untuk meramalkan hasil pertandingan NFL dalam satu musim menggunakan pohon keputusan dan regresi logistik. • Beberapa variabel digunakan sebagai prediktor (variabel independen). Ukuran hasil menang-kalah biner digunakan sebagai variabel target (dependen). • Pohon keputusan dan model regresi logistik biner dibangun untuk menggambarkan hubungan antara prediktor dan hasil pertandingan sepak bola di NFL. 	A predictive analytics model for forecasting outcomes in the National Football League games using decision tree and logistic regression - ScienceDirect
9	Analyzing Credit Card Fraud Cases With	<ul style="list-style-type: none"> • Kumpulan data analisis Penipuan Kartu Kredit, yang diperoleh dari 	http://dx.doi.org/10.25045/jpis.v15.i1.06

	Supervised Machine Learning Methods Logistic Regression And Naïve Bayes	<p>database Kaggle, digunakan dalam proses pemodelan bersama dengan metode regresi Logistik dan algoritma Naive Bayes</p> <ul style="list-style-type: none"> • Tujuan dari penelitian ini adalah untuk mengidentifikasi siapa yang melakukan transaksi dengan memeriksa periode ketika orang menggunakan kartu kredit mereka. • Pendekatan regresi logistik dan metode Naive Bayes keduanya memiliki tingkat keberhasilan 99,83%, yang merupakan yang tertinggi. Hasil kedua metode didasarkan pada kappa Cohen, akurasi, presisi, ingatan, dan metrik lainnya. 	
10	<p>Pemodelan Penerima Bantuan Sosial Masyarakat Kota Surabaya Tahun 2021 Menggunakan</p> <p>Regresi Logistik Multinomial</p>	<ul style="list-style-type: none"> • Hasil analisis diketahui variabel yang memiliki pengaruh signifikan terhadap penerimaan bantuan sosial adalah variabel desil, usia, dan pekerjaan 	JURNAL SAINS DAN SENI ITS Vol. 12, No. 1 (2023), 2337-3520 (2301-928X Print)

Cari penjelasan dari :

Model Evaluation	Penjelasan
------------------	------------

Area under ROC	<p>Area under ROC (Area di bawah kurva ROC (Receiver Operating Characteristic)), biasa disingkat sebagai AUC (Area Under the Curve) adalah ukuran kinerja yang umum digunakan untuk mengevaluasi kualitas dari model klasifikasi, termasuk regresi logistik.</p> <p>Nilai AUC berkisar dari 0 hingga 1, di mana semakin dekat ke 1, semakin baik kemampuan model dalam membedakan kelas.</p>				
Classification accuracy	<p>Classification accuracy adalah salah satu metrik evaluasi kinerja yang umum digunakan dalam konteks regresi logistik. Ini mengukur seberapa akurat model dapat mengklasifikasikan data ke dalam kelas yang benar. Secara matematis, classification accuracy dihitung sebagai rasio dari jumlah prediksi yang benar dibagi dengan total jumlah prediksi.</p> <div><div><div>Actual Values</div><div><div>Positive (1)</div><div>Negative (0)</div></div><div><div>Predicted Values</div><div><div>Positive (1)</div><div>Negative (0)</div></div><table><tr><td>TP</td><td>FP</td></tr><tr><td>FN</td><td>TN</td></tr></table></div></div></div> <div>Tabel Confusion matrix</div> <p>Untuk menghitung nilai akurasi, kita dapat menggunakan persamaan matematika berikut:</p> <div><div><div></div><div>$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$</div><div></div></div><div>Atau</div><div><div>$Accuracy = \frac{jumlah\ prediksi\ benar\ (positif\ dan\ negatif)}{jumlah\ keseluruhan\ data}$</div></div><div><div>1. True Positive (TP): merupakan jumlah tebakan yang benar positif/true dan kenyataannya memang benar</div><div>2. True Negative (TN): merupakan jumlah tebakan yang benar negatif/false dan kenyataannya memang benar</div><div>3. False Positive (FP): merupakan jumlah tebakan di mana model kita benar positif tapi kenyataannya itu</div></div></div>	TP	FP	FN	TN
TP	FP				
FN	TN				

	<p>salah</p> <p>4. False Negative (FN): merupakan jumlah tebakan di mana model kita benar negatif tapi kenyataannya itu salah</p>
Precision (Presisi)	<p>Presisi adalah metrik evaluasi yang mengukur seberapa baik model membuat prediksi yang benar untuk kelas positif dari total prediksi positif yang dilakukan.</p> <p>Untuk menghitung presisi, kita bisa menggunakan rumus matematika berikut:</p> $Precision = \frac{TP}{TP + FP}$ <p>Atau</p> $Precision = \frac{\text{jumlah prediksi benar (positif)}}{\text{prediksi benar (positif)} + \text{prediksi salah (positif)}}$
Recall	<p>Sensitivitas (Recall) adalah metrik evaluasi yang menggambarkan seberapa baik suatu model dalam mengidentifikasi kelas positif dengan benar.</p> <p>Untuk menghitung nilai Recall, kita dapat menggunakan persamaan matematika berikut:</p> $Recall = \frac{TP}{TP + FN}$ <p>Atau</p> $Recall = \frac{\text{jumlah prediksi benar (positif)}}{\text{prediksi benar (positif)} + \text{prediksi salah (negatif)}}$
Specificity	<p>Spesifisitas (Specificity) merupakan metrik evaluasi yang menunjukkan seberapa efektif suatu model dalam mengklasifikasikan kelas negatif secara akurat.</p> <p>Untuk menghitung Spesifisitas, kita gunakan rumus berikut:</p> $Specificity = \frac{TN}{TN + FP}$ <p>Atau</p> $Specificity = \frac{\text{jumlah prediksi benar (negatif)}}{\text{prediksi benar (negatif)} + \text{prediksi salah (positif)}}$
F-1	<p>F1 Score merupakan metrik evaluasi yang mencerminkan keseimbangan antara Presisi (Precision) dan Sensitivitas (Recall).</p>

	<p>Untuk menghitung F1 Score, kita menggunakan rumus matematika berikut:</p> $F1\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision}$
LogLoss (Logarithmic Loss) atau cross-entropy)	<p>LogLoss (Logarithmic Loss) atau cross-entropy adalah metrik evaluasi yang umum digunakan dalam regresi logistik, terutama dalam konteks klasifikasi biner. Tujuannya adalah untuk mengevaluasi seberapa baik probabilitas yang diprediksi oleh model sesuai dengan probabilitas sebenarnya dari data.</p> <p>Secara matematis, LogLoss dihitung sebagai:</p> $H(P, Q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(P(y_i)) + (1 - y_i) \cdot \log(1 - P(y_i))$
Matthews correlation coefficient	<p>Matthews Correlation Coefficient (MCC) biasanya digunakan dalam konteks klasifikasi, MCC adalah metrik evaluasi yang mengukur kualitas prediksi dalam klasifikasi biner, yang mempertimbangkan semua empat sel dari matriks kebenaran (true) dan prediksi.</p> <p>Formula matematis dari MCC adalah sebagai berikut:</p> $MCC = \frac{TN \times TP - FP \times FN}{\sqrt{(TN + FN)(FP + TP)(TN + FP)(FN + TP)}}$

TUGAS PERTEMUAN KE 4

MODEL DECISION TREE

TUGAS

1. Buat Model untuk Decision Tree
2. Kelebihan dan Kekurangan Decision Tree
3. Cari 10 Jurnal terkait pemanfaatan Decision Tree
4. Diskusikan dalam Forum
5. Tuliskan dalam laporan (dikumpulkan saat UTS)

1. Model Decision Tree

1.1. Deskripsi data

Dataset iris adalah dataset multivariat yang berisi data observasi dari 3 spesies bunga iris, yaitu Iris Setosa, Iris Virginica, dan Iris Versicolor, dengan 50 observasi per spesies.

Dataset iris berisi 4 atribut yang dapat mempengaruhi klasifikasi, yaitu: Panjang sepal (sepal length), Lebar sepal (sepal width), Panjang kelopak (petal length), Lebar kelopak (petal width)

1.2. Pengolahan Data

Untuk pengolahan data menggunakan python dengan tools **googlecolabs**, berikut ini scrip pengolahan datanya:

```
from sklearn.tree import DecisionTreeClassifier
from sklearn import datasets
import matplotlib.pyplot as plt
# membaca Dataset dari file ke pandas dataframe
irisDataset = pd.read_csv('Dataset Iris.csv',
                          delimiter=';',
                          header=0)
# menghapus kolom id karena tidak perlu diikuti sertakan
irisDataset = irisDataset.drop(labels="Id",axis=1)
# load datasets iris dari library
iris = datasets.iris_data()
features = iris['data']
target = iris['target']
# membuat object decision tree
decisiontree = DecisionTreeClassifier(random_state=0,
                                      max_depth=None,
                                      min_samples_split=2,
                                      min_samples_leaf=1,
                                      min_weight_fraction_leaf=0,
```

```

max_leaf_nodes=None,
min_impurity_decrease=0)

# mentraining decission tree
model = deicisiontree.fit(features,target)
# membuat grafik visualisasi diagrama decision tree
import pydotplus
from sklearn import tree
dot_data = tree.export_graphviz(deicisiontree,
                                out_file=None,
                                feature_names=iris['feature_names'],
                                class_names=iris['target_names'])

from IPython.display import Image
graph = pydotplus.graph_from_dot_data(dot_data)
Image(graph.create_png())
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
# misahin data jadi data latih dan data uji
X_train, X_test, y_train, y_test = train_test_split(iris.data,
iris.target, test_size=0.2, random_state=42)

# Membuat model klasifikasi (misalnya, Random Forest Classifier)
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)

# Melakukan prediksi terhadap data uji
y_pred = model.predict(X_test)

# memanggil fungsi confusion matrik
cm = confusion_matrix(y_test, y_pred)

# Plot confusion matrix
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, cmap='Blues', fmt='d',
xticklabels=iris.target_names, yticklabels=iris.target_names)
plt.xlabel('Predicted labels')
plt.ylabel('True labels')
plt.title('Confusion Matrix')
plt.show()

from sklearn.metrics import f1_score

# Menghitung nilai F1 score
f1 = f1_score(y_test, y_pred, average='weighted') # 'weighted'

```

```

untuk klasifikasi multikelas
print("F1 Score:", f1)
#Output
F1 Score: 1.0
from sklearn.metrics import recall_score

# Menghitung nilai recall
recall = recall_score(y_test, y_pred, average='weighted') #
'weighted' untuk klasifikasi multikelas
print("Recall Score:", recall)
from sklearn.metrics import matthews_corrcoef

# Menghitung nilai MCC
mcc = matthews_corrcoef(y_test, y_pred)
print("Matthews Correlation Coefficient (MCC):", mcc)

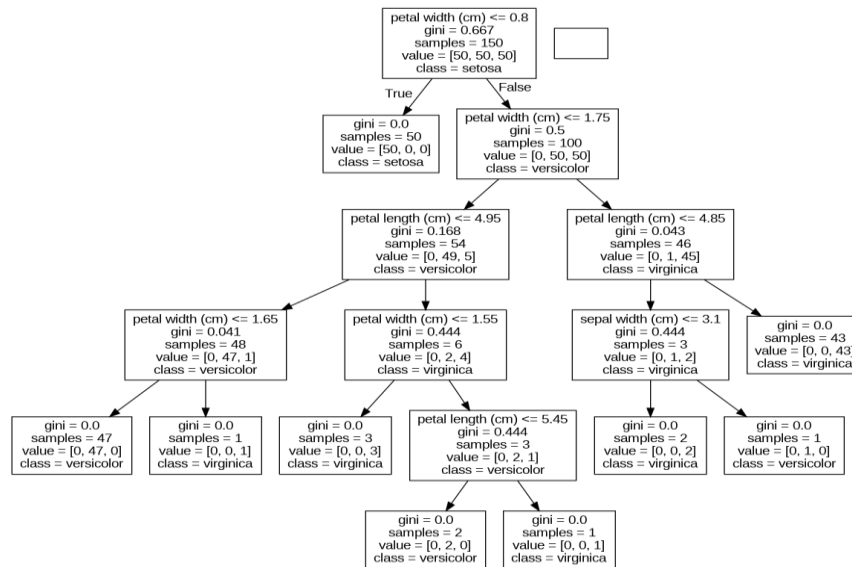
Recall Score: 1.0
from sklearn.metrics import accuracy_score

# Menghitung nilai CA
accuracy = accuracy_score(y_test, y_pred)
print("Classification Accuracy (CA):", accuracy)
Matthews Correlation Coefficient (MCC): 1.0
from sklearn.metrics import log_loss

# Menghitung nilai Log Loss
proba = model.predict_proba(X_test) # Prediksi probabilitas untuk
setiap kelas
logloss = log_loss(y_test, proba)
print("Log Loss:", logloss)
Log Loss: 0.025117573121763905
from sklearn.metrics import roc_auc_score

```

Adapun diagram decision tree didapat seperti dibawah ini:



Nilai output f1 didapat 1.0 ini artinya

```
from sklearn.metrics import f1_score

# Menghitung nilai F1 score
f1 = f1_score(y_test, y_pred, average='weighted') # 'weighted'
untuk klasifikasi multikelas
print("F1 Score:", f1)
```

2. Kelebihan dan Kekurangan Decision Tree

2.1 Kelebihan:

1. Interpretasi yang Mudah: Decision tree dapat dengan mudah diinterpretasikan oleh manusia. Anda bisa memvisualisasikan strukturnya dan melihat alur logika yang digunakan untuk membuat keputusan.
2. Mampu Menangani Data Non-linear: Decision tree dapat menangani hubungan non-linear antara fitur dan variabel target, tanpa perlu transformasi data tambahan.
3. Tidak Memerlukan Persyaratan Data Khusus: Decision tree tidak memerlukan asumsi tentang distribusi data atau keseimbangan kelas yang seragam seperti beberapa metode lainnya. Ini membuatnya cocok untuk berbagai jenis data.
4. Cocok untuk Pengklasifikasi Multi-kelas: Decision tree dapat dengan mudah diadaptasi untuk melakukan pengklasifikasi pada masalah dengan lebih dari dua kelas.

5. Cocok untuk Fitur Campuran: Decision tree dapat menangani campuran jenis fitur (numerik dan kategorikal) tanpa memerlukan pre-processing yang rumit.

2.2 Kekurangan

1. Kemungkinan Overfitting: Decision tree cenderung mempelajari pola yang terlalu spesifik pada data pelatihan dan menjadi overfit pada data tersebut. Hal ini dapat diatasi dengan menggunakan teknik seperti pruning atau ensemble methods.
2. Keterbatasan dalam Menangani Masalah XOR: Decision tree sulit untuk menangani masalah XOR (exclusive OR), di mana kelas-kelas yang didefinisikan oleh kombinasi fitur memiliki hubungan yang rumit.
3. Rentan terhadap Variasi Data: Decision tree dapat menghasilkan struktur yang sangat berbeda jika data yang digunakan sedikit berbeda, terutama pada dataset yang kecil atau rentan terhadap noise.
4. Tidak Stabil terhadap Perubahan Kecil dalam Data: Karena kecilnya perubahan dalam data dapat menyebabkan perubahan besar dalam struktur decision tree, model tersebut tidak stabil terhadap variasi data.
5. Tidak Cocok untuk Variabel Kontinu: Decision tree tidak dapat menangani variabel target yang kontinu secara langsung, meskipun dapat digunakan dalam masalah regresi melalui teknik seperti regresi tree.

SUMMARY JURNAL PENERAPAN DECISION TREE

No	Title Jurnal	Summary	Reference
1	Research on Chinese Public Policy Decision Model Based on Decision Tree Algorithm	<ul style="list-style-type: none"> Dalam Jurnal ini, algoritma klasifikasi pohon keputusan digunakan untuk menetapkan model prediksi keputusan kebijakan publik Tiongkok. Makalah ini memilih atribut karakteristik berdasarkan prinsip pembuatan kebijakan. Kemudian, dengan mengambil hasil keputusan kebijakan publik sebagai label target, makalah ini mengoptimalkan model dengan menyesuaikan kedalaman maksimum pohon keputusan, jumlah minimum sampel daun dan ambang batas keputusan. Set uji memverifikasi bahwa model pohon keputusan yang dioptimalkan memiliki efek prediktif yang baik pada prediksi hasil model keputusan kebijakan publik. Nilai AUC adalah 0,848 dan model memiliki kemampuan generalisasi yang kuat. Perbedaan AUC antara set pelatihan dan set tes kurang dari 0,04. 	https://drpress.org/ojs/index.php/HSET/article/view/10592
2	Optimal Tree Depth in Decision Tree Classifiers for Predicting Heart Failure Mortality	<ul style="list-style-type: none"> Penelitian ini mengusulkan validasi silang untuk menemukan kedalaman pohon yang optimal menggunakan data validasi. Dalam metode yang diusulkan, akurasi yang divalidasi silang untuk data pelatihan dan pengujian diuji secara empiris menggunakan himpunan data HF, yang berisi 299 pengamatan dengan 11 fitur yang dikumpulkan dari repositori data pembelajaran mesin (ML) Kaggle. Hasil pengamatan menunjukkan bahwa penyetelan kedalaman DT sangat penting untuk menyeimbangkan proses pembelajaran DT karena pola yang relevan ditangkap dan overfitting dihindari. Meskipun teknik validasi silang terbukti efektif dalam menentukan kedalaman DT optimal, penelitian ini tidak membandingkan metode yang berbeda untuk menentukan kedalaman optimal, seperti pencarian grid, algoritma pemangkasan, atau kriteria informasi. Inilah keterbatasan penelitian ini. 	HF_01.01_05.pdf (acadlore.com)
3	A predictive analytics model for forecasting outcomes in the National Football League games using decision tree and logistic regression	<ul style="list-style-type: none"> Dalam studi ini, Dalam jurnal ini membangun model analisis prediktif untuk meramalkan hasil pertandingan NFL dalam satu musim menggunakan pohon keputusan dan regresi logistik. Beberapa variabel digunakan sebagai prediktor (variabel independen). Ukuran hasil menang-kalah biner digunakan sebagai variabel target (dependen). Pohon keputusan dan model regresi logistik biner dibangun untuk menggambarkan hubungan antara prediktor dan hasil pertandingan sepak bola di NFL. 	A predictive analytics model for forecasting outcomes in the National Football League games using decision tree and logistic regression - ScienceDirect
4	Deteksi Penyakit Jantung Menggunakan	<ul style="list-style-type: none"> Penelitian ini bertujuan membandingkan kedua metode klasifikasi tersebut untuk mendeteksi adanya penyakit jantung berdasarkan 	Sains, Aplikasi, Komputasi dan Teknologi Informasi (unmul.ac.id)

	Metode Klasifikasi Decision Tree dan Regresi Logistik	<p>beberapa indicator</p> <ul style="list-style-type: none"> • variable yang digunakan diantaranya usia (age), Jenis kelamin pasien, Cp Tipe nyeri dada yang diderita pasien. Dll dan hal yang paling berpengaruh adalah variabel thal (tipe detak jantung pasien) sebagai simpul akar • Dari akurasi dari kedua model tersebut, regresi logistik lebih akurat untuk mendeteksi adanya penyakit jantung dibandingkan model decision tree. 	
5	Novel Classification Method: Neighborhood-Based Positive Unlabeled Learning Using Decision Tree (NPULUD)	<ul style="list-style-type: none"> • pada jurnal ini, disajikan metode baru: pembelajaran positif tanpa label berbasis lingkungan menggunakan pohon keputusan (NPULUD). • Pertama, NPULUD menggunakan pendekatan lingkungan terdekat untuk strategi PU dan kemudian menggunakan algoritma pohon keputusan untuk tugas klasifikasi dengan memanfaatkan ukuran entropi. • Entropi memainkan peran penting dalam menilai tingkat ketidakpastian dalam dataset pelatihan, karena pohon keputusan dikembangkan dengan tujuan klasifikasi • pada jurnal ini memvalidasi metode lebih dari 24 set data dunia nyata. Metode yang diusulkan mencapai akurasi rata-rata 87,24%, sedangkan pendekatan pembelajaran tradisional yang diawasi memperoleh akurasi rata-rata 83,99% pada dataset. • pada jurnal ini juga ditunjukkan bahwa metode ini memperoleh peningkatan yang signifikan secara statistik (7,74%), sehubungan dengan rekan-rekan canggih, rata-rata 	https://www.mdpi.com/1099-4300/26/5/403
6	Komparasi Metode Decision Tree dan Deep Learning dalam Meramalkan Jumlah Mahasiswa Drop Out Berdasarkan Nilai Akademik	<ul style="list-style-type: none"> • pada jurnal ini Peneliti menggunakan dua jenis metode yang berbeda yang dikomparasikan untuk menemukan hasil pemodelan terbaik. Metode yang digunakan yakni decision tree dengan algoritma C5.0 serta deep learning dengan algoritma GRU. • Data peramalan yang digunakan yaitu Data Mahasiswa Program Studi Teknologi Informasi angkatan tahun 2010-2016. • Metode dengan performa terbaik pada penelitian ini yaitu metode decision tree C5.0 yang menghasilkan nilai akurasi sebesar 95% dengan persentase kesalahan RMSE 0.13001950438859716 dan MAPE 2.26% Metode deep learning menunjukkan hasil yang cenderung lebih rendah dibanding decision tree C5.0 dengan nilai akurasi sebesar 92% dan persentase kesalahan RMSE 0.1873780487675864 MAPE 4.69%. 	https://journal.pubmedia.id/index.php/pjise/article/view/2327
7	Implementasi Algoritma Decision Tree dan Support Vector Machine untuk Klasifikasi Penyakit Kanker Paru	<ul style="list-style-type: none"> • Pada jurnal ini peneliti melakukan teknik klasifikasi yang merupakan suatu metode pengelompokan data yang memiliki karakter yang sama ke dalam beberapa kelompok. • Teknik klasifikasi yang diteliti membandingkan 2 algoritma yaitu, algoritma Decision Tree dan Support Vector Machine (SVM) untuk mengetahui algoritma 	https://journal.irpi.or.id/index.php/malcom/article/view/591

		<p>yang memberikan hasil terbaik.</p> <ul style="list-style-type: none"> • Dalam penelitian ini akan dilakukan seleksi fitur menggunakan forward selection yang bertujuan untuk menaikkan nilai akurasi. • Berdasarkan penelitian yang telah dilakukan didapatkan hasil dari algoritma SVM menggunakan feature selection mempunyai nilai akurasi yang lebih unggul yaitu 62,3% menggunakan splitting data 80:20. 	
8	Deteksi Serangan Low Rate Ddos Pada Jaringan Tradisional Menggunakan Machine Learning Dengan Algoritma Decision Tree	<ul style="list-style-type: none"> • pada jurnal ini aplikasinya uji model decesion tree untuk mendeteksi serangan Low Rate DDoS (Distributed Denial of Service) pada jaringan tradisional, decision tree dapat digunakan untuk memprediksi kemungkinan terjadinya serangan Low rate DDoS berdasarkan beberapa fitur yang dianggap penting dalam mengidentifikasi serangan tersebut. • Fitur-fitur tersebut bisa berupa jumlah traffic yang masuk ke jaringan, tipe traffic yang masuk, atau karakteristik traffic lainnya. Setelah fitur-fitur tersebut dikumpulkan, Decision tree dapat digunakan untuk memprediksi kemungkinan terjadinya serangan Low rate DDoS pada jaringan tradisional dengan menganalisis fitur-fitur yang dianggap penting dan membuat keputusan berdasarkan pertanyaan-pertanyaan yang sesuai. • pada jurnal ini peneliti bertujuan untuk menganalisis perbandingan hasil dari dua metode decision tree, yaitu algoritma Gini Index dan Entropy, untuk mendeteksi serangan low rate DDoS (Distributed Denial of Service) pada jaringan tradisional dengan menggunakan dataset CICIDS 2017 • Hasil analisis menunjukkan bahwa metode decision tree dengan algoritma Gini Index lebih baik dari Entropy untuk mendeteksi low rate DDoS (Distributed Denial of Service) pada jaringan tradisional berdasarkan nilai Accuracy, Precision, dan F1 Score, yaitu dengan nilai 99,740%, 99,113%, dan 99,231%. Namun, metode decision tree dengan algoritma Entropy lebih baik dari Gini Index berdasarkan nilai Recall, yaitu dengan nilai 99,351% 	https://ejournal.uin-suka.ac.id/sainstek/cybersecurity/article/view/3951
9	Online Learning Behavior Feature Mining Method Based on Decision Tree	<ul style="list-style-type: none"> • Pengumpulan data pada jurnal ini pengumpulan data perilaku pembelajaran online secara real-time situs pembelajaran jarak jauh. Teknologi OWC (komponen web kantor) digunakan untuk menggambar grafik waktu nyata di halaman. • Proses evaluasi untuk menghasilkan pohon keputusan yang lengkap adalah • diselesaikan dengan algoritma c4.5tree di C4.5, yang dapat diberi nama dengan akhiran .names. • Tipe file definisi digunakan untuk mencatat jenis setiap item atribut atau rentang nilai yang mungkin. Dalam studi, tingkat akurasi prediksi dalam memprediksi efek pembelajaran berdasarkan “perilaku pembelajaran 	https://www.igi-global.com/gateway/article/full-text-pdf/295244&riu=true

		online”	
		<ul style="list-style-type: none"> • mencapai lebih dari 66%. 	
10	Machine Learning Dengan Decision Tree untuk Prediksi Pembayaran Invoice, Case Study : Gramedia Jakarta	<ul style="list-style-type: none"> • latar belakang pada jurnal ini belum adanya alat yang dapat memprediksi pembayaran faktur di Gramedia menyulitkan bagian keuangan. Dari permasalahan itu, maka diterapkan machine learning untuk memprediksi pembayaran faktur oleh customer, apakah pembayarannya terlambat atau tidak terlambat. • Fitur data yang digunakan sebagai parameter yaitu invoice amount, payment method, paid invoice, average days late dan ratio amount of overdue by amount of balance • Data faktur penjualan diprediksi menggunakan model decision tree algoritma C5.0 dengan hasil akurasi mencapai 71.84% • Algoritma C5.0 terbukti mampu memprediksi faktur yang pembayarannya terlambat (melewati jatuh tempo) dan pembayarannya tepat waktu (sebelum jatuh tempo). 	https://ojs.uma.ac.id/index.php/jite/article/view/5066

TUGAS PERTEMUAN 5

NAÏVE BAYES

Tugas

1. Buat Model untuk Naïve Bayes
2. Kelebihan dan Kekurangan Naïve Bayes
3. Cari 10 Jurnal terkait pemanfaatan Naïve Bayes
4. Diskusikan dalam Forum
5. Tuliskan dalam laporan (dikumpulkan saat UTS)

1. Buat Model Naïve Bayes

1.1 Deskripsi data

Dataset yang digunakan adalah dataset *adult income* yang didapat dari kaagle, berikut gambaran datanya :

Pendapatan tahunan seseorang dihasilkan dari berbagai faktor. Secara intuitif dipengaruhi oleh tingkat pendidikan individu, usia, jenis kelamin, pekerjaan, dan lain-lain. Kumpulan data berisi 16 kolom

Target yang diajukan: Pendapatan -- Pendapatan dibagi menjadi dua kelas: $\leq 50K$ dan $> 50K$

1.2 Pengolahan data

Pengolahan data dilakukan dengan Bahasa pemrogramana python menggunakan tools jupyter python, berikut data pengolahan datanya :

```
# TUGAS NAIVE BAYES ASEP RIDWAN HIDAYAT
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, log_loss
from sklearn.preprocessing import RobustScaler
from sklearn.naive_bayes import GaussianNB
import seaborn as sns
%matplotlib inline

# select data
df = pd.read_csv ("Data Set Adults.csv")

# tambahkan nama kolom
nama_kolom = ['age', 'workclass', 'fnlwgt', 'education',
'education_num', 'marital_status', 'occupation', 'relationship',
```

```

        'race', 'sex', 'capital_gain', 'capital_loss',
        'hours_per_week', 'native_country', 'income']

# insert nama kolom ke dataframe
df.columns = nama_kolom

# tampilan data
# df.head()

# jadikan dataset into features dan target variable
X = df.drop(['income'], axis=1)
y = df['income']

# jadikan X y training dan testing sets
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size = 0.3, random_state = 0)

# encode data supaya jadi kategori
import category_encoders as encode
encoder = encode.OneHotEncoder(cols=['workclass', 'education',
'marital_status', 'occupation', 'relationship',
                                'race', 'sex',
'native_country'])

X_train = encoder.fit_transform(X_train)
X_test = encoder.transform(X_test)

# menskalakan data sesuai dengan rentang kuantil (default pada
IQR: Rentang Interkuartil).agar tidak ada outlier
cols = X_train.columns

scaler = RobustScaler()

X_train = scaler.fit_transform(X_train)

X_test = scaler.transform(X_test)

X_train = pd.DataFrame(X_train, columns=[cols])
X_test = pd.DataFrame(X_test, columns=[cols])

# train a Gaussian Naive Bayes classifier on the training set
# instantiate the model
gnb = GaussianNB()

```

```

# fit the model
gnb.fit(X_train, y_train)

#prediksi hasil
y_pred = gnb.predict(X_test)

#buat koefesioen matrik
from sklearn.metrics import confusion_matrix

cm = confusion_matrix(y_test, y_pred)

# Print Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)

#munculkan data
print('\nConfusion matrix\n', cm)
print('True Positives(TP) = ', cm[0,0])
print('True Negatives(TN) = ', cm[1,1])
print('False Positives(FP) = ', cm[0,1])
print('False Negatives(FN) = ', cm[1,0])

#ambil nilai dari matrik confusion diatas TP,TN,FP,FN
TP = cm[0,0]
TN = cm[1,1]
FP = cm[0,1]
FN = cm[1,0]

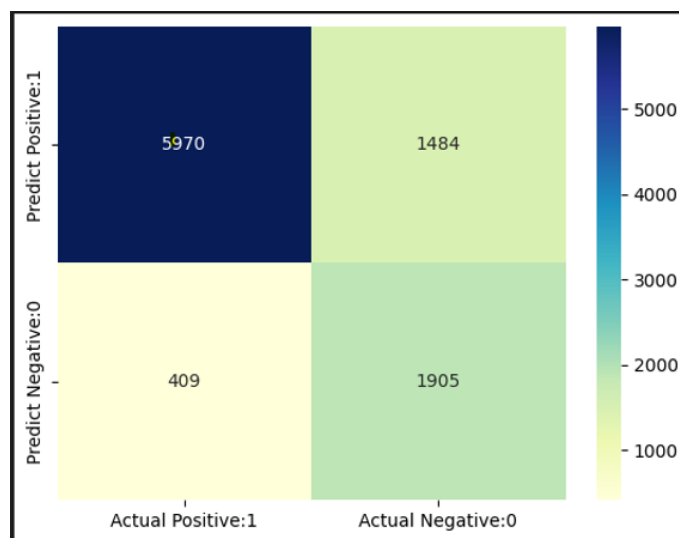
#Ambil nilai CA
classification_accuracy = (TP + TN) / float(TP + TN + FP + FN)
# Ambil nilai precision
precision = TP / float(TP + FP)
# Ambil nilai recall
recall = TP / float(TP + FN)
#ambil nilai spesifikasi
specificity = TN / (TN + FP)
#Ambil nilai f1
f1 = 2 * (precision * recall) / (precision + recall)
#Ambil nilai MCC
mcc = (TP * TN - FP * FN) / ((TP + FP) * (TP + FN) * (TN + FP) *
(TN + FN)) ** 0.5

#Munculkan nilai
print('Classification accuracy :
{0:0.4f}'.format(classification_accuracy))
print('Precision : {0:0.4f}'.format(precision))
print('Recall or Sensitivity : {0:0.4f}'.format(recall))

```

```
print('Specificity : {0:0.4f}'.format(specificity))
print('f1 : {0:0.4f}'.format(f1))
print('mcc {0:0.4f}'.format(mcc))
```

1.3 Output data



Gambar 1.3 Output Confusion Matrix

Dari hasil confusion matrix diatas dapat interpretasikan

Output nilai didapat :

- Classification accuracy : 0.8062
- Precision : 0.8009
- Recall or Sensitivity : 0.9359
- Specificity : 0.5621
- f1 : 0.8632
- mcc 0.5575

2. Kelebihan dan Kekurangan Naïve Bayes

Kelebihan

- Sederhana dan mudah diimplementasikan
- Efisien dalam kinerja
- Cocok untuk dataset besar
- Tahan terhadap noise
- Cocok untuk kategori diskrit
- Mudah diinterpretasi

Kekurangan

- Asumsi independensi yang kuat, yang sering tidak cocok dengan data dunia nyata
- Estimasi probabilitas nol dapat menyebabkan masalah prediksi
- Sensitif terhadap data latensi atau kesalahan dalam data
- Tidak mampu menangani masalah kontinu tanpa modifikasi
- Kinerja relatif rendah dalam kasus dengan hubungan fitur yang kompleks

SUMMARY JURNAL PENERAPAN NAIV BAYES

No	Title Jurnal	Summary	Reference
1	Predicting Students' Academic Performance Through Machine Learning Classifiers: A Study Employing the Naive Bayes Classifier (NBC)	<ul style="list-style-type: none">• pada jurna ini memprediksi kinerja yang akurat dan identifikasi siswa awal, dengan metode yang diterapkan secara luas dalam memprediksi kinerja siswa berdasarkan berbagai sifat.• Memanfaatkan model pengklasifikasi Naive Bayes (NBC), penelitian ini memprediksi kinerja siswa dengan memanfaatkan kemampuan kuat yang melekat pada alat klasifikasi ini.• Untuk meningkatkan efisiensi dan akurasi, model ini mengintegrasikan dua algoritma optimasi, yaitu Jellyfish Search Optimizer (JSO) dan Artificial Rabbit Optimization (ARO).• analisis komprehensif dari informasi yang berkaitan dengan 395 siswa telah dilakukan. Hasil penelitian menunjukkan bahwa dalam memprediksi G1, model NBAR, dengan F1_Score 0,882, berkinerja hampir 1,03% lebih baik daripada model NBJS, yang memiliki F1_Score 0,873. Dalam prediksi G3, model NBAR mengungguli model NBJS dengan nilai F1_Score masing-masing 0,893 dan 0,884.	https://thesai.org/Publications/ViewPaper?Volume=15&Issue=1&Code=IJACSA&SerialNo=99
2	Penerapan Text	<ul style="list-style-type: none">• Dari penelitian di jurnal ini yang dilakukan menghasilkan sebuah perangkat lunak penerapan text mining pada	https://www.neliti.com/publicatio

	<p>Mining Pada Sistem Klasifikasi Email Spam Menggunakan Naive Bayes</p>	<p>sistem klasifikasi email spam menggunakan metode naive bayes.</p> <ul style="list-style-type: none"> • Pada klasifikasi email dihitung nilai probabilitas berdasarkan kemunculan kata yang terdapat dalam data email. pengujian keakurasian sistem ditampilkan berupa grafik nilai keakurasian, false positif dan false negatif. • Hasil uji coba menunjukkan bahwa aplikasi ini layak dan dapat digunakan dan memiliki nilai keakurasian sistem sebesar 89,6 % 	<p>ns/487630/penerapan-text-mining-pada-sistem-klasifikasi-email-spam-menggunakan-naive-bayes</p>
3	<p>Classification of Farms for Recommendation of Rice Cultivation Using Naive Bayes and SVM</p>	<ul style="list-style-type: none"> • jurnal ini dilatarbelakangi masalah identifikasi pengguna di beberapa jejaring sosial (UIAMSNs) menarik perhatian yang cukup besar karena merupakan prasyarat untuk banyak tugas dan aplikasi. • Untuk mengatasi kesulitan di atas, kami mengusulkan algoritma identifikasi pengguna berdasarkan model Bayes naif (UI-NBM) dalam kerangka kerja berbasis fitur jaringan. • Pertama, indeks derajat pencocokan dirancang berdasarkan model Bayes naif, yang secara akurat dapat mengukur kontribusi pasangan simpul yang cocok umum (MNP) yang berbeda terhadap probabilitas koneksi pasangan simpul yang tidak cocok (UMNP). Kedua, derajat yang cocok dari semua UMNP dirumuskan sebagai produk matriks, sehingga menimbulkan pengurangan besar kompleksitas waktu dan ekspresi kompak • Akhirnya, dengan gagasan proses rekursi, lebih banyak UMNP dapat diprediksi secara iteratif bahkan ketika hanya sejumlah kecil informasi sebelumnya (yaitu, beberapa jumlah MNP) diketahui • Hasil eksperimen pada platform silang sintetis dan nyata menunjukkan bahwa metode ini mengungguli metode dasar dalam kerangka kerja berbasis fitur. 	<p>https://ieeexplore.ieee.org/document/9893090/</p>
4	<p>Analyzing Credit Card Fraud Cases With Supervised Machine Learning Methods Logistic Regression And Naïve Bayes</p>	<ul style="list-style-type: none"> • Kumpulan data analisis Penipuan Kartu Kredit, yang diperoleh dari database Kaggle, digunakan dalam proses pemodelan bersama dengan metode regresi Logistik dan algoritma Naive Bayes • Tujuan dari penelitian ini adalah untuk mengidentifikasi siapa yang melakukan transaksi dengan memeriksa periode ketika orang menggunakan kartu kredit mereka. • Pendekatan regresi logistik dan metode Naive Bayes keduanya memiliki tingkat keberhasilan 99,83%, yang merupakan yang tertinggi. Hasil kedua metode didasarkan pada kappa Cohen, akurasi, presisi, ingatan, dan metrik lainnya. 	<p>http://dx.doi.org/10.25045/jpis.v15.i1.06</p>
5	<p>Implementasi Algoritma Naïve Bayes Classifier dan K-Nearest Neighbor untuk Klasifikasi Penyakit Ginjal Kronik</p>	<ul style="list-style-type: none"> • latar belakang di penelitian jurnal ini adalah dari jumlah nilai prelevansi penderita Penyakit Ginjal Kronik di Indonesia yang terbilang besar. Penyakit ginjal kronik ialah kondisi di mana ginjal mengalami penurunan fungsi yang berlangsung dalam jangka waktu yang lama. • Berdasarkan Hasil penelitian yang diperoleh klasifikasi PGK menggunakan algoritma NBC memiliki akurasi sebesar 94,25%, rata-rata nilai recall 94,23%, presisi 98,40% dan AUC 0,961, Sedangkan klasifikasi menggunakan algoritma KNN memiliki akurasi sebesar 77,79%, recall 95,06%, presisi 80,20% dan AUC sebesar 0,627. 	<p>https://journal.irpi.or.id/index.php/malcom/article/view/1229</p>

		<ul style="list-style-type: none"> Dari kedua hasil menunjukan bahwa klasifikasi menggunakan algoritma NBC lebih baik dibanding menggunakan algoritma KNN. 	
6	Prediksi Risiko Stunting pada Keluarga Menggunakan Naïve Bayes Classifier dan Chi-Square	<ul style="list-style-type: none"> jurnal ini ditulis berdasarkan Meningkatnya kasus stunting pada balita memerlukan suatu upaya dalam penanganan dan pencegahan secara dini. Terdapat 17 atribut pada data stunting yang harus diperhatikan, dengan banyaknya atribut tersebut menyebabkan sulitnya menemukan atribut yang paling berpengaruh dalam memprediksi stunting. Pada penelitian ini diterapkan seleksi fitur menggunakan Chi Square dan menerapkan Algoritma Naïve Bayes untuk menemukan atribut yang harus diprioritaskan dalam memprediksi stunting. Hasil prediksi dengan menggunakan Naive bayes saja pada penelitian ini didapatkan nilai akurasi sebesar 94,3 %, nilai recall sebesar 93,9 % dan nilai precision sebesar 93,93% dengan waktu 0,07 detik Sedangkan dengan menerapkan seleksi fitur Chi square pada penelitian ini diperoleh 5 atribut yang paling berpengaruh terhadap prediksi stunting yang dapat meningkatkan kecepatan pembentukan model Algoritma Naiva Bayes dengan waktu 0,01 detik, namun tidak dapat meningkatkan akurasi, recall dan presisi. 	https://journal.irpi.or.id/index.php/malcom/article/view/1074
7	Prediksi Dampak Pembelajaran Hybrid Learning Menggunakan Naive Bayes	<ul style="list-style-type: none"> Penelitian pada jurnal ini digunakan untuk memprediksi dampak pembelajaran hybrid terhadap mahasiswa Politeknik Negeri Medan Data sampel yang digunakan berasal dari mahasiswa Program Studi Teknologi Rekayasa Perangkat Lunak Politeknik Negeri Medan Hasil prediksi yang dilakukan secara manual dengan naïve bayes, dengan data latihan 100 (seratus) siswa dan data tes 1 (satu) siswa, menghasilkan hasil sebesar 0,012, yang mengindikasikan adanya peningkatan hasil akademik siswa. Hasil pengujian dibuktikan dengan menggunakan bahasa pemrograman python. Hasil tes pertama, dengan data tes 20%, menghasilkan peningkatan hasil akademik sebesar 86% sekitar 13 siswa dengan nilai akurasi 80%, dan tes kedua, dengan data tes 40%, menghasilkan peningkatan hasil akademik sebesar 92% sekitar 29 siswa dengan nilai akurasi 88%. 	https://journal.fkpt.org/index.php/BIT/article/view/968
8	Komparasi Algoritma K-NN, Naive Bayes dan SVM untuk Prediksi Kelulusan Mahasiswa Tingkat Akhir	<ul style="list-style-type: none"> Pada jurnal ini pemilihan dan penentuan data yang digunakan, diambil dari data publik. Dengan 379 orang mahasiswa tahap akhir sebagai responden Pengujian ini membandingkan algoritma K-NN, NBC, dan SVM yang lebih baik menyelesaikan masalah terkait prediksi tingkat kelulusan mahasiswa pascasarjana. Berdasarkan perbandingan algoritma tersebut dengan teknik splitting data, didapatkan bahwa Algoritma K-NN (K-Nearest Neighbor) memiliki rata-rata lebih tinggi dibandingkan (NBC) Naïve Bayes Classifier dan SVM (Support Vector Machine) untuk prediksi kelulusan mahasiswa tingkat akhir dengan akurasi 87,8%, presisi 87,8%, dan recall 84%. 	https://journal.irpi.or.id/index.php/malcom/article/view/610
9	Penerapan metode machine learning -	<ul style="list-style-type: none"> Penelitian ini bertujuan menganalisis sentimen masyarakat terhadap pemindahan ibu kota negara (IKN) 	https://ojs3.unpatti.ac.id/index.php/parameter

	naive bayes pada analisis sentimen pemindahan ibu kota negara baru	<p>pada twitter di tahun 2023 apakah masih menjadi suatu kontroversi atau sudah lebih beraroma positif dibandingkan dengan opini masyarakat pada saat pertama kali isu ini naik ke media sosial di tahun 2019</p> <ul style="list-style-type: none"> • Hasil yang diperoleh dari analisis sentimen terhadap pemindahan ibu kota negara (ikn) baru dengan presentase nilai positif sebesar 55,85% dan sentimen negatif sebesar 45,15 • artinya respon dari masyarakat terhadap isu IKN di media sosial pada tahun 2023 sudah lebih banyak positif menerima pemindahan ibu kota negara dengan segala urgensinya dibandingkan dengan pada tahun 2019 saat Presiden Joko Widodo pertama kali mengumumkan wacana tersebut. • Diperoleh nilai akurasi (accuracy) sebesar 99,12%, nilai akurasi recall untuk hasil negatif yaitu 98,37% dan hasil positif 99,71%. Kemudian untuk nilai akurasi precision untuk pred negatif yaitu 99,63% dan pred positif 99,72% • Dengan demikian maka metode naïve bayes memiliki nilai akurasi yang cukup tinggi 	/article/view/9216
10	Comparison of Naive Bayes, K-Nearest Neighbor, and Support Vector Machine Classification Methods in Semi-Supervised Learning for Sentiment Analysis of Kereta Cepat Jakarta Bandung (KCJB)	<ul style="list-style-type: none"> • Penelitian ini bertujuan untuk membandingkan metode klasifikasi Naïve Bayes, K-Nearest Neighbor (K-NN), dan Support Vector Machine (SVM) dalam mengklasifikasikan sentimen dalam tweet tentang kereta berkecepatan tinggi yang diperoleh dari kumpulan tweet di Twitter • Proses perbandingan dilakukan dengan menggunakan semi-supervised learning, dan hasilnya menunjukkan bahwa model SVM semi-supervised memiliki kinerja terbaik dengan akurasi rata-rata 86%, diikuti oleh model semi-supervised Naïve Bayes dan semi-supervised K-NN dengan akurasi rata-rata masing-masing 81% dan 58%. Secara keseluruhan • hasil prediksi dari ketiga model tersebut menyimpulkan bahwa ada lebih banyak tweet dengan sentimen negatif daripada tweet dengan sentimen positif dan netral 	https://proceedings.stis.ac.id/icdsos/article/view/332

TUGAS PERTEMUAN KE 6

K-NEAREST NEIGHBOR (KNN)

Tugas

1. Buat Model untuk KNN
2. Kelebihan dan Kekurangan KNN
3. Cari 10 Jurnal terkait pemanfaatan KNN
4. Diskusikan dalam Forum
5. Tuliskan dalam laporan (dikumpulkan saat UTS)

Model K-NN

1.1 Data yang akan dianalisa adalah dataset titanic, dengan target kolom survived, memprediksi klasifikasi kelas mana yang membuat penumpang bisa survive

1.2 Berikut code script di python

```
2 from sklearn.preprocessing import LabelEncoder
3 import pandas as pd
4 import numpy as np
5 import matplotlib.pyplot as plt
6 import seaborn as sns
7 from sklearn.model_selection import train_test_split
8 from sklearn.preprocessing import LabelEncoder, StandardScaler
9 from sklearn.neighbors import KNeighborsClassifier
10 from sklearn.metrics import classification_report, confusion_matrix
11
12 # Load dataset from local folder in Colab
13 df = pd.read_csv('passenger titanic dataset.csv') # Ubah
14     'filename.csv' sesuai dengan nama file Anda
15
16 # hilangkan data yang tidak dipakai
17 df = df.drop(['Name', 'Ticket', 'PassengerId', 'Cabin'], axis=1)
18
19 # Mengganti nilai NaN dalam kolom yg terdapat Nan dengan nilai
20     tertentu untuk setiap kolom
21 replacement_values = { 'Fare': df['Fare'].median(), }
22 df.fillna(replacement_values, inplace=True)
23
24 # fungsi replace null pada kolom age
25 def impute_train_age(cols):
26     Age = cols.iloc[0]
27     Pclass = cols.iloc[1]
```

```

26
27     if pd.isnull(Age):
28
29         if Pclass == 1:
30             return 37
31
32         elif Pclass == 2:
33             return 29
34
35         else:
36             return 24
37
38     else:
39         return Age
40
41 df['Age'] = df[['Age', 'Pclass']].apply(impute_train_age,axis=1)
42
43 # pelabelan
44 label = LabelEncoder()
45 data_column = ['Sex', 'Embarked']
46 for column in data_column:
47     df[column] = label.fit_transform(df[column])
48
49 # tentukan target
50 X = df.drop(['Survived'],axis = 1)
51 y = df['Survived']
52
53 # buat pembagian dataset
54 X_train, X_test, y_train, y_test = train_test_split(X, y,
55     test_size=0.2,random_state=42)
56
57 # Training the K-NN model
58 k = 5 # Number of neighbors
59 knn_classifier = KNeighborsClassifier(n_neighbors=k)
60 knn_classifier.fit(X_train, y_train)
61
62 # Making predictions on the test set
63 y_pred = knn_classifier.predict(X_test)
64
65 # Confusion Matrix
66 cm = confusion_matrix(y_test, y_pred)
67 plt.figure(figsize=(10,7))
68 sns.heatmap(cm, annot=True, fmt='d')
69 plt.xlabel('Predicted')
70 plt.ylabel('Actual')
71 plt.title('Confusion Matrix')
72 plt.show()

```

```

72
73 # Classification Report
74 report = classification_report(y_test, y_pred)
75 print("\nClassification Report:\n", report)
76
77 # # Scatter Plot
78 plt.figure(figsize=(10,7))
79 plt.scatter(y_test, y_pred)
80 plt.xlabel('Actual Prices')
81 plt.ylabel('Predicted Prices')
82 plt.title('Actual Prices vs Predicted Prices')
83 plt.show()
84
85 # # Test and Score
86 score = knn_classifier.score(X_test, y_test)
87 print("\nAccuracy:", score)
88

```

1.3 Out put seperti dibawah ini

Classification Report:

	precision	recall	f1-score	support
0	0.69	0.86	0.77	50
1	0.68	0.44	0.54	34
accuracy			0.69	84
macro avg	0.69	0.65	0.65	84
weighted avg	0.69	0.69	0.67	84

Accuracy: 0.6904761904761905

- Precision (Presisi): 0.69

Ini adalah proporsi dari hasil positif yang 69% diidentifikasi dengan benar oleh model dari semua hasil positif yang diprediksi oleh model. Semakin tinggi nilainya, semakin sedikit hasil positif palsu yang dihasilkan oleh model.

- Recall (Recall): 0.86

Ini adalah proporsi dari hasil positif 86% yang diidentifikasi dengan benar oleh model dari semua hasil positif yang sebenarnya dalam data (actual). Semakin tinggi nilainya, semakin banyak hasil positif yang diidentifikasi dengan benar oleh model.

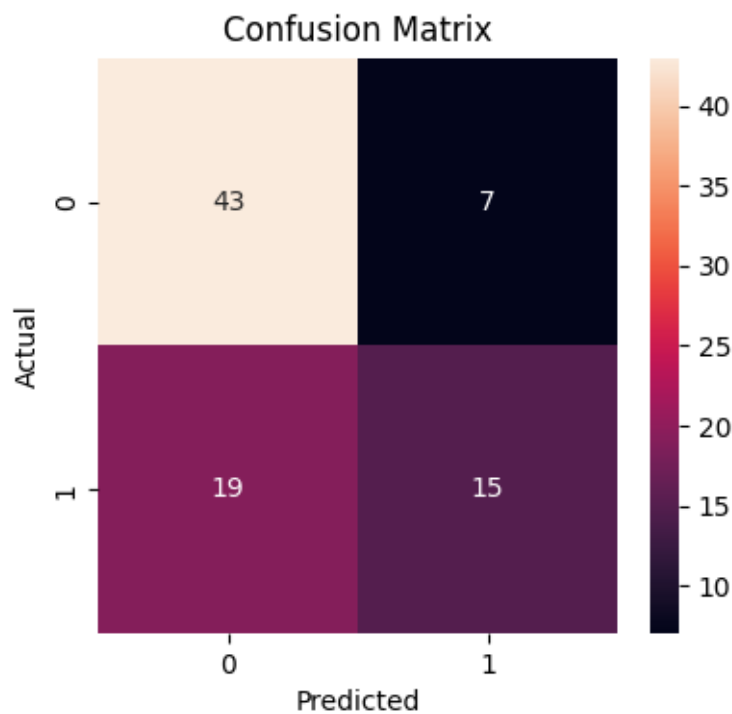
- F1 Score: 0.77

Ini adalah rata-rata harmonik dari precision dan recall. F1 score mencapai nilai terbaik pada 1 dan terburuk pada 0. Semakin tinggi nilainya, semakin baik modelnya dalam memprediksi kelas positif tanpa mengabaikan kelas negatif. Dalam hal ini cukup baik mendekati 1

- Accuracy : 0.69

Ini adalah proporsi total prediksi yang benar yang dilakukan oleh model. Semakin tinggi nilainya, semakin baik kinerja model dalam melakukan klasifikasi secara keseluruhan.

- Confusion matrik yang didapat



SUMMARY JURNAL PENERAPAN K-NEAREST NEIGHBOR (K-NN)

No	Title Jurnal	Summary	Reference
1	An experimental comparison of the nearest-neighbor and nearest-hyperrectangle algorithms	<ul style="list-style-type: none"> dalam jurnal ini menggabungkan karakter berbasis jarak dari pengklasifikasi nearest neighbor (NN) dengan representasi persegi panjang sumbu-paralel yang digunakan dalam banyak sistem pembelajaran aturan mplementasi NGE dibandingkan dengan algoritmak-nearest neighbor (kNN) di 11 domain dan ditemukan secara signifikan lebih rendah daripada kNN di 9 di antaranya Hasil terbaik diperoleh dalam penelitian ini ketika bobot dihitung menggunakan informasi timbal balik antara fitur dan kelas output. Versi terbaik dari NGE yang dikembangkan adalah algoritma batch (BNGE FWMI) yang tidak memiliki parameter yang dapat disetel pengguna. BNGE FWMIKinerja sebanding dengan algoritma tetangga terdekat pertama (juga menggabungkan bobot fitur) Namun, algoritmak-nearest neighbor masih jauh lebih unggul daripada BNGE FWMI di 7 dari 11 domain, dan lebih rendah darinya hanya dalam 2. peneliti menyimpulkan bahwa, bahkan dengan perbaikan kami, pendekatan NGE sangat sensitif terhadap bentuk batas keputusan dalam masalah klasifikasi. Dalam domain di mana batas-batas keputusan adalah sumbu-paralel, pendekatan NGE dapat menghasilkan generalisasi yang sangat baik dengan hipotesis yang dapat ditafsirkan. Di semua domain yang diuji, algoritma NGE membutuhkan lebih sedikit memori untuk menyimpan contoh umum daripada yang dibutuhkan oleh algoritma NN. 	https://link.springer.com/article/10.1007/BF00994658
2	Analisis Sentimen Ulasan Aplikasi PLN Mobile Menggunakan Algoritma Naïve Bayes Classifier dan K-Nearest Neighbor	<ul style="list-style-type: none"> Peneliti dalam jurnal ini melakukan analisis sentimen terhadap ulasan pengguna aplikasi PLN Mobile menggunakan pendekatan text mining. Data dikumpulkan menggunakan teknik scrapping pada Google Play Store dan mendapatkan 3000 baris data Data tersebut kemudian diberi label oleh seorang pakar sehingga menghasilkan 2099 sentimen positif (69,97%), 368 netral (12,27%) dan 533 negatif (17,77%) Selanjutnya dilakukan pemodelan menggunakan algoritma NBC dan KNN dengan K-Fold Cross Validation sebagai teknik validasi. Hasilnya menunjukkan model NBC lebih baik dibandingkan KNN dengan akurasi sebesar 77,69%, recall 53,14%, precision 59,84% dan F1-Score 54,09% Selanjutnya proses analisis dilakukan dengan visualisasi data menggunakan word cloud. Hasilnya yaitu dengan adanya aplikasi PLN Mobile memberikan kemudahan kepada pelanggan dalam menggunakan layanan PLN seperti pembelian token, pengaduan, dan berbagai fitur 	https://journal.irpi.or.id/index.php/malcom/article/view/983

		lainnya. Namun aplikasi PLN Mobile masih memiliki beberapa permasalahan yang sering menjadi ulasan penggunaannya salah satunya adalah saat melakukan pembayaran token.	
3	Implementasi Algoritma K-Nearest Neighbor untuk Prediksi Penjualan Alat Kesehatan pada Media Alkes	<ul style="list-style-type: none"> • Peneliti telah mengelola dan menganalisis data penjualan yang ada untuk memahami kebutuhan pelanggan terhadap Alat Kesehatan • peneliti menggunakan algoritma K-Nearest Neighbor untuk memprediksi penjualan Alat Kesehatan di Media Alat Kesehatan • Informasi mengenai jumlah penjualan Alat Kesehatan dengan kriteria Sangat laris, Cukup laris dan Kurang laris dapat dilihat melalui data penjualan tahun 2020 hingga tahun 2022 pada Media Laporan Penjualan Alat Kesehatan • Penelitian dilakukan dengan menerapkan metode K-Nearest Neighbor (KNN) baik dengan perhitungan secara manual maupun menggunakan sistem RapidMiner • Hasil dari prediksi yang menggunakan sistem RapidMiner menunjukkan tingkat akurasi sebesar 95,00% dari data yang disebut penjualan. Dengan hasil prediksi yang didapat yang Sangat bagus tersebut 	https://journal.irpi.or.id/index.php/malcom/article/view/1326
4	Perbandingan Algoritma K-Nearest Neighbor dan Naïve Bayes Classifier untuk Klasifikasi Status Gizi Pada Balita	<ul style="list-style-type: none"> • Penelitian di jurnal ini bertujuan untuk menguji dan membandingkan performa algoritma K-Nearest Neighbors dan Naïve Bayes Classifier untuk klasifikasi data penimbangan masal balita di Kota Solok. • Nilai akurasi yang diperoleh dari algoritma KNN sebesar 96,24 % sedangkan pada algoritma NBC sebesar 91,00%. 	https://journal.irpi.or.id/index.php/malcom/article/view/474
5	Implementasi Algoritma Naïve Bayes Classifier dan K-Nearest Neighbor untuk Klasifikasi Penyakit Ginjal Kronik	<ul style="list-style-type: none"> • latar belakang di penelitian jurnal ini adalah dari jumlah nilai prelevansi penderita Penyakit Ginjal Kronik di Indonesia yang terbilang besar. Penyakit ginjal kronik ialah kondisi di mana ginjal mengalami penurunan fungsi yang berlangsung dalam jangka waktu yang lama. • Berdasarkan Hasil penelitian yang diperoleh klasifikasi PGK menggunakan algoritma NBC memiliki akurasi sebesar 94,25%, rata-rata nilai recall 94,23%, presisi 98,40% dan AUC 0,961, Sedangkan klasifikasi menggunakan algoritma KNN memiliki akurasi sebesar 77,79%, recall 95,06%, presisi 80,20% dan AUC sebesar 0,627. • Dari kedua hasil menunjukan bahwa klasifikasi menggunakan algoritma NBC lebih baik dibanding menggunakan algoritma KNN. 	https://journal.irpi.or.id/index.php/malcom/article/view/1229

6	Komparasi Algoritma K-NN, Naive Bayes dan SVM untuk Prediksi Kelulusan Mahasiswa Tingkat Akhir	<ul style="list-style-type: none"> • Pada jurnal ini pemilihan dan penentuan data yang digunakan, diambil dari data publik. Dengan 379 orang mahasiswa tahap akhir sebagai responden • Pengujian ini membandingkan algoritma K-NN, NBC, dan SVM yang lebih baik menyelesaikan masalah terkait prediksi tingkat kelulusan mahasiswa pascasarjana. • Berdasarkan perbandingan algoritma tersebut dengan teknik splitting data, didapatkan bahwa Algoritma K-NN (K-Nearest Neighbor) memiliki rata-rata lebih tinggi dibandingkan (NBC) Naïve Bayes Classifier dan SVM (Support Vector Machine) untuk prediksi kelulusan mahasiswa tingkat akhir dengan akurasi 87,8%, presisi 87,8%, dan recall 84%. 	https://journal.irpi.or.id/index.php/malcom/article/view/610
7	Comparison of Naive Bayes, K-Nearest Neighbor, and Support Vector Machine	<ul style="list-style-type: none"> • Penelitian ini bertujuan untuk membandingkan metode klasifikasi Naïve Bayes, K-Nearest Neighbor (K-NN), dan Support Vector Machine (SVM) dalam mengklasifikasikan sentimen dalam tweet tentang kereta berkecepatan tinggi yang diperoleh dari kumpulan tweet di Twitter • Proses perbandingan dilakukan dengan menggunakan semi-supervised learning, dan hasilnya menunjukkan bahwa model SVM semi-supervised memiliki kinerja terbaik dengan akurasi rata-rata 86%, diikuti oleh model semi-supervised Naïve Bayes dan semi-supervised K-NN dengan akurasi rata-rata masing-masing 81% dan 58%. Secara keseluruhan • hasil prediksi dari ketiga model tersebut menyimpulkan bahwa ada lebih banyak tweet dengan sentimen negatif daripada tweet dengan sentimen positif dan netral 	https://proceedings.stis.ac.id/icdsos/article/view/332
8	Penerapan Algoritma K-Nearest Neighbor Menggunakan Wrapper Sebagai Preprocessing untuk Penentuan Keterangan Berat Badan Manusia	<ul style="list-style-type: none"> • Penelitian dalam jurnal ini mengidentifikasi permasalahan dalam penentuan keterangan berat badan dan mencari solusi melalui penggunaan model prediksi. • Algoritma K-NN terpilih karena kemampuannya dalam menangani permasalahan klasifikasi dengan dataset yang kompleks • Metode Wrapper digunakan sebagai langkah preprocessing untuk memilih subset fitur yang paling signifikan • Temuan penelitian menunjukkan bahwa penerapan Algoritma K-NN dengan Wrapper preprocessing dapat meningkatkan akurasi penentuan keterangan berat badan manusia • Penerapan metode K-Nearest Neighbor dan K-Nearest Neighbor dengan Wrapper sebagai tahap preprocessing dalam menentukan keterangan berat manusia mendapatkan hasil nilai akurasi yang sama yaitu sebesar 91%. 	https://journal.irpi.or.id/index.php/malcom/article/view/1085

9	Perbandingan Klasifikasi Antara Naive Bayes dan K-Nearest Neighbor Terhadap Resiko Diabetes pada Ibu Hamil	<ul style="list-style-type: none"> • Penelitian di jurnal ini data yang digunakan adalah kemungkinan resiko ibu hamil terkena diabetes, data tersebut diolah memakai teknik data mining yaitu naïve bayes • Setelah dilakukan pengolahan data memakai teknik klasifikasi data mining algoritma naïve bayes, hasil yang didapat untuk pembagian data menggunakan K-Fold Cross Validation K=10 pada algoritma naïve bayes didapat lah hasil 75,78% dan untuk pengolahan menggunakan knn dengan nilai K=25 didapat hasil 74,48% • Dari hasil tersebut naïve bayes lebih baik dibandingkan K-Nearest Neighbor (KNN). 	https://journal.irpi.or.id/index.php/malcom/article/view/432
10	Perbandingan Algoritma Linear Regression, Neural Network, Deep Learning, Dan K-Nearest Neighbor (K-Nn) Untuk Prediksi Harga Bitcoin	<ul style="list-style-type: none"> • Penelitian dalam jurnal ini bertujuan untuk membandingkan algoritma yang digunakan untuk memprediksi harga Bitcoin • Dalam penelitian ini akan dilakukan prediksi terhadap harga Bitcoin dengan membandingkan empat model algoritma yaitu Linear Regression, Neural Network, Deep Learning, dan k-Nearest Neighbor (k-NN) • Tingkat akurasi dari tiap model algoritma akan diuji dengan metode validasi K-Fold Cross Validation dan dievaluasi menggunakan Root Mean Square Error (RMSE). Hasil dengan uji T-Test dalam penelitian ini menyimpulkan bahwa model terbaik untuk memprediksi harga Bitcoin adalah model algoritma Linear Regression dan Neural Network, yaitu dengan hasil RMSE 296.227 +/- 60.125 (micro average: 301.655 +/- 0.000) dan 338.988 +/- 47.837 (micro average: 342.000 +/- 0.000). 	https://ejournal.unsri.ac.id/index.php/jsi/article/view/16561

2. Kelebihan dan Kekurangan KNN

Metode K-Nearest Neighbors (KNN) adalah salah satu algoritma pembelajaran mesin yang

sederhana dan intuitif. Di bawah ini adalah kelebihan dan kekurangan dari algoritma KNN:

Kelebihan KNN:

1. Sederhana dan Mudah Dipahami:

KNN adalah algoritma yang mudah dipahami dan diterapkan. Konsep dasarnya relatif sederhana,

di mana prediksi dilakukan dengan membandingkan dengan tetangga terdekat.

2. Non-Parametrik:

KNN adalah algoritma non-parametrik yang berarti ia tidak membuat asumsi tertentu tentang

distribusi data. Ini membuatnya cocok untuk berbagai jenis data.

Tugas Data Mining Pascasarjana Universitas Pamulang [Iwan Muttaqin,S.Kom]-
Page | 41

3. Tidak Memerlukan Proses Training yang Panjang:

KNN adalah algoritma instance-based, yang berarti tidak memerlukan proses pelatihan yang

panjang karena hanya menyimpan data latihan. Ini membuatnya cepat dalam mengadopsi

perubahan dalam data.

4. Mampu Menangani Data Nonlinear dan Multikelas:

KNN tidak mengandalkan asumsi linearitas, sehingga mampu menangani data yang kompleks atau

nonlinear dengan baik. Selain itu, dapat dengan mudah diperluas untuk masalah klasifikasi dengan

lebih dari dua kelas.

5. Kinerja yang Baik dengan Data Terstruktur:

KNN cenderung berkinerja baik ketika data terstruktur dengan baik dan distribusi kelas yang relatif

seragam di seluruh ruang fitur.

Kekurangan KNN:

1. Sensitif terhadap Outlier:

KNN sangat sensitif terhadap data outlier karena prediksinya sangat dipengaruhi oleh tetangga

terdekat. Outlier dapat menyebabkan perubahan signifikan dalam hasil prediksi.

2. Perhitungan yang Mahal:

KNN memerlukan perhitungan jarak antara setiap titik data dalam ruang fitur. Ini dapat menjadi

mahal secara komputasi, terutama untuk dataset besar atau dengan banyak fitur.

3. Membutuhkan Penyesuaian Parameter:

Pemilihan parameter K dalam KNN penting. Nilai K yang terlalu kecil dapat menyebabkan model

menjadi rentan terhadap noise, sedangkan nilai K yang terlalu besar dapat menyebabkan model

menjadi terlalu umum.

4. Tidak Efisien pada Dimensi Tinggi:

Kinerja KNN menurun secara signifikan dengan peningkatan dimensi fitur. Ini disebabkan oleh

"Kerumitan Dimensi Tinggi", di mana ruang fitur menjadi semakin kosong dengan meningkatnya

dimensi, yang membuat jarak antara titik-titik data kurang bermakna.

5. Membutuhkan Data Terstruktur dengan Baik:

KNN cenderung tidak berkinerja baik dengan data yang memiliki banyak variabel yang tidak

relevan atau tidak memiliki pola yang jelas. Ini karena prediksi KNN sangat bergantung pada kedekatan antar tetangga.

PERTEMUAN KE 7

Tugas

Cari Kapan menggunakan f1 score,accuracy,presisi?

kapan menggunakan semuanya

1. jika jumlah data setiap kelas relatif berimbang (balanced)gunakan akurasi
2. jika jumlah data setiap kelas tidak berimbang (imbalanced) gunakan ukuran f1-score
3. jika false positif (fp) lebih baik terjadi daripada false negatif (fn),maka gunakan recall, contoh kasus deteksi spam
4. jika true positif (tp) lebih baik terjadi daripada false positif (fp) maka gunakan presisi (precision)
5. jika tidak menginginkan terjadinya false positif(fp) maka gunakan specificity