1. Model Decision Tree: Struktur, Pemisahan, dan Perbandingan

a. Bagaimana Struktur Decision Tree Bekerja dalam Memisahkan Data Berdasarkan Fitur (Entropy dan Information Gain)

Decision Tree adalah model pembelajaran Supervised Learning yang digunakan untuk klasifikasi dan regresi. Model ini memisahkan data dengan memilih fitur yang memberikan perbedaan paling besar antara kelaskelas yang ada. Ide utama dalam Decision Tree adalah memaksimalkan homogenitas data di setiap pemisahan (node) hingga setiap node akhir (leaf) berisi data dari satu kelas saja.

Untuk menentukan pemisahan, Decision Tree menggunakan metrik seperti Entropy dan Information Gain:

- Entropy mengukur ketidakmurnian atau ketidakteraturan dalam suatu dataset. Dataset dengan kelas campuran memiliki entropy tinggi, sedangkan yang hanya berisi satu kelas memiliki entropy rendah.

$$\boldsymbol{Entropy} = \boldsymbol{-\Sigma p_i \log_2(p_i)}$$

di mana p_i adalah probabilitas untuk kelas i.

- Information Gain mengukur pengurangan entropy ketika dataset dibagi berdasarkan fitur tertentu. Information Gain membantu memilih fitur terbaik untuk pemisahan.

$$Information\ Gain = Entropy\ (Parent) - \frac{(Jumlah\ Sampel\ Child)}{(Jumlah\ Sampel\ Parent)}\ x\ Entropy\ (\ Child)$$

Contoh: Misalkan kita ingin mengklasifikasikan apakah seseorang akan membeli produk atau tidak berdasarkan umur. Kita biasanya menemukan kasus bahwa pemisahan pada ambang batas usia tertentu mengurangi entropy lebih banyak dibanding ambang batas lainnya, maka ambang batas usia ini dipilih sebagai node.

b. Perbandingan dengan Linear Regression, Logistic Regression, dan SVM

Model	**Kelebihan**	**Kekurangan**	**Contoh Kasus**
Decision Tree	Mudah diinterpretasi; mampu menangani data non-linear; dapat menangani data kategorikal.	Rentan terhadap overfitting; bisa menghasilkan model yang kompleks.	Cocok saat interpretabilitas penting, misalnya pada penilaian risiko kredit berdasarkan pendapatan.
Linear Regression	Sederhana dan efisien untuk data kontinu; mudah diinterpretasi.	Asumsi linearitas; performa kurang pada data non-linear.	Prediksi nilai kontinu seperti harga rumah.

Logistic Regression	Baik untuk klasifikasi biner; output probabilistik.	Terbatas pada batas linear; kurang interpretatif pada data kompleks.	Prediksi kemungkinan pembelian pelanggan terhadap suatu produk.
SVM	Efektif di ruang berdimensi tinggi; tahan terhadap overfitting dengan parameter yang tepat.	Sulit diinterpretasi; waktu pelatihan lambat untuk dataset besar.	Klasifikasi gambar dengan pemisahan kelas yang jelas.

Kasus: Decision Tree mungkin lebih sesuai dalam skenario di mana pola nonlinear kompleks ada (misalnya, pembobotan kredit dengan banyak kondisi percabangan) dan interpretabilitas model sangat penting.

2. Membangun Model untuk Memprediksi Pembelian Produk

a. Model yang Diusulkan adalah Logistic Regression

Memprediksi kemungkinan seseorang membeli produk, Logistic Regression lebih sesuai dibandingkan Linear Regression. Logistic Regression dirancang untuk klasifikasi biner dan menghasilkan probabilitas, sehingga ideal untuk hasil ya atau tidak karena pada kasus kemungkinan seseorang akan membeli produk berdasarkan data historis hanya akan menghasilkan keluaran berupa nilai kelas 1 misalnya berarti "orang tersebut membeli produk A" atau nilai kelas 0 misalnya berarti "orang tersebut tidak membeli produk A". Kemudian yang menjadi factor X pendukung kenapa seseorang jadi membeli suatu produk tidak selamanya berupa data angka seperti budget yang diinginkan dan uang tersedia bisa juga factor lain seperti selera, warna, bentuk, kegunaan dan pertimbangan lainnya yang menghasilkan nilai kategoris dan hanya akan bisa dibaca menjadi data biner agar kita bisa menghasilkan prediksi yang sesuai.

b. Persiapan Data dan Metrik Evaluasi

- Persiapan Data:
  Muat data bisa berupa csv maupun excel worksheet. Bersihkan data, pastikan fitur-fitur dinormalisasi atau distandardisasi jika diperlukan. Inspeksi dataset dari data redundant, data null, ketidakseimbangan data, data yang beragam tipenya dan analisa statistik deskriptif dari dataset.

- Data Preprocessing:
  Lakukan one hot encoding dataset (encode fitur- fitur kategorikal), undersampling, split dataset (bagi data ke dalam set pelatihan dan pengujian untuk mengevaluasi generalisasi model) dan mengetahui ukuran data training dan test.

- Metrik Evaluasi:
  Confusion Matrix, Accuracy Score, Class Report dan Visualisasi Confusion Matrix.

3. Apa itu Overfitting dalam Machine Learning?

Overfitting terjadi ketika model mempelajari data pelatihan terlalu detail, termasuk gangguan dan pola spesifik yang tidak akan muncul pada data baru. Hal ini menyebabkan akurasi tinggi pada set pelatihan tetapi performa yang buruk pada set pengujian. Model bisa jadi terlalu mengingat pola saat diberikan dataset pelatihan atau bisa disebut terlalu overthinking dan perfeksionis. Padahal saat diberikan data uji model hanya mengingat pola pelatihan dan akan menjadi salah dalam menyelesaikan data uji yang bisa jadi data dan polanya berbeda dari data pelatihan.

Mendeteksi overfitting:

- Cara paling mudah adalah setelah membagi dataset menjadi training dan test kemudian kita memodelkan dan memprediksi maka bis akita lihat perbandingan nilai hasilnya. Model yang mengalami overfitting biasanya akan menunjukkan kinerja yang sangat baik pada data pelatihan tetapi buruk pada data pengujian atau hanya berkinerja ideal pada salah satu kelasnya saja. Nilai yangkita bandingkan pada metrik evaluasi adalah seperti akurasi, precision, recall, atau F1 score antara data pelatihan dan data pengujian.
- CrossValidation: Gunakan crossvalidation untuk melihat apakah performa model menurun pada fold yang tidak terlihat.
- Perhitungkan MAD.
- Learning Curves: Buat grafik kesalahan pelatihan dan validasi jika kesalahan pelatihan rendah dan kesalahan validasi tinggi, kemungkinan terjadi overfitting.

4. Mengapa Overfitting Menjadi Masalah dalam Decision Tree, dan Bagaimana Mengatasinya?

Decision Tree rentan terhadap overfitting karena model ini terus membagi hingga setiap leaf menjadi benar-benar nilai tunggal, sehingga menangkap gangguan selain sinyal. Cara untuk mengatasi overfitting:

- Pruning (Pemangkasan): Batasi kedalaman tree atau hapus node dengan information gain minimal.
- Minimum Samples per Leaf: Tetapkan jumlah sampel minimum pada setiap leaf untuk mengurangi peluang overfitting pada data kecil.
- Ensemble Methods: Gunakan metode seperti Random Forest, yang mengurangi overfitting dengan rata-rata dari beberapa Decision Tree, atau Gradient Boosting, yang membangun tree secara berurutan untuk meminimalkan kesalahan.

5. Dua Fungsi Kernel yang Sering Digunakan dalam SVM

Support Vector Machines (SVM) menggunakan fungsi kernel untuk mentransformasikan data ke dalam dimensi yang lebih tinggi sehingga dapat dipisahkan secara linear. Dua fungsi kernel yang sering digunakan adalah:

- Linear Kernel:

Sederhana dan cocok untuk data yang dapat dipisahkan secara linear. Sesuai untuk kasus di mana data memiliki sedikit fitur dan hampir dapat dipisahkan secara linear.

$$K(x, \quad x') = x \,.\, x'$$

- Radial Basis Function (RBF) Kernel:

Cocok untuk data nonlinear karena dapat mentransformasi data ke ruang berdimensi lebih tinggi. Efektif untuk data dengan hubungan kompleks yang tidak dapat dipisahkan secara linear.

$$K(x, \quad x') = \exp(-\gamma \, ||x - x'| \,|^2)$$

Masing-masing dari kernel memiliki keunggulan tergantung pada dataset dan masalah yang dihadapi, serta memungkinkan SVM untuk mencapai fleksibilitas dalam mengklasifikasikan data linear dan nonlinear.