Analisis Komponen Utama PCA

PCA digunakan untuk merangkum data yang kompleks menjadi beberapa dimensi yang lebih sederhana, sehingga kita dapat memahami variabilitas dalam data dengan lebih baik. Dalam PCA, kita mencari arah utama variabilitas (komponen utama) dan menggambarkannya ke dalam grafik dua atau tiga dimensi untuk memahami hubungan antar data atau kelompok data. PCA juga dapat digunakan untuk mengidentifikasi gen-gen yang berperan dalam perbedaan antar kelompok data. Ada beberapa teknik diagnostik untuk memastikan keberhasilan PCA, seperti scree plot. Jadi, PCA adalah alat yang bermanfaat untuk analisis data yang kompleks.

K-Nearest Neighbor (KNN)

Dalam K-NN, data yang sudah ada dengan kategori diklasifikasikan berdasarkan kedekatan dengan tetangga terdekat. Nilai K menentukan berapa banyak tetangga yang digunakan dalam klasifikasi, dan kategori yang paling banyak muncul di antara tetangga tersebut akan menjadi kategori yang diberikan pada data yang tidak diketahui kategorinya.

Algoritma K-NN dapat digunakan untuk mengklasifikasikan data berdasarkan kedekatan dengan data pelatihan yang sudah diketahui kategorinya. Penentuan nilai K yang optimal bisa melibatkan percobaan dan penyesuaian, karena tidak ada aturan baku untuk memilih nilai K yang paling tepat. K-NN cocok digunakan untuk data dengan jumlah sampel yang cukup besar, tetapi harus dipilih dengan hati-hati agar tidak terlalu besar sehingga kategori minoritas selalu terabaikan.

Decision and Classification Trees

Tulisan di atas menjelaskan mengenai decision trees (pohon keputusan) dalam konteks klasifikasi data. Decision tree adalah algoritma yang digunakan untuk mengklasifikasikan data berdasarkan serangkaian pertanyaan yang dibuat berdasarkan atribut data. Berikut beberapa poin penting yang dijelaskan:

1. Decision Tree: Decision tree adalah model yang digunakan untuk membuat keputusan berdasarkan serangkaian pertanyaan. Model ini dapat digunakan untuk klasifikasi data (classification trees) atau prediksi nilai numerik (regression trees).

2. Struktur Decision Tree: Decision tree memiliki struktur berhiaskan cabang dan daun. Cabang-cabangnya mewakili pertanyaan atau keputusan, sementara daun-daunnya mewakili kategori atau nilai output.

3. Pembuatan Decision Tree: Pembuatan decision tree dimulai dengan memilih atribut yang paling baik untuk digunakan sebagai pertanyaan di akar pohon. Ini melibatkan perhitungan impurity (seperti genie impurity) untuk setiap kandidat atribut. Atribut dengan impurity terendah akan dipilih.

4. Impurity: Impurity adalah ukuran sejauh mana suatu daun (leaf) dalam decision tree berisi campuran data dari kategori yang berbeda. Tujuannya adalah untuk meminimalkan impurity sehingga daun memiliki kemurnian yang tinggi.

5. Pruning: Untuk menghindari overfitting, kita dapat melakukan pruning pada decision tree. Pruning melibatkan pemangkasan cabang-cabang yang tidak signifikan atau mengharuskan minimal jumlah data dalam setiap daun.

6. Cross-Validation: Untuk menentukan parameter seperti jumlah data minimal dalam setiap daun atau atribut yang paling baik digunakan, cross-validation digunakan untuk menguji kinerja model decision tree dengan berbagai konfigurasi.