ISC UAS DATMIN 2024

RANGKUMAN

Perbedaan Supervised dan Unsupervised Learning

- Supervised learning adalah suatu pendekatan dalam pembuatan AI. Disebut "supervised" karena dalam pendekatan ini, machine learning dilatih untuk mengenali pola antara input data dan label output. Jadi, pada supervised learning, terdapat data training dan data testing yang sudah diberikan label.
- Unsupervised learning adalah suatu teknik yang digunakan machine learning dalam pembuatan artificial intelligence. Dalam pendekatan ini, algoritma komputer tidak perlu dilatih untuk mengenali pola penyusun AI. Model dirancang untuk bisa "belajar mandiri" dalam mengumpulkan informasi. Jadi, pada unsupervised learning, data tidak diberikan label.

Perbedaan Klasifikasi dan Clustering

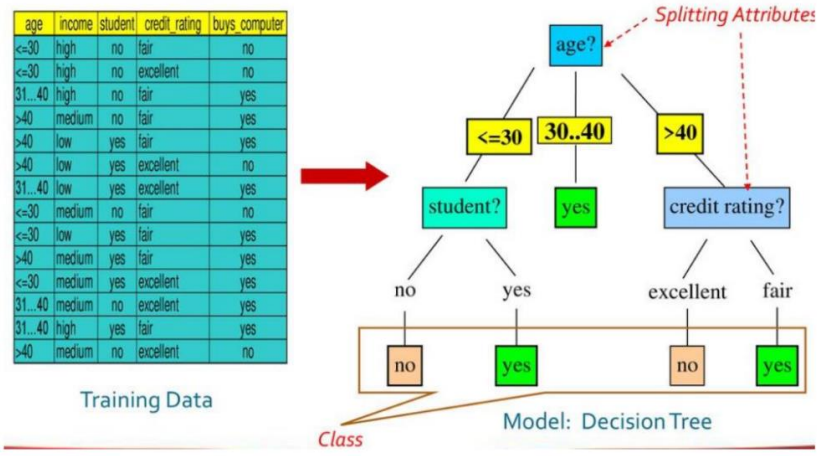
Klasifikasi	Clustering
• Menggunakan pendekatan Supervised Learning • Memerlukan data training dan data testing • Contoh : Naive Bayes, KNN (K-Nearest Neighbors), Logistic Regression, SVM (Support Vector Machine), dll	• Menggunakan pendekatan Unsupervised Learning • Tidak memerlukan data training dan data testing • Contoh : K-Means Clustering, Fuzzy C-Means Clustering, dll

Perbedaan Data Training & Data Testing

- Data training adalah bagian dataset yang kita gunakan untuk melatih algoritma untuk membuat prediksi atau menjalankan fungsinya sesuai tujuannya masing-masing.
- Data testing adalah bagian dataset yang kita tes untuk melihat keakuratan algoritma.

Decision Tree

Decision Tree merupakan salah satu cara data processing dalam memprediksi masa depan dengan cara membangun klasifikasi atau regresi model dalam bentuk struktur pohon. Hal tersebut dilakukan dengan cara memecah terus ke dalam himpunan bagian yang lebih kecil lalu pada saat itu juga sebuah pohon keputusan secara bertahap dikembangkan. Hasil akhir dari proses tersebut adalah pohon dengan node keputusan dan node daun.



Gambar 1. Contoh Decision Tree

Algoritma C4.5

Algoritma C.45 adalah salah satu metode algoritma klasifikasi atau pengelompokan pada dataset. Dasar dari algoritma C4.5 adalah pembentukan pohon keputusan (Decision Tree).

Langkah-langkah:

1. Hitung nilai entropy dan gain dari seluruh atribut
   Rumus Entropy

$$\text{Entropy}(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$$

Gambar 2. Rumus Entropy C4.5

   Rumus Gain

$$\text{Gain}(S,A) = \text{Entrophy}(s) - \sum_{i=1}^{n} \frac{|si|}{|s|} * \text{Entrophy}(si)$$

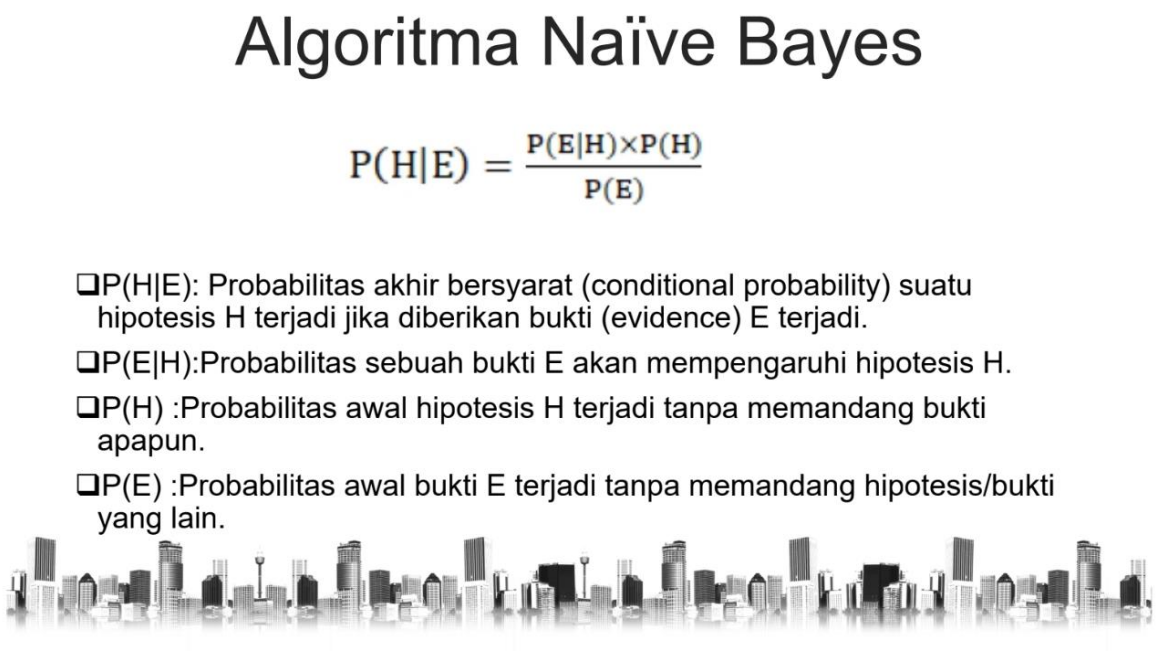
Gambar 3. Rumus Gain C4.5

2. Pilih atribut dengan nilai gain tertinggi sebagai akar percabangan
3. Buat cabang untuk masing-masing nilai dari atribut terpilih
4. Apabila cabang belum jelas outputnya, maka lakukan pemecahan
5. Untuk melakukan pemecahan, ulangi langkah 1 hingga 3 pada kasus cabang, tetapi kali ini hitung nilai entropy dan gain nya berdasarkan kasus cabang tersebut

Naive Bayes

Naive Bayes Classifier adalah salah satu algoritma machine learning yang digunakan untuk klasifikasi. Algoritma ini didasarkan pada Teorema Bayes dan mengasumsikan bahwa atribut-atribut dalam

dataset saling independen (independen bersyarat). Asumsi yang sangat kuat (naïf) akan independensi dari masing-masing kondisi/kejadian merupakan ciri utama dari Naive Bayes.



Gambar 4. Rumus Naive Bayes

Langkah-langkah:

1. Hitung probabilitas masing-masing output
2. Hitung probabilitas kondisional
3. Lakukan perhitungan algoritma naive bayes pada data testing untuk masing-masing output
4. Bandingkan probabilitas output, pilih kelas dengan probabilitas tertinggi

K-Nearest Neighbors

K-NN adalah algoritma berbasis instance (instance-based learning), yang berarti bahwa algoritma ini tidak membuat asumsi eksplisit tentang distribusi data, melainkan menyimpan semua contoh pelatihan dan membuat keputusan berdasarkan kemiripan dengan contoh-contoh ini. Algoritma ini mencari k tetangga terdekat (nearest neighbors) dari sebuah instance baru dan menggunakan informasi dari tetangga-tetangga ini untuk melakukan prediksi.

KNN tidak memiliki fase khusus training. Training dilakukan untuk setiap data testing. Oleh karena itu, algoritma ini terkenal lambat.

Langkah-langkah:

1. Tentukan nilai K (jumlah tetangga terdekat)
2. Untuk sebuah data testing, hitung jarak euclidean pada setiap data training
3. Urutkan semua data berdasarkan jarak euclidean terkecil hingga terbesar
4. Eliminasi data sesuai dengan nilai K

5. Klasifikasi ditentukan dengan target terbanyak yang ada pada data hasil eliminasi

$$d = \sqrt{\sum(X_1 - X_2)^2}$$

- d= jarak Euclidean
- X1 = data training
- X2 = data testing

Gambar 5. Rumus Jarak Euclidean

K-Means Clustering

K-means clustering adalah algoritma pembelajaran unsupervised learning yang digunakan untuk mengelompokkan data ke dalam k kelompok (klaster) yang berbeda berdasarkan kemiripan atribut mereka.

Langkah-langkah:

1. Tentukan jumlah cluster (k)
2. Pilih titik centroid awal secara acak
3. Menetapkan setiap data ke cluster terdekat berdasarkan jarak euclidean dengan nilai atribut sebagai x1 dan nilai centroid sebagai x2 (apabila belum memenuhi batas iterasi atau minimum SSE, lanjut ke langkah 4)
4. Hitung ulang centroid berdasarkan rata-rata dari semua titik dalam setiap cluster
5. Ulangi proses 3-4 sampai batas iterasi atau minimum SSE
   *SSE = Sum Squared Error

LATIHAN SOAL

1. Terdapat data sebagai berikut :

ID	Jenis Kelamin	Pendapatan	Tanggungan Keluarga	Kategori
1	PEREMPUAN	RENDAH	1	YA
2	PEREMPUAN	TINGGI	1	TIDAK
3	LAKI-LAKI	RENDAH	2	YA
4	PEREMPUAN	RENDAH	3	YA
5	PEREMPUAN	TINGGI	3	TIDAK
6	LAKI-LAKI	TINGGI	1	TIDAK
7	PEREMPUAN	RENDAH	2	YA
8	LAKI-LAKI	TINGGI	3	TIDAK
9	PEREMPUAN	RENDAH	2	TIDAK
10	LAKI-LAKI	RENDAH	2	YA

Apabila terdapat data dengan ID = 11, Jenis Kelamin = PEREMPUAN, Pendapatan=TINGGI, Tanggungan Keluarga = 2.
Tentukan prediksi untuk klasifikasi data ID 11 tersebut dengan menggunakan :

1. Naive Bayes
2. KNN dengan K = 3

3. Jika terdapat data sebagai berikut :

Jumlah Anak	Jumlah Kendaraan
3	4
1	1
4	3
2	2
5	3

Lakukan clustering dengan menggunakan K-Means sejumlah 2 cluster dan 2 iterasi.

Jawaban

1. Naive bayes

   Probabilitas Output
   P(Ya) : 5/10 : 0,5
   P(Tidak) : 5/10 : 0,5

   Probabilitas Kondisional
   Jenis Kelamin
   P(Perempuan|Ya) : 3/6 = 0,5
   P(Perempuan|Tidak) : 3/6 = 0,5
   P(Laki-Laki|Ya) : 2/4 = 0,5
   P(Laki-Laki|Tidak) : 2/4 = 0,5

   Pendapatan
   P(Tinggi|Ya) : 0
   P(Tinggi|Tidak) : 4/4 = 1
   P(Rendah|Ya) : 5/6 = 0,833
   P(Rendah|Tidak) : 1/6 = 0,167

   Tanggungan Keluarga
   P(1|Ya) : 1/3 = 0,33
   P(1|Tidak) : 2/3 = 0,67
   P(2|Ya) : ¾ = 0,75
   P(2|Tidak) : ¼ = 0,25
   P(3|Ya) : 1/3 = 0,33
   P(3|Tidak) : 2/3 = 0,67

   Implementasi Algoritma
   Data X
   Jenis Kelamin = Perempuan
   Pendapatan = Tinggi
   Tanggungan Keluarga = 2

   - P(x1,x2...|Ya) x P(Ya) / ~~P(x1,x2,...)~~
   - P(x1,x2...|Tidak) x P(Tidak) / ~~P(x1,x2...)~~

   P(Ya|X) = P(Perempuan|Ya) x P(Tinggi|Ya) x P(2|Ya) x P(Ya)
   = 0,5 x 0 x 0,75 x 0,5
   = 0

   P(Tidak|X) = P(Perempuan|Tidak) x P(Tinggi|Tidak) x P(2|Tidak) x P(Tidak)
   = 0,5 x 1 x 0,25 x 0,5
   = 0,0625

   Setelah dibandingkan, hasilnya adalah P(Tidak|X) > P(Ya|X). Jadi, data X terklasifikasi sebagai kategori Tidak.

2. KNN
   K = 3

   Kita anggap,
   Perempuan = 1
   Laki-laki = 2

   Pendapatan Tinggi = 1
   Pendapatan Rendah = 2

   Tanggungan keluarga 1,2,3

   Data X
   Jenis Kelamin = Perempuan = 1
   Pendapatan = Tinggi = 1
   Tanggungan Keluarga = 2

   Hitung jarak euclidean untuk setiap data training

Data Ke-	Jarak Euclidean
1	√ (1-1)^2+(2-1)^2+(1-2)^2 = 1.414213562
2	√ (1-1)^2+(1-1)^2+(1-2)^2 = 1
3	√ (2-1)^2+(2-1)^2+(2-2)^2 = 1.414213562
4	√ (1-1)^2+(2-1)^2+(3-2)^2 = 1.414213562
5	√ (1-1)^2+(1-1)^2+(3-2)^2 = 1
6	√ (2-1)^2+(1-1)^2+(1-2)^2 = 1.414213562
7	√ (1-1)^2+(2-1)^2+(2-2)^2 = 1
8	√ (2-1)^2+(1-1)^2+(3-2)^2 = 1.414213562
9	√ (1-1)^2+(2-1)^2+(2-2)^2 = 1
10	√ (2-1)^2+(2-1)^2+(2-2)^2 = 1.414213562

   Urutkan dari jarak terkecil ke terbesar
   Jarak terkecil = 1
   Data yang memiliki jarak 1 = 2, 5, 7, 9
   Data yang diambil = 2,5,7 (secara acak)

   Data ke 2 = Tidak
   Data ke 5 = Tidak
   Data ke 7 = Ya

   Jadi, data X terklasifikasikan sebagai Tidak.

3. K-means
   K = 2
   Batas iterasi = 2

   Karena kluster ada 2, maka titik centroid juga ada 2.

   C1 = (1,1)

C2 = (5,3)

Hitung Jarak

Data Ke-	Jarak dengan C1	Jarak dengan C2
1	√ (3-1)^2+(4-1)^2 = 3.605551275	√ (3-5)^2+(4-3)^2 = 2.236067977
2	√ (1-1)^2+(1-1)^2 = 0	√ (1-5)^2+(1-3)^2 = 4.472135955
3	√ (4-1)^2+(3-1)^2 = 3.605551275	√ (4-5)^2+(3-3)^2 = 1
4	√ (2-1)^2+(2-1)^2 = 1.414213562	√ (2-5)^2+(2-3)^2 = 3.16227766
5	√ (5-1)^2+(3-1)^2 = 4.472135955	√ (5-5)^2+(3-3)^2 = 0

Untuk iterasi ke-2, kita hitung lagi centroid nya berdasarkan nilai rata-rata dari setiap titik pada kluster.

Kluster 1 : Data ke-2, 4

Data Ke-	Jumlah Anak	Jumlah Kendaraan
2	1	1
4	2	2

Kluster 2 : Data ke-1, 3, 5

Data Ke-	Jumlah Anak	Jumlah Kendaraan
1	3	4
3	4	3
5	5	3

Hitung kembali C1, dan C2

C1

⇨ Jml Anak = (1+2)/2 = 1,5

⇨ Jml Kendaraan = (1+2)/2 = 1,5

⇨ Jadi, C1 = (1,5 dan 1,5)

C2

⇨ Jml Anak = (3+4+5)/3 = 4

⇨ Jml Kendaraan = (4+3+3)/3 = 3,33

⇨ Jadi, C2 = (4 dan 3,33)

Hitung Jarak

Data Ke-	Jarak dengan C1	Jarak dengan C2
1	√ (3-1,5)^2+(4-1,5)^2 = 2.915475947	√ (3-4)^2+(4-3,33)^2 = 1.203702621
2	√ (1-1,5)^2+(1-1,5)^2 = 0.7071067812	√ (1-4)^2+(1-3,33)^2 = 3.798539193
3	√ (4-1,5)^2+(3-1,5)^2 = 2.915475947	√ (4-4)^2+(3-3,33)^2 = 0,33
4	√ (2-1,5)^2+(2-1,5)^2 = 0.7071067812	√ (2-4)^2+(2-3,33)^2 = 2.401853451
5	√ (5-1,5)^2+(3-1,5)^2 = 3.807886553	√ (5-4)^2+(3-3,33)^2 = 1.053043209

Karena iterasi sudah 2 (sudah mencapai batas iterasi), maka proses diberhetinkan.

Jadi, Data ke-2 dan ke-4 masuk sebagai kluster 1, sedangkan data ke-1, ke-3, dan ke-5 masuk sebagai kluster 2.