



Data Mining

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

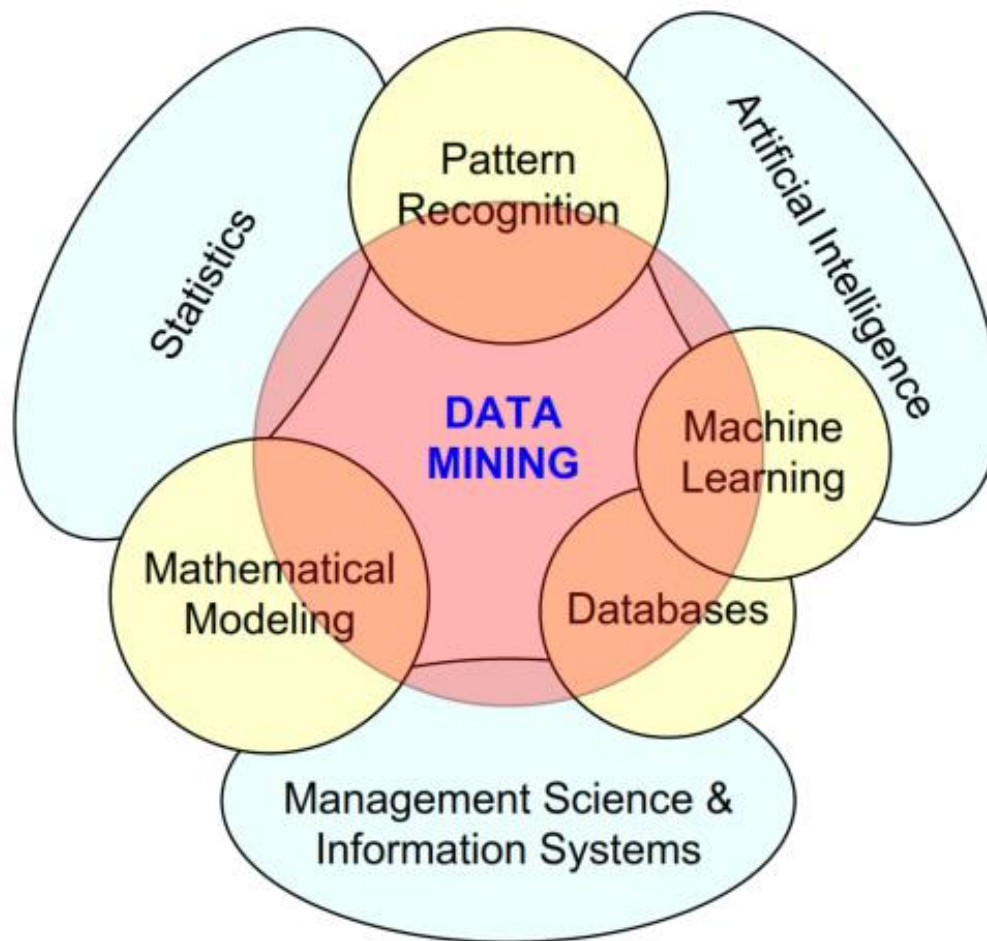
Copyright © Yudi Agusta, PhD 2024

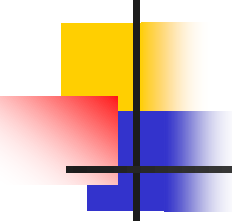


Definisi Data Mining

- Pencarian informasi prediksi yang tersembunyi di dalam database yang besar
- i.e. Knowledge Discovery/Penemuan Pengetahuan
- Suatu ilmu yang baru yang berada di antara statistik, teknologi database, pengenalan pola, machine learning, dan berfokus pada analisa tahap kedua dari database yang besar untuk menemukan hubungan yang tidak terpikirkan sebelumnya, yang menjadi interest dari pemilik data - *Hand, American Statistician, 1998.*

Data Mining





Sistem Informasi, Data Warehousing, dan Data Mining

- **Sistem Informasi** merupakan suatu sistem yang menyimpan data yang bersifat operasional
- **Data Warehousing** merupakan konsep yang datanya ditransformasi (ETL) dari data di dalam Sistem Informasi yang bersifat operasional/transaksional
 - Data Warehousing merupakan konsep yang **dipasangkan** dengan Sistem Informasi untuk tujuan pelaporan/pembuatan laporan
- **Data Mining** merupakan konsep yang memanfaatkan data yang ada di dalam Sistem Informasi yang bersifat operasional atau Data Warehouse yang bersifat pelaporan/summary
 - Data Mining bertujuan untuk membentuk model yang digunakan untuk melakukan prediksi kondisi masa yang akan datang



Data Mining, Machine Learning & Statistics

- Data Mining merupakan istilah yang digunakan untuk **pemodelan** yang dilakukan terhadap **data**
- Machine Learning **juga** merupakan pemodelan terhadap **data**
- Beberapa metode yang digunakan dalam pengembangan software package data mining **terkelompok di dalam** Machine Learning
- Beberapa metode yang terkelompok di dalam Machine Learning **dikembangkan** menggunakan Statistik



Data Mining Dalam Kecerdasan Bisnis

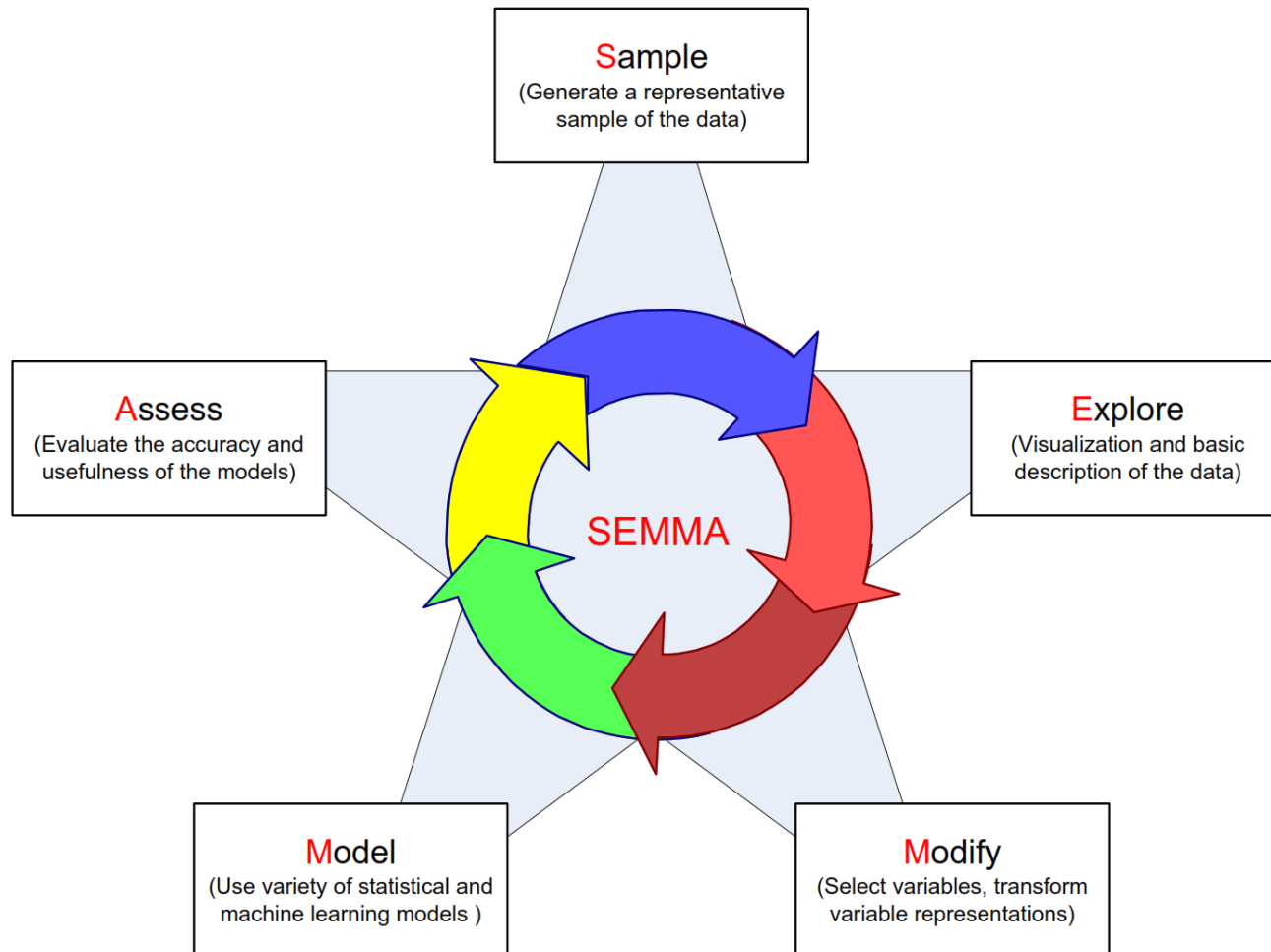
- Data di dalam suatu perusahaan sudah disediakan sesuai dengan kebutuhan manajemen baik dari sisi operasional (Sistem Informasi) maupun dari sisi pelaporan summary (Data Warehouse)
- Dalam melakukan data mining untuk mendukung kecerdasan bisnis, kegiatan yang perlu dilaksanakan:
 - Mengeksplorasi data yang tersedia di dalam Data Warehouse atau di dalam Sistem Informasi
 - Mempersiapkan (memilah dan memproses) data yang akan digunakan untuk pemodelan
 - Membangun dan mengevaluasi model Data Mining
 - Mengimplementasikan model
 - Mengevaluasi hasil implementasi model



Data Mining Dalam Kecerdasan Bisnis

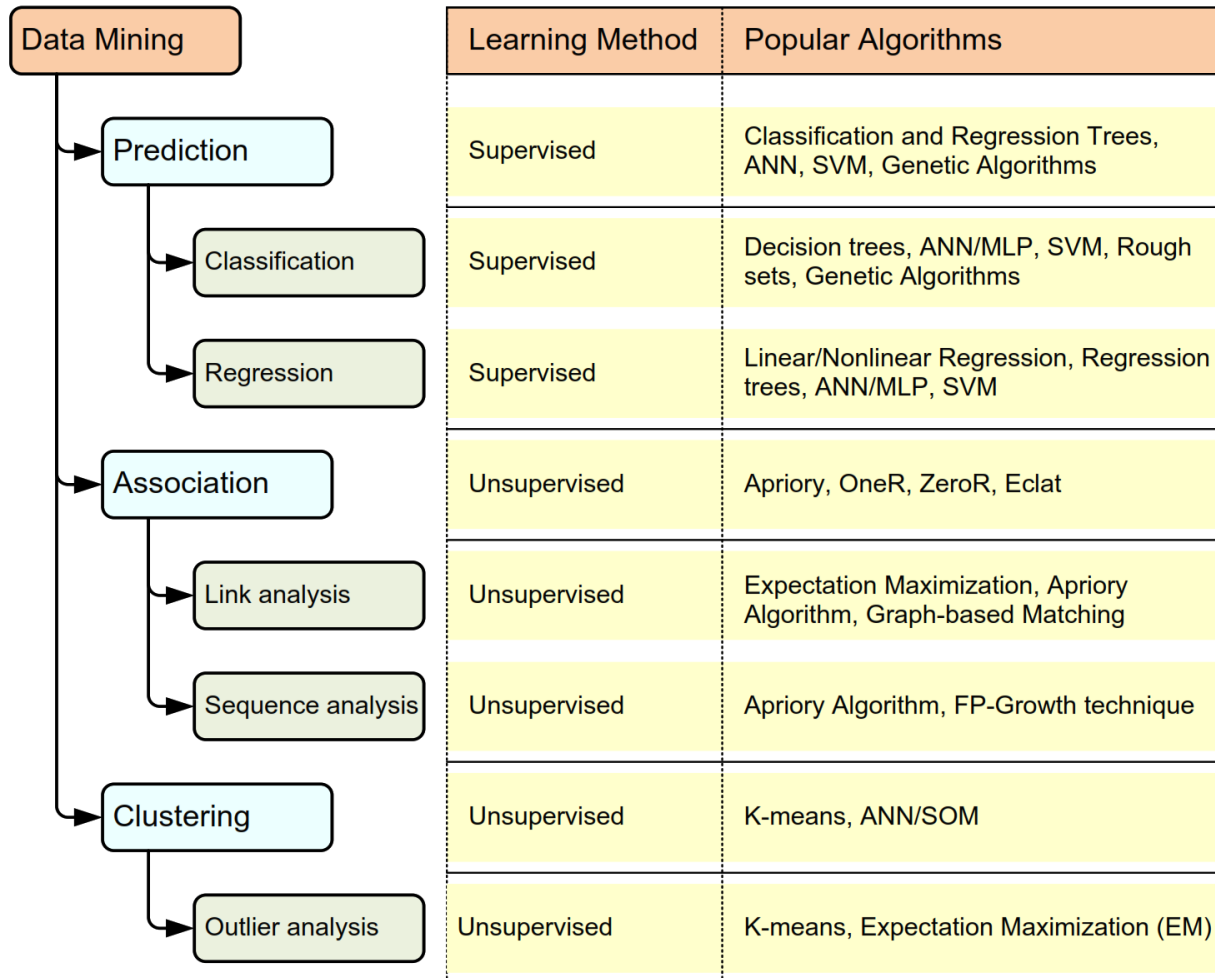
- Data di dalam suatu perusahaan sudah disediakan sesuai dengan kebutuhan manajemen baik dari sisi operasional (Sistem Informasi) maupun dari sisi pelaporan summary (Data Warehouse)
- Dalam melakukan data mining untuk mendukung kecerdasan bisnis, kegiatan yang perlu dilaksanakan:
 - Mengeksplorasi data yang tersedia di dalam Data Warehouse atau di dalam Sistem Informasi
 - Mempersiapkan (memilah dan memproses) data yang akan digunakan untuk pemodelan
 - Membangun dan mengevaluasi model Data Mining
 - Mengimplementasikan model
 - Mengevaluasi hasil implementasi model

Data Mining Process

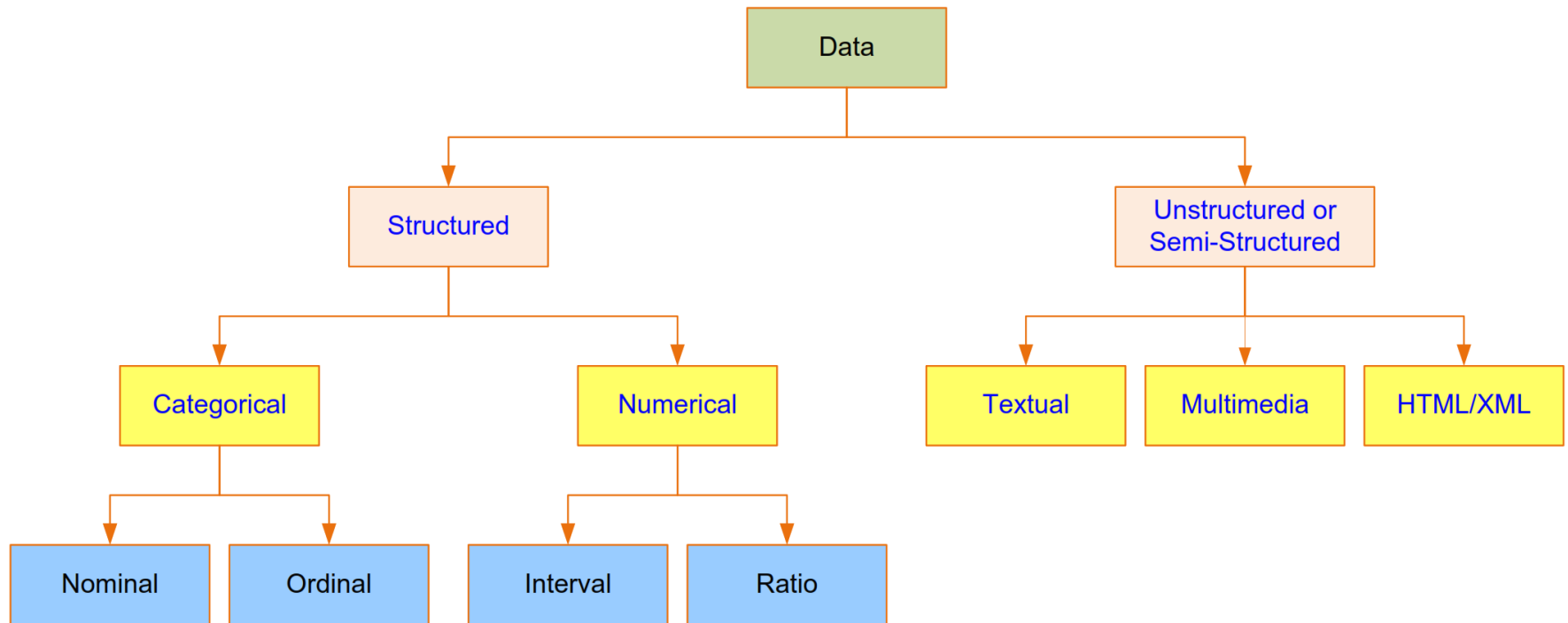




Data Mining



Tipe Data





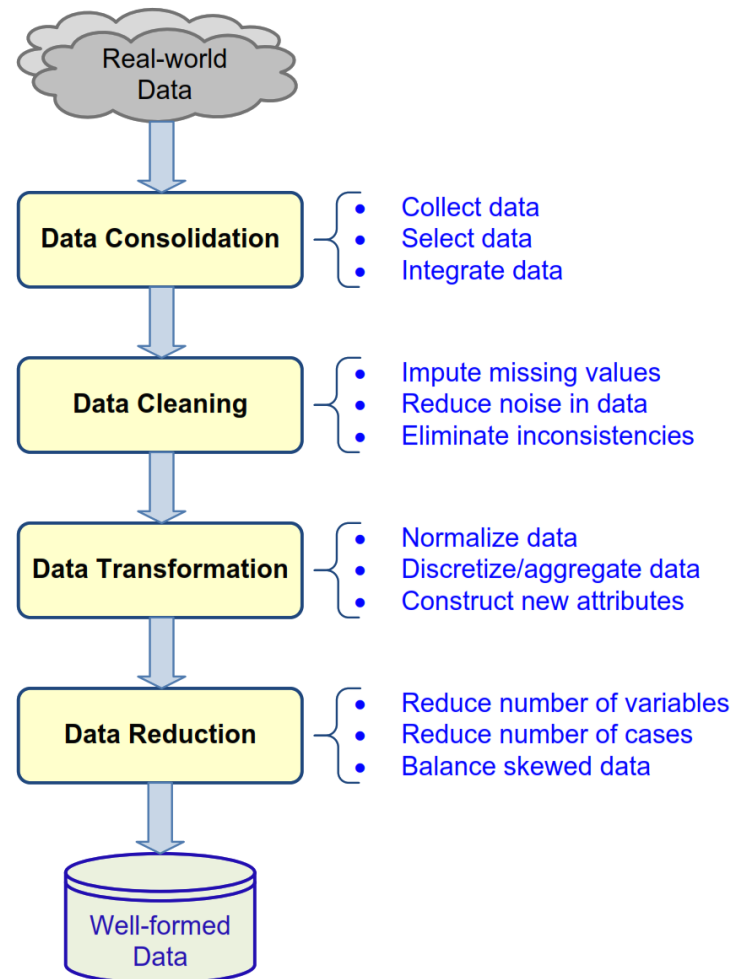
Data Preprocessing dan SAS EM Instalasi

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024

Data Preprocessing





Data Preprocessing

- Data yang tersedia kemungkinan mencakup beberapa komponen yang tidak sesuai seperti adanya outliers, mengandung missing values, kondisinya tidak normal (untuk parametric data mining), mempunyai variable yang tidak penting, mempunyai variable yang bukan angka, dll
- Data preprocessing merupakan proses data mining yang harus dilakukan sebelum pemodelan data mining bisa dilakukan
- Preprocessing dilakukan dengan berbagai cara termasuk penghapusan data, imputasi, transformasi, pemilihan variable, dll



Variable Selection

- Variable selection merupakan suatu preprocessing yang dilakukan untuk memilih variable-variable yang bisa digunakan dan variable-variable yang tidak bisa digunakan dalam pemodelan
- Dataset kemungkinan mencakup variable bukan angka (text) yang mempunyai nilai yang tidak bisa diubah menjadi nominal variable. Untuk jenis data seperti ini diperlukan pemilahan, dan dipastikan data didrop pada saat melakukan pemodelan
- Variable selection dilakukan dengan proses dropping dan keeping variable dari dataset



Variable Transformation

- Transformasi Variabel diperlukan untuk data-data yang mempunyai kondisi yang tidak sesuai dengan yang diharapkan untuk pemodelan data mining
- Kondisi yang tercakup seperti data yang belum dalam bentuk angka, data yang tidak dalam keadaan normal (untuk parametric data mining method), atau adanya kebutuhan data yang bersumber dari beberapa variable yang tersedia
- Transformasi bisa dilakukan dengan berbagai method seperti log transformation, binning, dll



Variable Reduction

- Variable reduction merupakan suatu preprocessing yang dilakukan untuk memilih variable-variable yang mempunyai berpengaruh terhadap pemodelan
- Variable reduction didahului dengan proses analisa korelasi atau analisa variable importance untuk mengetahui variable yang berpengaruh dalam pemodelan
- Variable reduction dilakukan dengan melakukan dropping variable yang tidak digunakan, dan keeping variable yang akan digunakan dalam pemodelan
- Variable reduction juga dilakukan dengan membentuk variable baru yang bersumber dari beberapa variable dengan jumlah yang lebih besar. Salah satu metode yang sering digunakan ada Principal Component Analysis



Penghapusan Outliers

- Outliers merupakan data yang mempunyai pola yang berbeda dari data-data yang lain yang ada di dalam dataset
- Keberadaan outliers sering menjadi permasalahan karena bisa mempengaruhi hasil pemodelan dengan cukup signifikan
- Dua cara yang sering dilakukan untuk menangani outliers yaitu imputasi atau penghapusan data yang merupakan outliers
- Imputasi bisa dilakukan dengan mencari nilai rata-rata atau median dari variable. Imputasi sering dianggap kurang bagus, karena data aslinya diubah menjadi data yang dibentuk secara buatan
- Penghapusan data yang mengandung outliers dianggap proses yang alami dan tidak menciderai konsep-konsep pemodelan. Akan tetapi, hal ini tidak begitu bagus, apabila jumlah data yang tersedia memang sedikit



Normalization

- Normalisasi merupakan suatu tahapan preprocessing untuk handle variable yang mempunyai bentuk yang tidak normal seperti skewed ke kiri, skewed ke kanan, atau mempunyai kurtosis yang terlalu tinggi atau rendah.
- Normalisasi khususnya penting untuk melakukan pemodelan data mining menggunakan metode parametric seperti Naïve Bayes, Neural Networks, Bayesian Network, Regression atau metode-metode lainnya
- Untuk handle kondisi ini beberapa metode bisa digunakan untuk mengubah variable menjadi bentuk yang lebih normal seperti log transformation, binning dan lain-lain



Data Partition Untuk Persiapan Pemodelan

- Data proses pemodelan data mining, umumnya pemodelan dilakukan dalam dua tahap, yaitu tahap pencarian model dan tahap testing model
- Pencarian model merupakan tahapan untuk mendapatkan model dari satu set data training, sedang tahapan testing merupakan suatu tahapan untuk melakukan pengujian seberapa akurat model yang didapatkan dalam proses pencarian model mengklasifikasi sekumpulan data yang ada dalam testing dataset
- Kedua tahap memerlukan sekumpulan data yang berbeda yaitu training dataset dan testing dataset. Untuk mendapatkan kedua jenis dataset dari dataset awal, dilakukan dengan proses data partition



Data Sampling Untuk Persiapan Pemodelan

- Dalam beberapa kondisi data, data sampling perlu untuk dilakukan seperti jumlah data untuk suatu kelas tertentu dalam kondisi yang jauh lebih besar dari kelas yang lainnya
- Kondisi yang lain juga memerlukan data sampling misalnya dalam melakukan proses pemodelan dengan membagi data secara random menjadi beberapa kelompok data yang berbeda-beda. Untuk pemodelan yang terakhir ini, umumnya diperlukan untuk memastikan bahwa model yang didapat cukup representative untuk semua kondisi data yang ditemukan
- Data sampling biasanya dilakukan dengan proses transformation yang memilah data menjadi beberapa sampel dataset yang berbeda-beda



SAS EM Installation

- Akses
<https://welcome.oda.sas.com/login>
- Create Profile
- Verify untuk mengaktifkan SAS Profile
- Login dengan SAS Profile yang sudah dibuat



SAS EM Installation

- Membuat Course untuk memunculkan SAS Enterprise Miner link
 - Membuat course dengan memilih paket SAS yang akan digunakan. Pastikan untuk memilih SAS Enterprise Miner agar muncul dalam list aplikasi SAS yang bisa digunakan
- Mendownload file instalasi SAS Enterprise Miner
- Menginstall SAS Enterprise Miner
- Membuka aplikasi SAS Enterprise Miner



Similarity Measure

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



Distance Base Similarity Measure

- Mengukur jarak berbasis jarak koordinat antar objek
- Beberapa metode pengukuran:
 - Manhattan Distance (L1 Norm)
 - Euclidian Distance (L2 Norm)
 - Minkowski Distance (Lp Norm)
 - Chebyshev Distance (L_infinity Norm)
 - Mahalanobis Distance
 - Hamming Distance
 - Levenshtein Distance



Distance Base Similarity Measure

- Manhattan Distance ($L1$ Norm)

$$d_{L1}(x_1 - x_2) = \sum_{i=1}^n |x_{1i} - x_{2i}|$$

- Euclidean Distance ($L2$ Norm)

$$d_{L2}(x_1 - x_2) = \sqrt{\sum_{i=1}^n (x_{1i} - x_{2i})^2}$$



Distance Base Similarity Measure

- Minkowski Distance (L_p Norm)

$$d_{Lp}(x_1 - x_2) = \sqrt[p]{\sum_{i=1}^n (x_{1i} - x_{2i})^p}$$

- Chebyshev Distance ($L_Infinity$ Norm)

$$d_{L\infty}(x_1 - x_2) = \max_i \{x_{1i} - x_{2i}\}$$



Exercise: Distance Base Similarity Measure

- $X1 = \{8, 6, 3\}$
- $X2 = \{2, 5, 6\}$
- Coba Hitung:
 - Manhattan Distance
 - Euclidean Distance
 - Minkowski Distance ($p=3$)
 - Chebyshev Distance
- Bandingkan Perbedaan Hasil Hitungnya



Distance Base Similarity Measure

- Mahalanobis Distance

$$d_{Mahalanobis}(x_1 - x_2) \\ = \sqrt{[x_{1i} - x_{2i}]^T \Sigma [x_{1i} - x_{2i}]}$$

- Σ =Covariance Matrix

- Kalau Σ adalah Matrix Identity, maka Mahalanobis Distance adalah Eucledian Distance



Distance Base Similarity Measure

- Kalau Σ adalah Matrix Diagonal, maka Mahalanobis Distance adalah Normalized Euclidean Distance yang dihitung dengan:

$$d_{Normalized}(x_1 - x_2) = \sqrt{\sum_{i=1}^n \frac{(x_{1i} - x_{2i})^2}{\sigma^2}}$$

- Matrix Diagonal $\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$



Exercise: Distance Base Similarity Measure

- $X1=\{8,6,3\}$
- $X2=\{2,5,6\}$
- Coba Hitung Jarak Mahalanobis Dengan Matrix Covariance

$$\begin{bmatrix} 5 & 3 & 2 \\ 1 & 4 & 0 \\ 1 & 2 & 3 \end{bmatrix}$$

- Hitung juga Jarak Mahalanobis Dengan Matrix Identity, dan Matrix Diagonal Dengan Angka di Diagonal Adalah Angka 2



Distance Base Similarity Measure

- Hamming Distance merupakan alat ukur kemiripan dengan antara dua string yang ukurannya sama dengan membandingkan simbol-simbol yang terdapat pada kedua string pada posisi yang sama.
- Hamming distance dari dua string adalah jumlah simbol dari kedua string yang berbeda.
 - Sebagai contoh Hamming distance antara string 'toned' dan 'roses' adalah 3.
- Hamming Distance juga digunakan untuk mengukur jarak antar dua string binary misalnya jarak antara 10011101 dengan 10001001 adalah 2.



Distance Base Similarity Measure

- Levenshtein Distance adalah alat ukur kemiripan antara dua string yang ukurannya tidak sama dengan menghitung jumlah pengoperasian yang perlu dilakukan untuk mengubah string yang satu menjadi string yang kedua yang diperbandingkan.
- Pengoperasian yang dilakukan termasuk operasi insert, delete dan substitusi.
- Sebagai contoh Levenshtein distance antara string 'kitten' dan 'sitting' adalah 3 dengan pengoperasian substitusi k dengan s, substitusi e dengan i, dan insert g.



Probability Base Similarity Measure

- Kullback Leibler Distance mengukur tingkat kemiripan variabel objek yang direpresentasikan dalam bentuk probabilitas dua distribusi statistik.
- Sering disebut juga **information distance**, **information gain**, atau **relative entropy**.
- Jarak antara dua objek yang bernilai diskrit dalam Kullback Leibler distance dihitung dengan rumus sebagai berikut:

$$d_{KL}(P, Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

- Sedang untuk objek yang bernilai continuous dihitung dengan rumus sebagai berikut:

$$d_{KL}(P, Q) = \int_i P(i) \log \frac{P(i)}{Q(i)}$$



Probability Base Similarity Measure

- Dua buah populasi yang terdiri dari 3 kategori: A, B, dan C, mempunyai kemungkinan untuk masing-masing kategori sebagai berikut:
 - Populasi I = $\{1/4, 1/2, 1/4\}$
 - Populasi II = $\{1/2, 1/8, 3/8\}$
- Hitung Jarak Kullback Leibler di antara kedua populasi tersebut



Set Base Similarity Measure

- Jaccard Index adalah indeks yang menunjukkan tingkat kesamaan antara suatu himpunan (set) data dengan himpunan (set) data yang lain.
- Jaccard Index dihitung menggunakan rumus sebagai berikut:

$$J(A, B) = \frac{A \cap B}{A \cup B}$$

- Sebagai kebalikannya, tingkat ketidak samaan antara dua himpunan dihitung dengan:

$$J_{\delta}(A, B) = \frac{((A \cup B) - (A \cap B))}{(A \cup B)}$$



Set Base Similarity Measure

- Dua himpunan A dan B, masing-masing beranggotakan:
 - Himpunan $A = \{a, b, c, d, e, f, g\}$
 - Himpunan $B = \{d, e, f, g, h, i\}$
- Hitung Jaccard Index dari kedua himpunan tersebut
- Hitung juga tingkat ketidaksetaraan di antara kedua himpunan tersebut



Feature Base Similarity Measure

- Feature-based similarity measure melakukan penghitungan tingkat kemiripan dengan merepresentasikan objek ke dalam bentuk feature-feature yang ingin diperbandingkan.
- Feature-based similarity measure banyak digunakan dalam melakukan pengklasifikasian atau pattern matching untuk gambar dan text.



Context Base Similarity Measure

- Context-based similarity measure melakukan penghitungan tingkat kemiripan objek-objek yang mempunyai struktur yang tidak biasa
- Objek yang diukur direpresentasikan struktur yang tidak biasa seperti dengan tree structure, diagram, atau struktur kompleks yang lainnya.



Studi Case Data Warehouse Usaha Retail

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



Studi Kasus Usaha Retail

- Latar Belakang
 - Perusahaan grocery besar dengan perkiraan 500 outlet
 - Setiap outlet mempunyai sekitar 60000 produk dalam tampilannya
 - Teknologi yang digunakan
 - SKU – Stock Keeping Unit
 - UPC – Universal Product Code



Studi Kasus Usaha Retail

- Problem Statement:
 - Perlu untuk memaksimalkan keuntungan dan tetap menjaga stok agar tetap ada
 - Perlu informasi untuk mendukung pengambilan keputusan dalam hal penetapan harga dan promosi. Tipe promosi yang dilaksanakan:
 - Discount harga sementara
 - Reklame surat kabar
 - Tampilan lemari dan lorong
 - Kupon

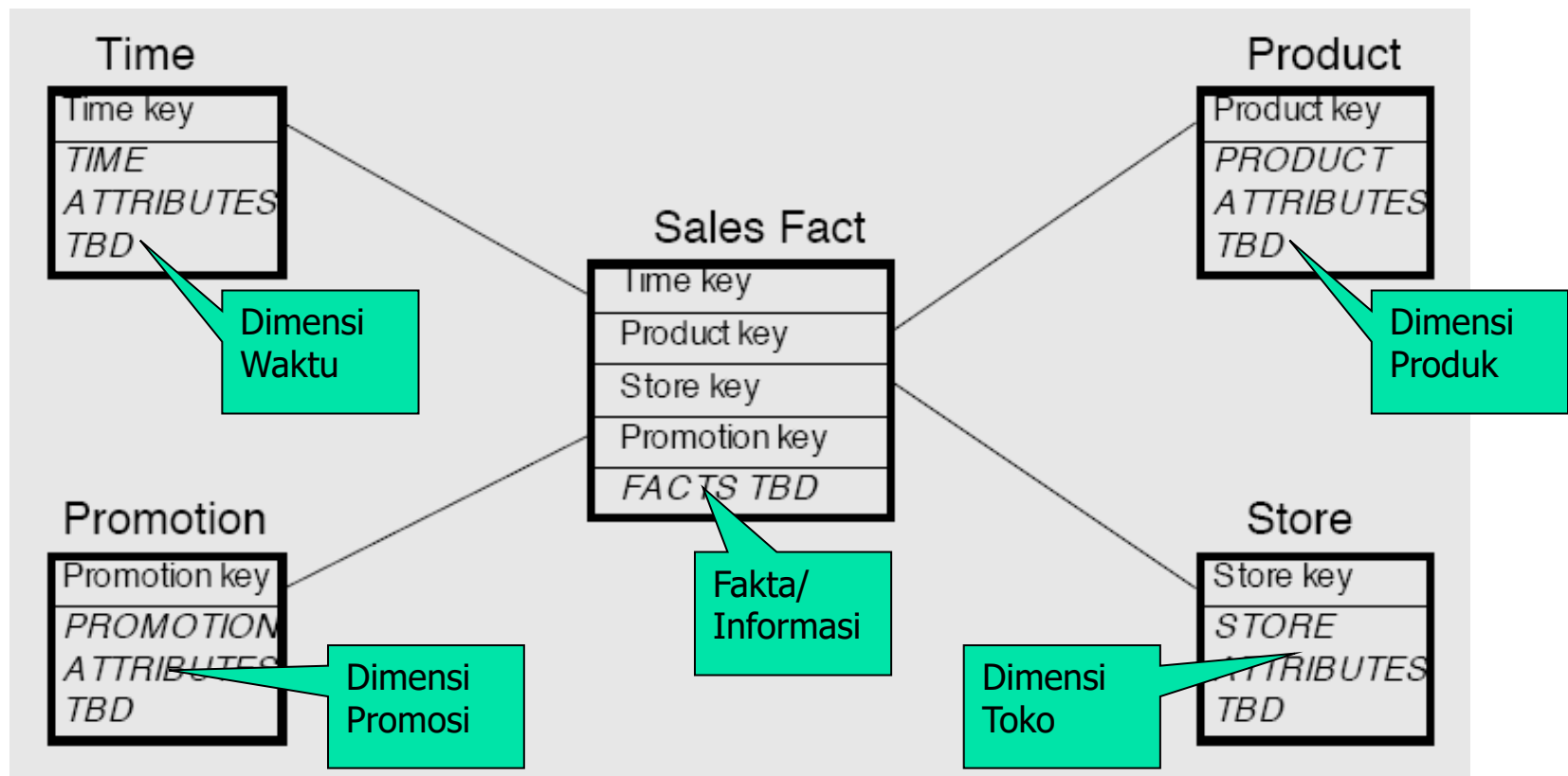


Usaha Retail

- Proses yang perlu dianalisa
 - Memilih Proses Bisnis
 - Pergerakan barang harian yang mencakup transaksi penjualan dan pembelian barang
 - Memilih fakta/informasi untuk menjawab permasalahan
 - **Unit Terjual** by Toko by Promosi by Hari
 - Memilih variable dimensi
 - Waktu, Produk, Toko dan Promosi

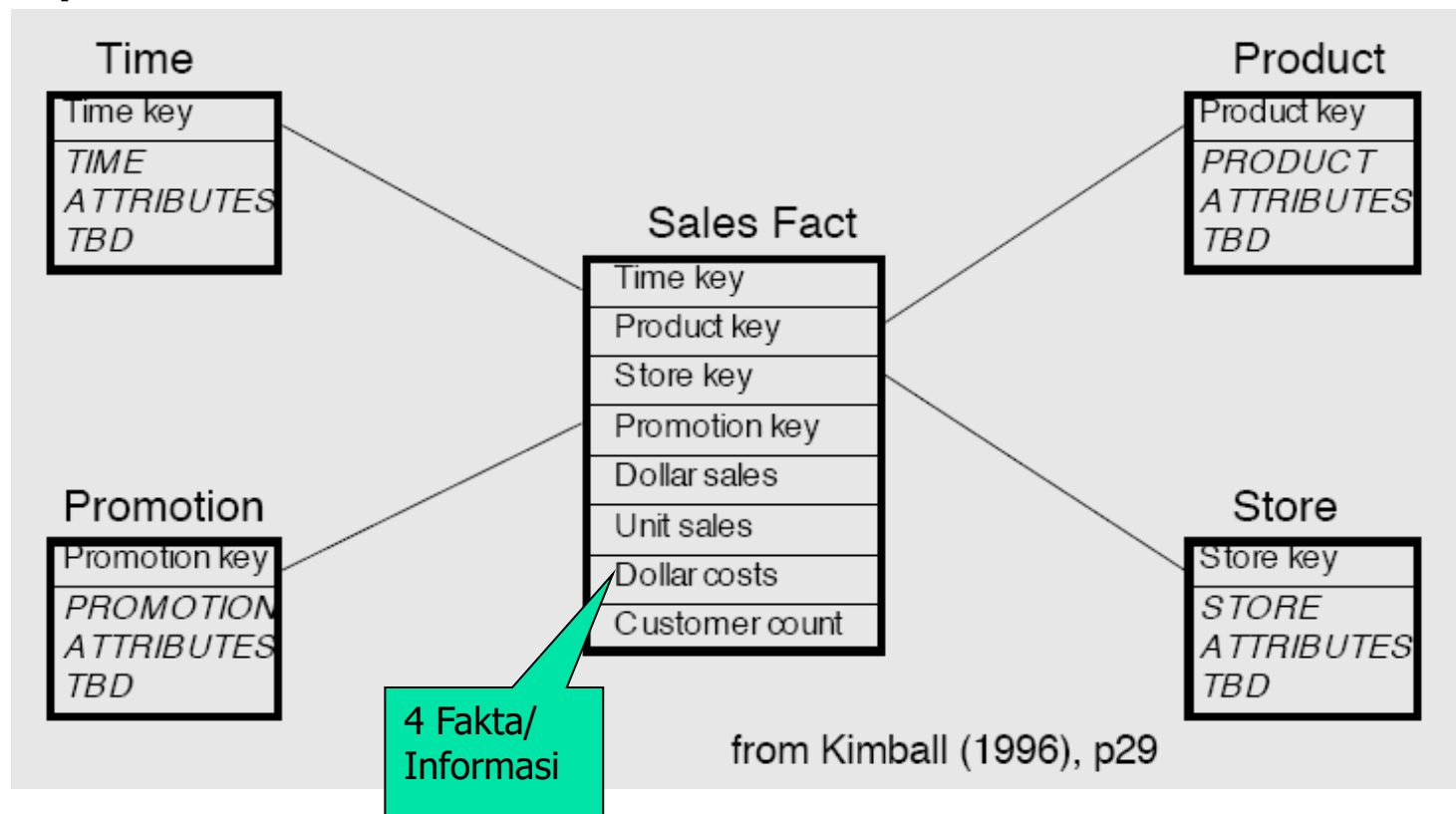
Disain Data Warehouse Usaha Retail

- Fakta/Informasi Untuk Menjawab Permasalahan dan Varibel Dimensi



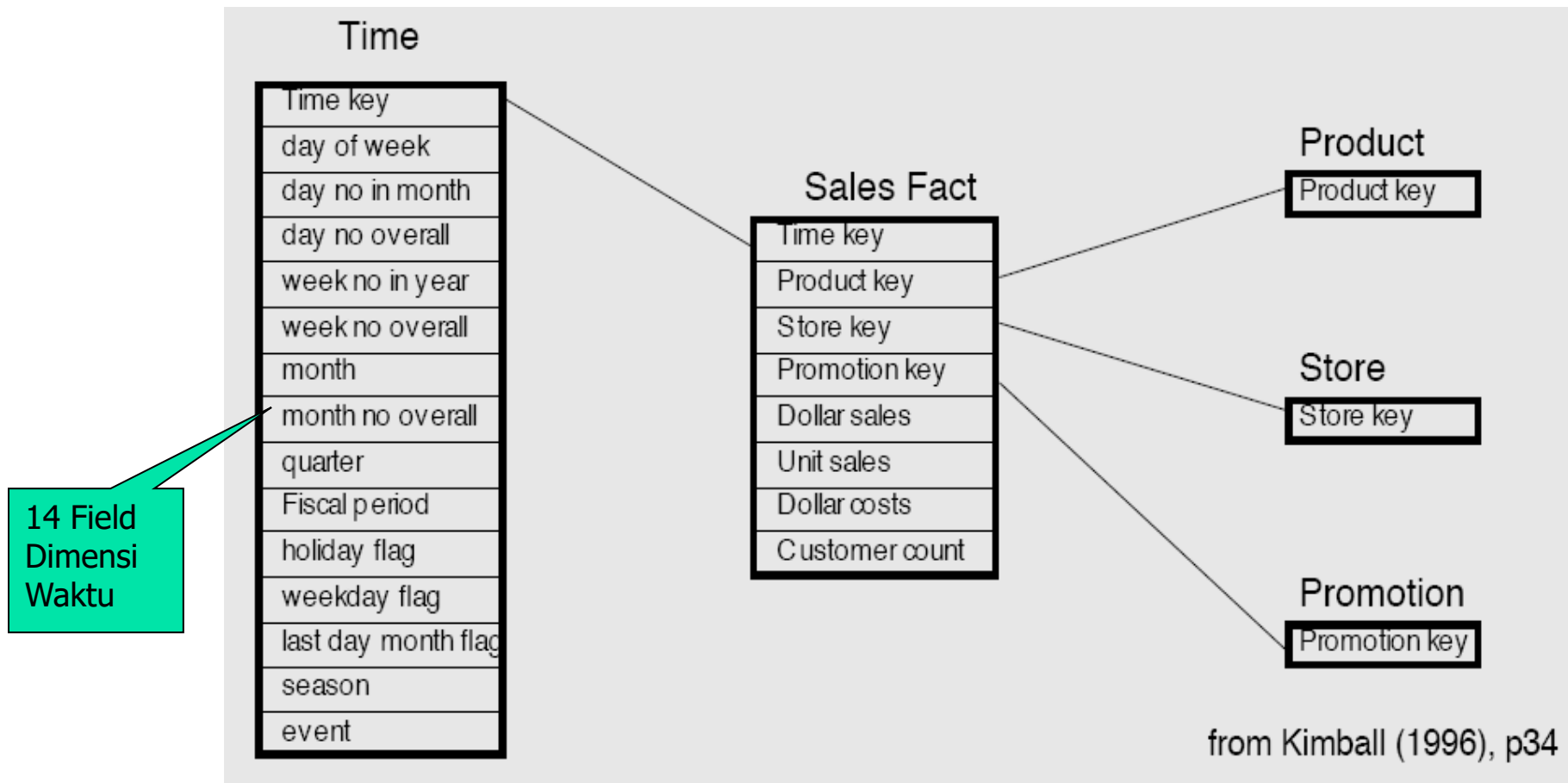
Desain Data Warehouse Usaha Retail

- Fakta/informasi untuk menjawab permasalahan



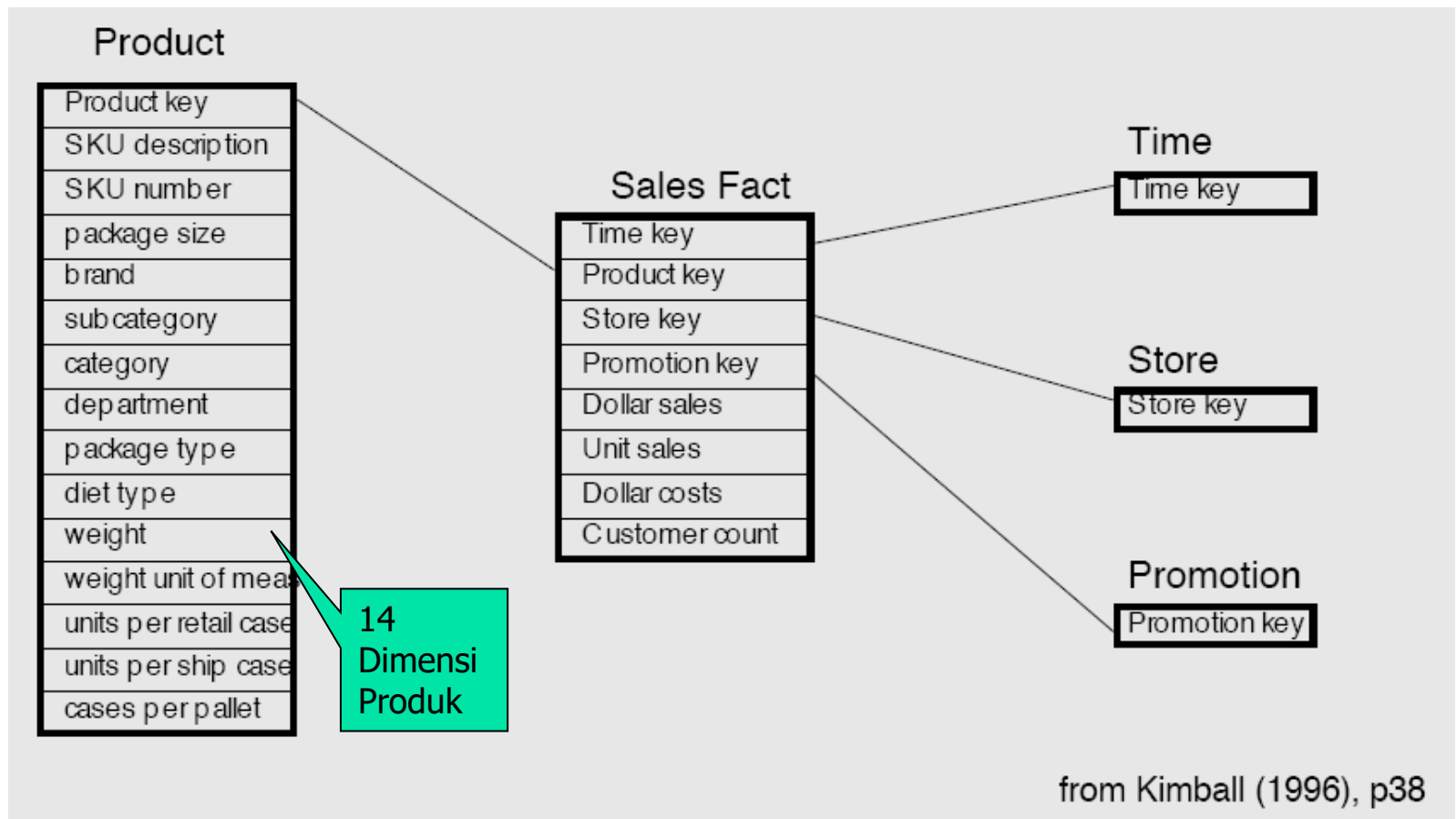
Design Data Warehouse Usaha Retail

- Dimensi Waktu



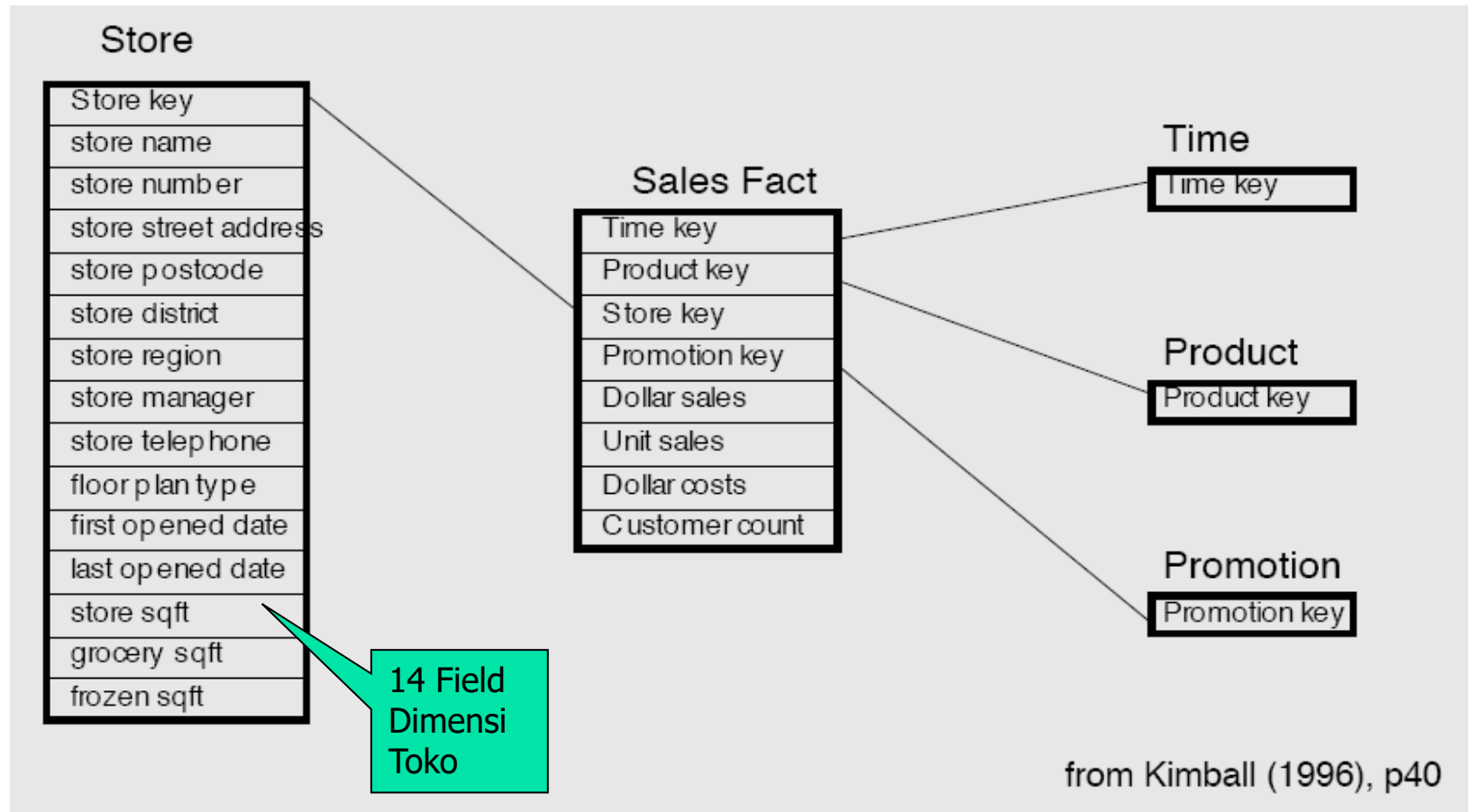
Design Data Warehouse Usaha Retail

■ Dimensi Produk



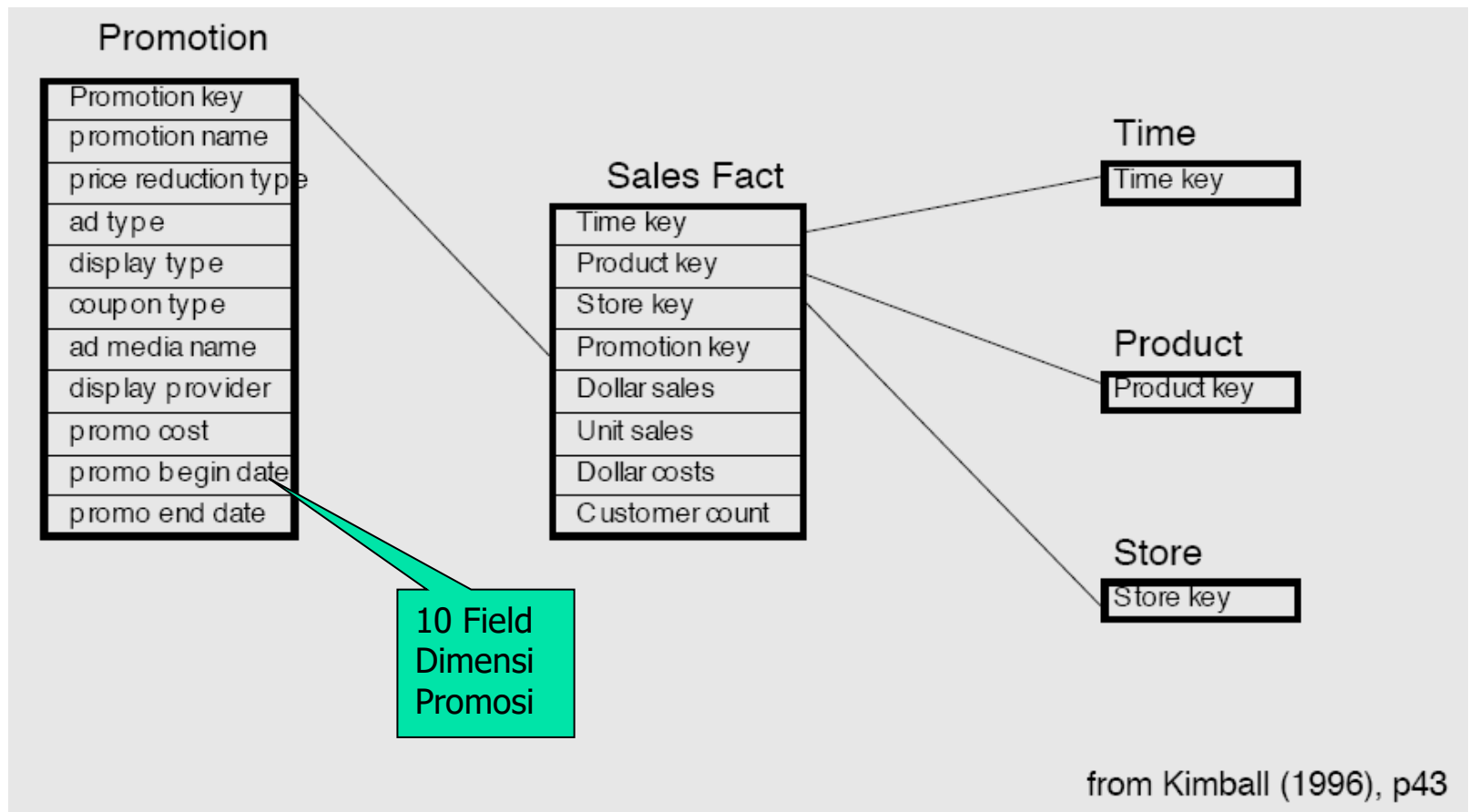
Design Data Warehouse Usaha Retail

■ Dimensi Toko



Usaha Retail: Dimensi Promosi

■ Dimensi Promosi



Disain Database Sistem Infomasi Usaha Retail

ID Hari Libur
Tanggal
Nama Hari Libur

ID Event
Tanggal
Nama Event

ID Toko
Store Name
Store Number
Store Street Address
Store Post Code
Store District
Store Region
Store Manager
Store Telephone
Floor Plan Type
First Opened Date
Last Opened Date
Store sqft
Grocery sqft
Frozen sqft

ID Customer
Name
Address
Phone Number
Email

ID Transaksi
Tanggal Transaksi
ID Toko
ID Customer
ID Produk
ID Promotion
Volume
Harga Satuan
Total Penjualan
Biaya Transaksi
Biaya Pengiriman

ID Promotion
Promotion Name
Price Reduction Type
Ad Type
Display Type
Coupon Type
Ad Media Type
Display Provider
Promo Cost
Promo Begin Date
Promo End Date

ID Produk
Produk Name
SKU Description
SKU Number
Size
Brand
Sub Category
Category
Department
Package Type
Diet Type
Weight
Weight Unit of Measurement
Units per Retail Case
Units per Ship Case
Cases per Pallet



Potensi Implementasi Data Mining

- Problem Statement:

- A: Perlu untuk memaksimalkan keuntungan dan tetap menjaga stok agar tetap ada
- B: Perlu informasi untuk mendukung pengambilan keputusan dalam hal penetapan harga dan promosi

- Solusi Data Mining

- Melihat trend keuntungan (A)
- Melihat trend penjualan per barang dan melakukan prediksi penjualan per barang (dan membandingkan dengan prediksi kondisi stok per barang) (A)
- Melihat keterkaitan satu barang dengan barang yang lain dalam satu transaksi (B)
- Melihat trend penjualan per jenis promosi (B)



Potensi Implementasi Data Mining

- Melihat trend keuntungan (A)
 - Metode: Regression
- Melihat trend penjualan per barang dan melakukan prediksi penjualan per barang (dan membandingkan dengan prediksi kondisi stok per barang) (A)
 - Metode: Regression, Neural Networks
- Melihat keterkaitan satu barang dengan barang yang lain dalam satu transaksi (B)
 - Metode: Association Rules
- Melihat trend penjualan per jenis promosi (B)
 - Metode: Regression



Potensi Implementasi Data Mining

- Market segmentation dengan mengelompokkan customer
 - Metode: Clustering
- Memprediksi penjualan berdasarkan beberapa informasi seperti hari libur/tidak, ada event/tidak, ada promosi/tidak, kategori barang, dll
 - Metode: Decision Trees, Neural Networks, Naïve Bayes



Decision Trees

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



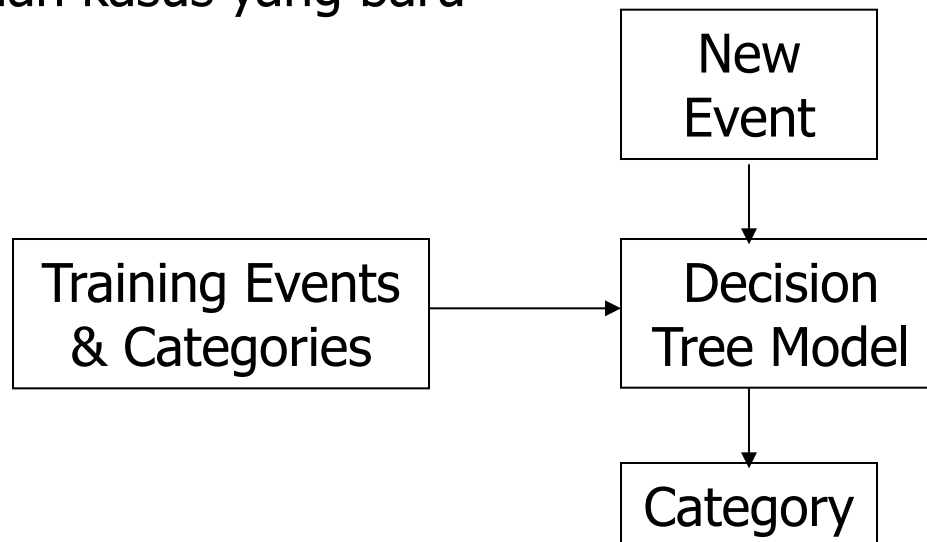
Decision Trees

- Diberikan suatu case, diminta untuk memprediksi kategori dari case tersebut. Contoh:
 - Siapa yang akan memenangkan pertandingan sepak bola
 - Bagaimana kita harus menyimpan email yang masuk
- Case = sekumpulan variable, contoh:
 - Pertandingan sepak bola: siapa penjaga gawangnya?
 - Email: siapa yang mengirimkan email



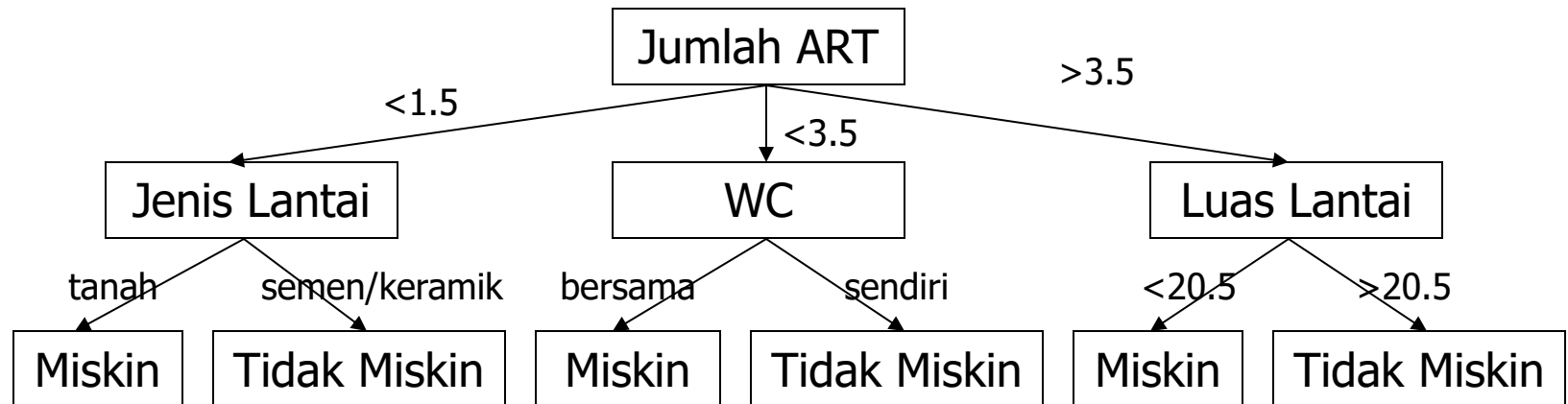
Decision Trees

- Semua data di dalam training data mempunyai kategori masing-masing
- Menggunakan data training untuk membangun pohon Keputusan
- Menggunakan model decision tree model untuk memprediksi kategori dan kasus yang baru



Decision Trees

- Sebuah decision tree ada sebuah pohon Dimana:
 - Setiap node di pertengahan dilabeli dengan nama variable
 - Setiap tanda panah yang keluar dari node pertengahan dilabeli dengan nilai dari variable tsb
 - Setiap daun dilabeli dengan sebuah kategori





Contoh Training Data

Jumlah ART	Jenis Lantai	WC	Luas Lantai	Category
5	Tanah	Sendiri	15	Miskin
3	Tanah	Sendiri	20	Tidak Miskin
4	Semen	Sendiri	50	Tidak Miskin
1	Semen	Bersama	40	Tidak Miskin
1	Tanah	Sendiri	30	Miskin
2	Semen	Bersama	40	Miskin
8	Semen	Bersama	18	Miskin



Menemukan Decision Tree yang Konsisten

- Terlalu banyak solusi decision tree untuk dicoba
- Membangun sebuah decision tree secara top-down, menggunakan recursive partitioning, sebagai berikut:
 - Kalau semua kasus yang masuk memiliki kategori yang sama:
 - Buat sebuah daun dengan kategori tersebut
 - Kalau tidak
 - Ambil suatu variabel, dan buat node lagi untuk feature yang baru diambil
 - Untuk semua nilai feature:
 - Kembali ke step pertama untuk membuat sub-tree untuk semua event dengan nilai feature tersebut



Membuat Decision Tree

- Menggunakan training data yang terdiri dari beberapa case dan kategori untuk membuat decision tree
- Karakteristik seperti apa yang sebuah decision tree harus punya?
 - Harus konsisten dengan training data
 - Banyak decision tree yang konsisten dengan training data
 - Harus tetap sederhana
 - Occam's Razor
 - Model yang sederhana menggeneralisasikan keadaan lebih baik
 - Dalam sebuah decision tree yang lebih sederhana, setiap node berbasis pada jumlah data yang lebih banyak (secara rata-rata)



Menemukan Decision Tree Yang Bagus

- Feature mana yang akan dipakai sebagai node pembagi
 - Kita ingin decision tree yang kecil
 - Ambil sebuah feature yang memberikan informasi paling banyak tentang category kelas pilihan
- Ilustrasi: Menanyakan 20 Pertanyaan
 - Saya memikirkan angka dari 1 sampai 1000
 - Anda dapat menanyakan pertanyaan dengan jawaban ya/tidak (binary)
 - Pertanyaan apa yang akan anda tanyakan?
 - Secara rata-rata, beberapa pertanyaan memberikan informasi lebih banyak daripada yang lainnya
 - Apakah 752? Apakah bilangan prima? Apakah bilangan di antara 1 sampai 500?
 - Pilih pertanyaan yang memberikan informasi paling banyak



Entropy

- Entropy memberikan suatu ukuran seberapa banyak kita tahu tentang class kategori pilihan atau seberapa banyak informasi yang kita dapatkan
 - Secara rata-rata, berapa banyak pertanyaan ya/tidak yang perlu ditanyakan untuk menentukan kategori kelasnya
 - Semakin banyak kita tahu, semakin kecil entropy nya
- Entropy dari sekumpulan case/data E adalah $H(E)$
$$H(E) = \sum_{c \in C} P(c) \log_2 P(c)$$
 - Dimana $P(c)$ adalah probabilitas dari suatu case di dalam E yang memiliki kategori kelas c
- Entropy berarti rata-rata yang bisa kita harapkan agar suatu case itu terjadi



Entropy

- Contoh dari perhitungan Entropy

$$H(E) = \sum_{c \in C} P(c) \log_2 P(c)$$

- Untuk dataset di atas, $H(E)$ dihitung sebagai berikut:
 - $H(E) = P(\text{tidak miskin}) \log P(\text{tidak miskin}) + P(\text{miskin}) \log P(\text{miskin})$
 - $H(E) = 3/7 \log 3/7 + 4/7 \log 4/7$
 - $H(E) = 0.43 \times (-0.37) + 0.57 \times (-0.24)$
 - $H(E) = -0.16 - 0.17 = -0.33$



Information Gain

- Seberapa banyak informasi yang diberikan oleh sebuah feature tentang kategori class pilihan
 - $H(E)$ =entropy dari sekumpulan kasus E
 - $H(E/f)$ =entropy harapan dari sekumpulan kasus E , setelah kita mengetahui nilai dari feature f
 - $G(E,f)=H(E)-H(E/f)$ =jumlah informasi yang diberikan oleh feature f
- Pecah decision tree dengan feature yang memaksimalkan information gain



Information Gain

- Information Gain untuk feature 'Jenis Lantai' dihitung sebagai berikut:
 - $H(E|f) =$
 - $P(\text{tidak miskin}) \times \{P(\text{tanah}|\text{tidak miskin}) \log P(\text{tanah}|\text{tidak miskin}) + P(\text{semen}|\text{tidak miskin}) \log P(\text{semen}|\text{tidak miskin})\} +$
 - $P(\text{miskin}) \times \{P(\text{tanah}|\text{miskin}) \log P(\text{tanah}|\text{miskin}) + P(\text{semen}|\text{miskin}) \log P(\text{semen}|\text{miskin})\}$
 - $H(E|f) =$
 - $3/7 \times \{1/3 \log 1/3 + 2/3 \log 2/3\} + 4/7 \times \{1/2 \log 1/2 + 1/2 \log 1/2\}$
 - $0,43 \times \{0,33 \times (-0,48) + 0,67 \times (-0,17)\} + 0,57 \times \{0,5 \times (-0,30) + 0,5 \times (-0,30)\}$
 - $0,43 \times \{-0,16 - 0,11\} + 0,57 \times \{-0,15 - 0,15\}$
 - $0,43 \times 0,27 + 0,57 \times 0,30 = -0,3861 - 0,171 = -0,5561$
 - $G(E,f) = H(E) - H(E|f) = -0,33 - (-0,5561) = 0,2261$



Information Gain

- Di lain sisi, feature 'Jumlah ART' dan 'Luas Lantai', harus melaksanakan feature value split menjadi bentuk data category, karena feature tersebut adalah feature data continuous
- Menentukan nilai split juga dilaksanakan dengan melihat information gain yang bisa didapatkan dari pengkategorian data category setiap kali melakukan pemecahan feature
- Cara perhitungan information gain sama dengan cara perhitungan feature yang bersifat category



Information Gain Ratio

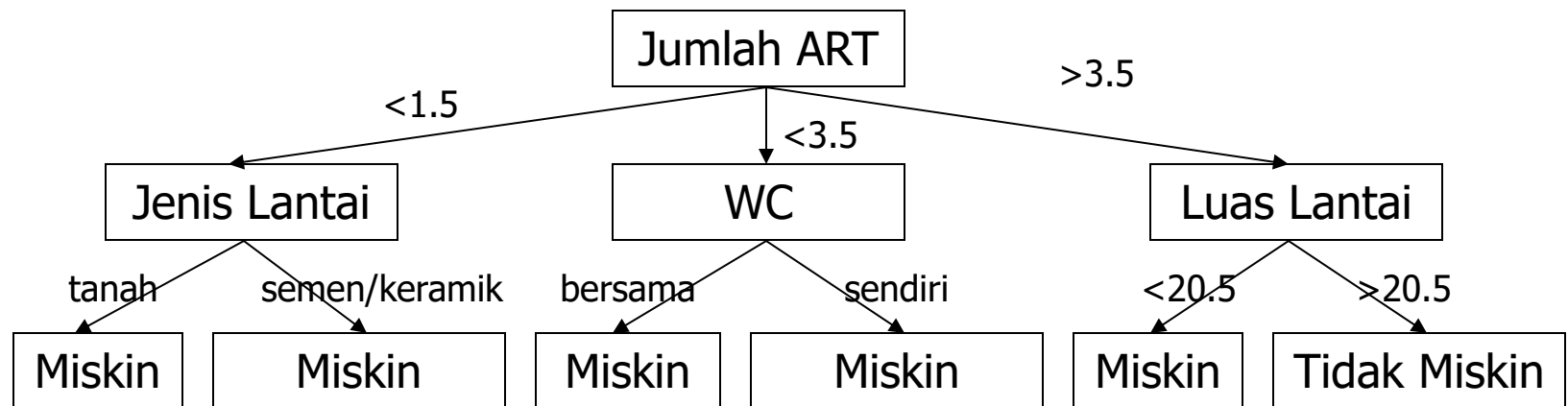
- Akan tetapi, information gain bersifat bias kepada feature yang mempunyai pilihan yang lebih banyak
 - Nilai information gain maksimum untuk suatu feature tergantung pada jumlah nilai dari feature tersebut:
 - Max information gain untuk feature binary adalah 2
 - Max information gain dengan feature value sebanyak 1024 adalah 10
- Gunakan information gain ratio untuk menghilangkan bias:

$$GR(E, f) = \frac{G(E, f)}{-\sum_{v \in V} P(v) \log_2 P(v)}$$

- Dimana v ada di dalam V (semua nilai di dalam feature yang digunakan dalam perhitungan)
- Dalam kasus feature "Jenis Lantai": $P(\text{tanah}) \log P(\text{tanah}) + P(\text{semen}) \log P(\text{semen})$

Permasalahan Overfitting

- Decision tree mungkin terbentuk dari kekhasan dari data training.
 - Contoh, misalnya menggunakan decision tree ini



- Seberapa banyak kita bisa mempercayai daun "Tidak Miskin", kalau itu didasarkan pada hanya satu case data training



Permasalahan Overfitting

- Decision tree yang konsisten secara penuh kemungkinan akan menggeneralisasikan keadaan secara berlebihan
 - Khususnya jika decision tree terlalu besar
 - Atau jika tidak tersedia data training yang cukup
- Keseimbangan antara konsistensi dan kesederhanaan:
 - Decision tree yang lebih besar bisa lebih konsisten
 - Decision trees yang lebih kecil menggeneralisasikan kondisi lebih baik



Solusi dengan Pruning

- Kita data mengurangi overfitting dengan mengurangi ukuran decision tree:
 - Membuat decision tree lengkap
 - Hilangkan daun-daun yang tidak bisa dipercaya
 - Ulangi sampai decision treenya sederhana
- Daun mana yang tidak bisa dipercaya?
 - Daun dengan jumlah data lebih kecil
- Kapan proses pruning harus diberhentikan?
 - Sudah tidak ada lagi daun-daun yang tidak bisa dipercaya



Test Data

- Bagaimana kita bisa mengatakan bahwa suatu decision tree overfitting?
 - Menggunakan data training untuk membuat model ("training set")
 - Menggunakan sisa data training untuk melakukan testing terhadap overfitting ("test set")
 - Untuk setiap daun:
 - Test performance dari decision tree menggunakan decision tree tanpa daun yang diuji
 - Jika performance meningkat, hilangkan daun tersebut
 - Ulangi sampai tidak ada lagi daun yang perlu dihilangkan



Latihan

Jumlah ART	Jenis Lantai	WC	Luas Lantai	Category
5	Tanah	Sendiri	15	Miskin
3	Tanah	Sendiri	20	Tidak Miskin
4	Semen	Sendiri	50	Tidak Miskin
1	Semen	Bersama	40	Tidak Miskin
1	Tanah	Sendiri	30	Miskin
2	Semen	Bersama	40	Miskin
8	Semen	Bersama	18	Miskin



Exercise

- Hitung information gain dari feature “Jenis WC” dan “Luas Lantai” pada saat dilakukan split yang pertama



Decision Trees Dengan SAS EM

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



Pemodelan Decision Tree

- Pilih tombol Decision Tree pada toolbar Model
- Drag pada workspace
- Koneksikan dengan proses data processing terakhir
- Run



Pemodelan Dengan Data Partition

- Pilih tombol Data Partition pada toolbar Sample, drag pada workspace
- Koneksikan dengan proses data processing terakhir
- Set Training dan Validation Data pada window Property dengan nilai 50%:50% (Testing Data 0%)
- Pilih tombol Decision Tree pada toolbar Model, drag pada workspace
- Koneksikan dengan Data Partition
- Run



Pemodelan Dengan Mengubah Parameter Pemodelan

- Pilih tombol Decision Tree pada toolbar Model, drag pada workspace
- Koneksikan dengan Data Partition
- Ubah Nilai Maximum Branch menjadi 5
- Koneksikan model yang baru dibuat ke Model Comparion
- Run, dan lihat hasilnya serta bandingkan



Neural Networks

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

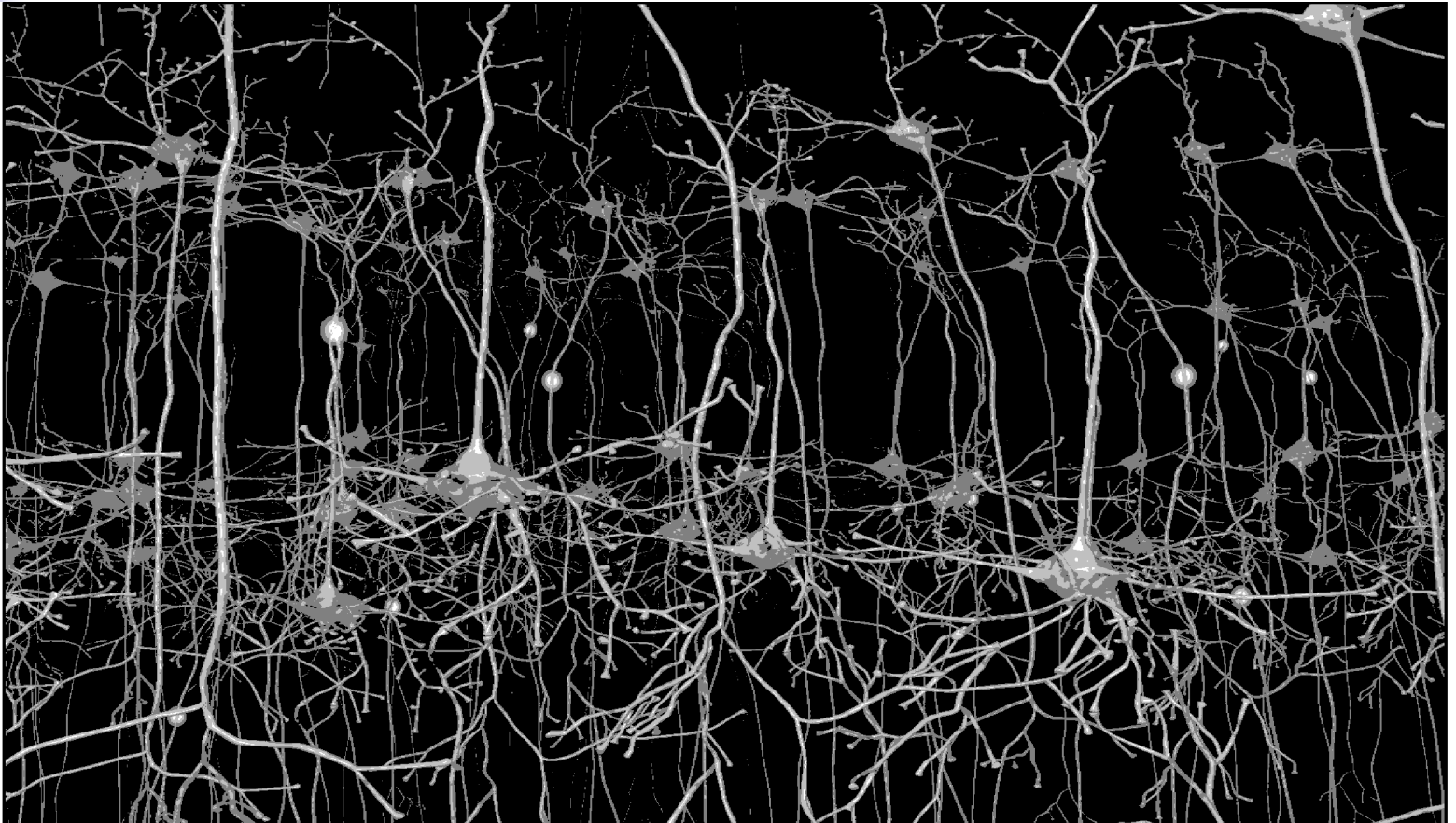
Copyright © Yudi Agusta, PhD 2024



Neural Networks

- Pemodelan artificial neuron dilakukan pertama kali oleh McCulloch dan Pitts (1943)
- Beberapa penelitian kemudian dilakukan oleh peneliti lain termasuk di dalamnya thesis PhD dari Minsky (1954)
- Artificial neural networks adalah suatu usaha untuk memodel kemampuan memproses informasi dari sistem saraf.

Biological Neuron



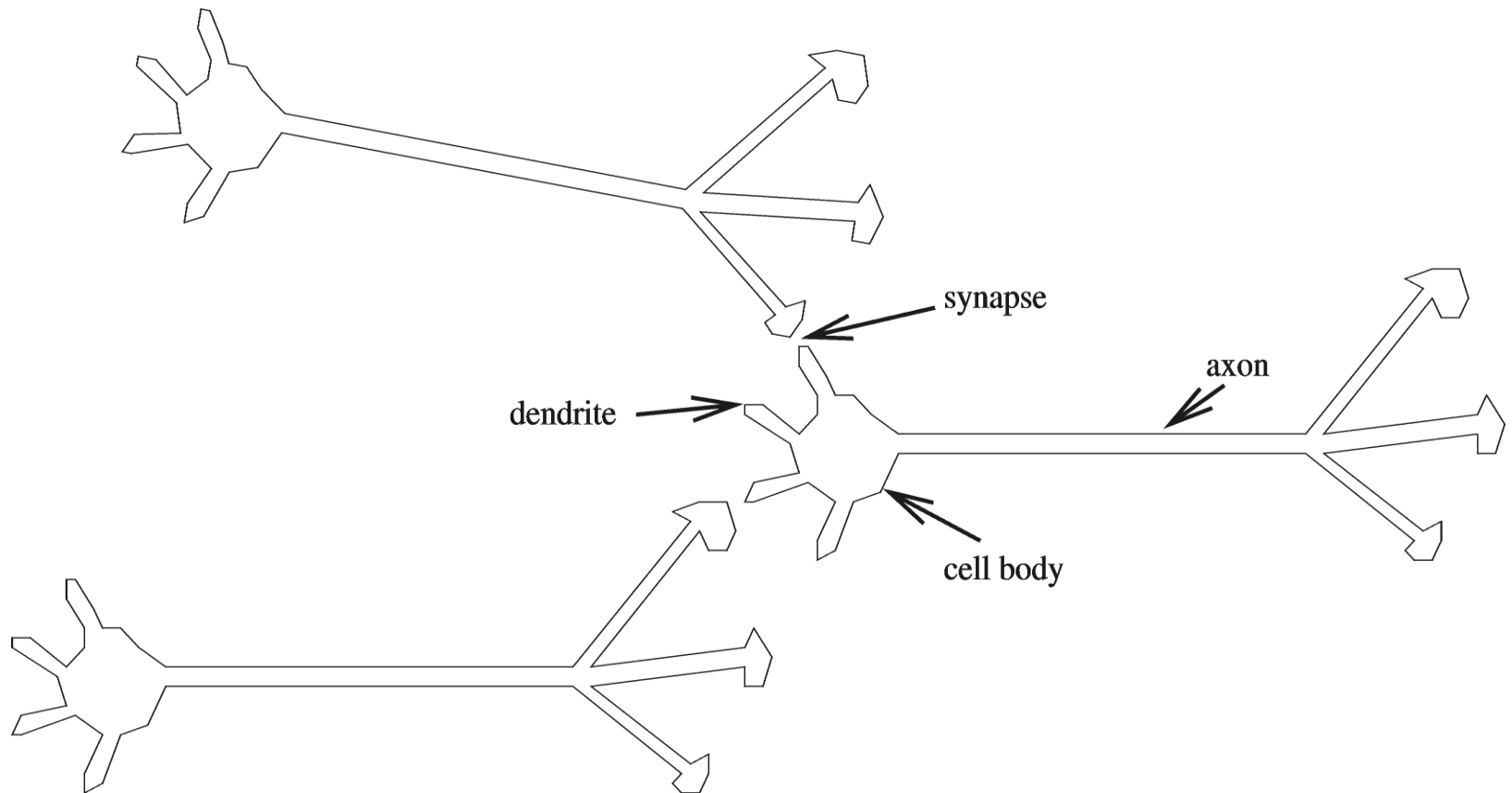
© Digital Studio, Paris - France. All rights reserved.

CG image of the vertical organization of neurons in the primary visual cortex (V1).
Smooth stellate and spiny stellate cells relay visual information coming out from the retina to pyramidal cells, themselves doing a first basic computation of visual motion perception.
version of July 2000





Artificial Neuron





Biological Neuron Networks

- UNIT:

- Nerve cells dinamakan neurons
- Banyak tipe dan sangat kompleks
- Sekitar 10^{11} neurons di dalam otak

- INTERAKSI

- Sinyal diteruskan oleh potensi keputusan akshi
- Interaksi bisa sama secara kimia (melepas atau menerima ion) atau secara elektrik
- Setiap neuron membuat kontak dengan sekitar 10^3 neurons yang lain

- STRUKTUR

- Feedforward, feedback, self-activation recurrent



Artificial Neural Networks (ANN)

- UNIT

- Artificial Neuron (Linear or Non-Linear Input-Output Unit)
- Jumlah Kecil – Beberapa Ratus Unit

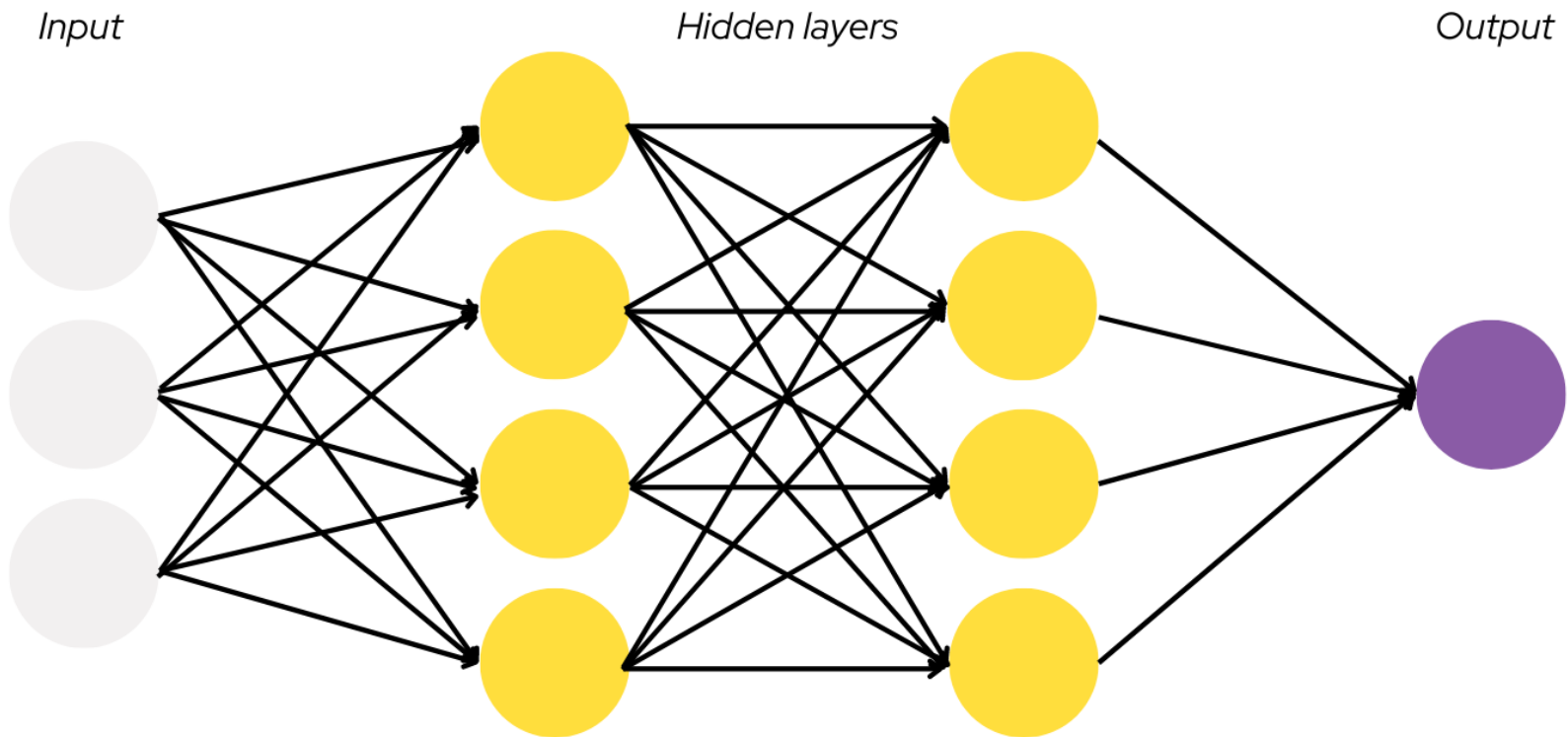
- INTERAKSI

- Dilakukan dengan weights (bobot)
- Seberapa besar sebuah neuron mempengaruhi yang lain

- STRUKTUR

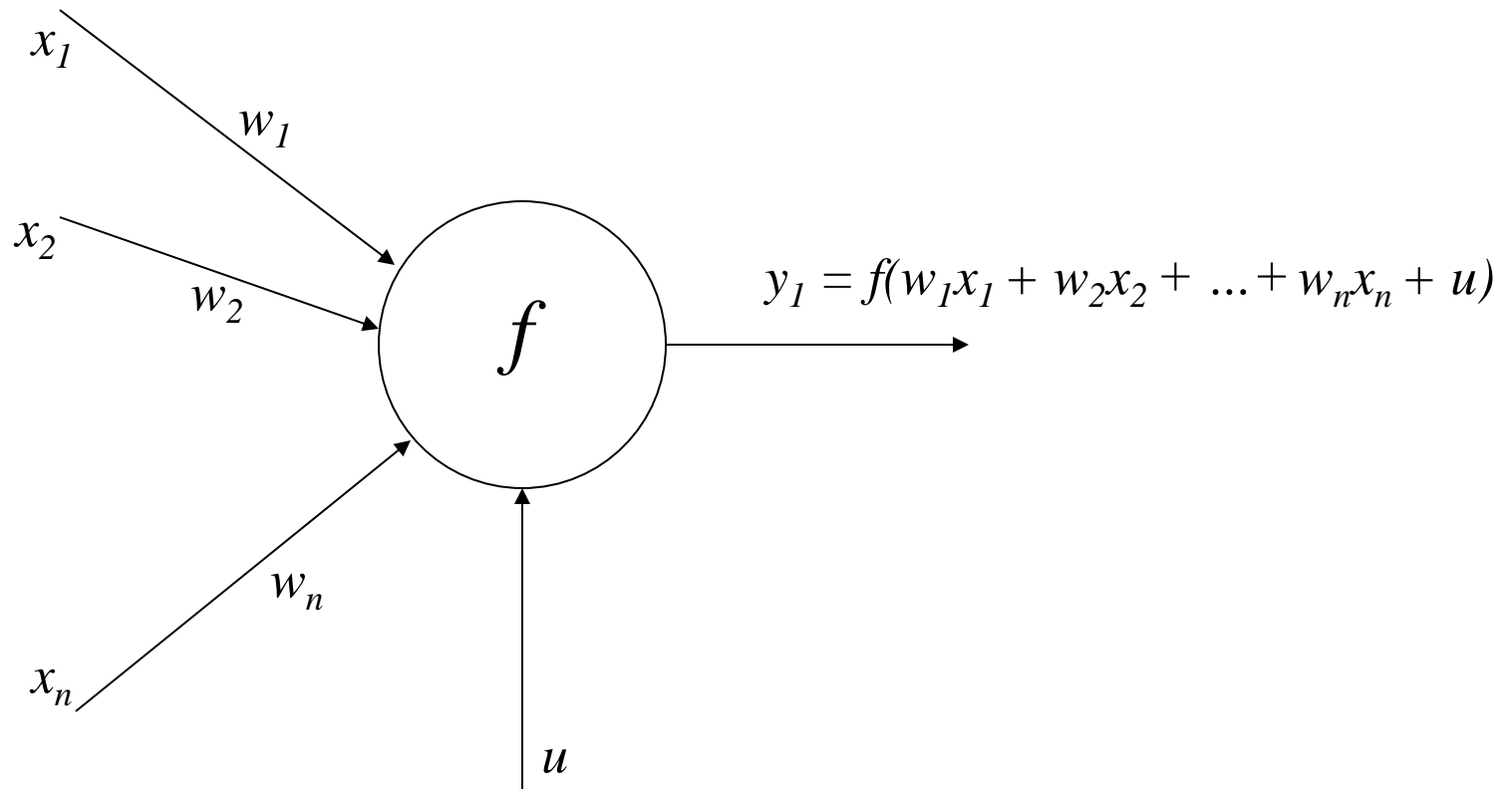
- Bisa feedforward, feedback atau recurrent

Struktur ANN





Model of ANN (Perceptron)





Components of ANN

- Sekumpulan input, x_i ,
- Sekumpulan output yang diharapkan, d_i ,
- Sekumpulan weight, w_i ,
- Suatu nilai bias, u ,
- Learning rate, α ,
- Suatu fungsi aktivasi, f ,
- Sekumpulan output yang dihasilkan, y_i , dan
- Error rate, ε .



Model ANN (Perceptron)

- Bias parameter, u , dapat dianggap sebagai suatu weight dengan input bernilai 1
- Fungsi aktivasi, f , dapat berupa fungsi aktivasi hard limiter atau yang lainnya, seperti:

$$y = \begin{cases} +1 : \text{if } \sum_{i=1}^m w_i x_i + u \geq 0 \\ -1 : \text{if } \sum_{i=1}^m w_i x_i + u < 0 \end{cases}$$



Perceptron Learning Rule

Learning rate > 0
↓

$$w_{i+1} = w_i + \alpha_i (d_i - f(w_i x_i)) x_i$$

- Kalau hasil klasifikasi benar, weights tidak diupdate
- Kalau hasil klasifikasi salah, weight diupdate dengan arah terbalik, sehingga output bergerak menuju output yang diharapkan



Perceptron Learning Step

- Inisialisasi
 - Set $w_0=0$. Kemudian lakukan perhitungan berikut untuk step $i=1,2,\dots$
- Aktivasi
 - Pada step i , aktivasi perceptron dengan mengaplikasikan input vector, x_i , dan output yang diharapkan, d_i
- Perhitungan Output yang Dihasilkan
 - Hitung output yang dihasilkan dari perceptron $y_i=f(w_i x_i)$, dimana f adalah fungsi aktivasi
- Adaptasi Weight Vector
 - Update weight vector dari perceptron
$$w_{i+1} = w_i + \alpha_i (d_i - f(w_i x_i)) x_i$$
- Lanjutkan sampai perbedaan output memenuhi error rate, ϵ .



Dua Types Network Learning

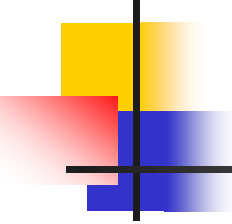
- Sequential mode (online, stochastic, atau per-pattern)
 - Weights diupdate setelah semua pola diproses (Perceptron ada dalam kelompok ini)
- Batch mode (offline, or per-epoch)
 - Weights diupdate sebelum semua pola diproses



Tipe Learning Lain

- Adaline (ADaptive LINear Neuron)
 - Dikembangkan oleh Bernard Widrow and Marcian Hoff (1960)
 - Juga diketahui sebagai least-mean-squares rule, delta rule, dan the Widrow-Hoff rule
 - Meminimalkan error output menggunakan metode gradient descent method (bola digelindingkan turun dari bukit)
- Back Propagation
 - Dikembangkan oleh Rumelhart, Hinton and Williams (1986)
 - Mempunyai dua fase:
 - Fase Forward Pass: Menghitung 'functional signal'
 - Fase Backward Pass: Menghitung 'error signal'

Contoh Perhitungan



27	-65	11	74	33	+1
----	-----	----	----	----	----

Class Label

- Set input untuk suatu neuron sesuai list data di atas
- Hitung nilai output yang dihasilkan
- Update weight
- Catatan:
 - Weight awal ditentukan secara random (0.0 – 1.0, Contoh: 0.2, 0.4, 0.3, 0.1, 0.2)
 - Nilai bias = 1
 - Learning rate = 0.001

Contoh Perhitungan

27	-65	11	74	33	+1
----	-----	----	----	----	----

Class Label

- Hitung nilai output yang dihasilkan

$$y = \begin{cases} +1 : \text{if } \sum_{i=1}^m w_i x_i + u \geq 0 \\ -1 : \text{if } \sum_{i=1}^m w_i x_i + u < 0 \end{cases}$$

- $0.2*27+0.4*(-65)+0.3*11+0.1*74+0.2*33+1$
- $5.4-26.0+3.3+7.4+6.6+1=-2.3$ (It is <0)
- $y = -1$ (Berbeda dari Class Label dari data)

Contoh Perhitungan

27	-65	11	74	33	+1
----	-----	----	----	----	----

Class Label

- Update Weight

$$w_{i+1} = w_i + \alpha_i (d_i - f(w_i x_i)) x_i$$

- w1: $0.2 + 0.001 * (1 - (-1)) * 27 = 0.2 + 0.054 = 0.254$
- w2: $0.4 + 0.001 * (1 - (-1)) * (-65) = 0.4 - 0.130 = 0.270$
- w3: $0.3 + 0.001 * (1 - (-1)) * 11 = 0.3 + 0.022 = 0.322$
- w4: $0.1 + 0.001 * (1 - (-1)) * 74 = 0.1 + 0.147 = 0.247$
- w5: $0.2 + 0.001 * (1 - (-1)) * 33 = 0.2 + 0.066 = 0.266$



Latihan

32	12	25	-15	20	-1
----	----	----	-----	----	----

Class Label

- Set input untuk suatu neuron sesuai list data di atas
- Hitung nilai output yang dihasilkan
- Update weight
- Catatan:
 - Weight awal ditentukan secara random (0.0 – 1.0)
 - Nilai bias = 1
 - Learning rate = 0.002



Neural Networks Dengan SAS EM

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



Pemodelan Dengan Perbandingan

- Pilih tombol Neural Network pada toolbar Model, drag pada workspace
- Koneksikan dengan Data Partition
- Pilih tombol Model Comparison pada toolbar Assess, drag pada workspace
- Koneksikan dengan kedua model yang sudah dibuat (Decision Tree yang sudah dilakukan sebelumnya dan Neural Network)
- Run dan lihat hasil pemodelan dan bandingkan antar kedua model



Pemodelan Dengan Mengubah Parameter Pemodelan

- Pilih tombol Neural Network pada toolbar Model, drag pada workspace
- Koneksikan dengan Data Partition
- Tekan tombol titik tiga (...) pada parameter Network di Property
- Ubah Nilai Target Layer Activation Function menjadi Identity, klik OK
- Koneksikan model yang baru dibuat ke Model Comparion
- Run, dan lihat hasilnya serta bandingkan



Clustering

Yudi Agusta, PhD

Artificial Intelligence, Lecture 10 11 12 13 14

Copyright © Yudi Agusta, PhD 2024



Hierarchical Clustering

- Melakukan pengelompokan data dengan membagi data ke setiap grup dalam beberapa step
- Ada dua metode yang bisa digunakan:
 - Agglomerative Methods: Melakukan pengelompokan dengan menggabungkan data yang satu dengan data yang lain. Proses berjalan dari data dalam jumlah N menjadi satu cluster utama
 - Devisive Methods: Melakukan pengelompokan dengan memilah kelompok-kelompok data menjadi bagian yang lebih kecil. Proses berjalan dari satu cluster menjadi N cluster, dimana N adalah jumlah data

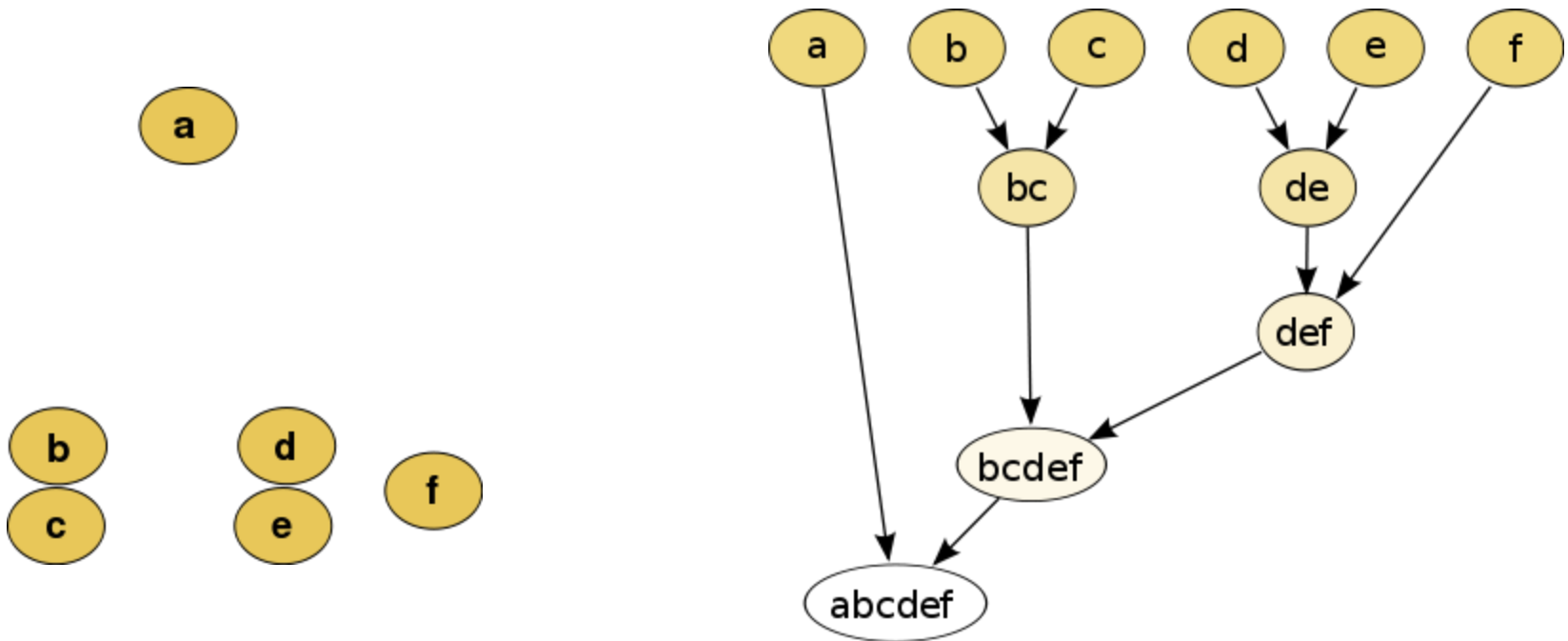


Similarity Measure

- Tingkat kemiripan data dalam proses hierarchical clustering bisa dilakukan dengan menggunakan konsep:
 - Single Linkage: keterikatan ditentukan dengan memilih jarak minimum antara suatu data dengan data dari suatu kelompok
 - Complete Linkage: keterikatan ditentukan dengan memilih jarak maksimum antara suatu data dengan data dari suatu kelompok
 - Average Linkage: keterikatan ditentukan dengan menghitung rata-rata jarak antara suatu data dengan data dari suatu kelompok
- Adapun jarak antar data bisa dihitung dengan berbagai konsep jarak, seperti konsep jarak dalam Euclidean distance space
- Untuk mempermudah, proses clustering bisa didukung dengan pembuatan matriks similarity antar data

Dendogram

Merupakan suatu bentuk representasi dari hasil pemodelan dengan metode hierarchical clustering





Partitional Clustering

- Merupakan suatu tipe clustering yang melakukan analisis dengan secara berulang membagi data dari satu kelas ke kelas yang lain sampai fungsi tujuan (objective function) terminimalisasi
- Metode yang tercakup:
 - C-Means & Fuzzy C-Means: Merupakan metode klasifikasi tanpa supervise dengan jumlah kelas perlu untuk ditentukan di awal proses klasifikasi
 - Mixture Modelling: Merupakan metode klasifikasi tanpa supervise yang komplek dengan jumlah kelas ditentukan bersamaan dengan proses penentuan karakteristik kelas



Non-Probability-Based Clustering

- Contoh: Metode Fuzzy C-Means Method
- Dikembangkan oleh Dunn (1974) dan selanjutnya oleh Bezdek (1981)
- Meminimalkan fungsi tujuan (objective function) yang mencakup sekumpulan membership function (u_{ik}) dan sekumpulan cluster center (v_i)

$$J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ik})^m d(x_k, v_i)^2$$

- Membership function u_{ik} adalah derajat keanggotaan data k ke kelas i



Parameter Setting

- Fungsi Tujuan (Objective Function)

$$J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ik})^m d(x_k, v_i)^2$$

- Parameters:

- m bobot pemangkat (>1.0)
- N jumlah data
- c jumlah class
- x_k data ke k
- v_i class center ke i
- u_{ik} membership function data ke k ke class ke i
- $d(x_k, v_i)$ jarak antara data x_k dan class center v_i



Proses Minimalisasi

- Minimalisasi langsung dari fungsi tujuan (objective function), $J_m(U, v)$, diketahui sangat sulit
- Dilakukan secara parsial dengan meminimalkan secara bergantian class center, $v_{i/}$ dan kemudian membership function, u_{ik}
- Proses dilakukan secara berulang sampai fungsi tujuan (objective function), $J_m(U, v)$, diminimalkan



Other Definitions

- $d(x_k, v_i)$, jarak antara data x_k dengan class center v_i ,
 - Dapat mengambil bentuk apa saja termasuk ukuran jarak (similarity measure) Manhattan (L_1), Euclidean (L_2) atau Mahallanobis
 - Untuk jarak Euclidean, jarak dihitung dengan:

$$d(x_k, v_i) = \sum_{k=1}^N (x_{kj} - v_{ij})^2$$

- Jumlah class harus diketahui sebelum proses minimalisasi dilakukan
- Jumlah class bisa dihitung menggunakan metode seperti:
 - Partition Entropy (PE) diusulkan oleh Bezdek (1981),
 - GAP Statistics oleh Tibshirani, Walther and Hastie (2000), atau
 - Elbow Criterion.
 - Tetapi mereka tidak terlalu bagus, karena cara menghitungkan berbeda konsep dengan perhitungan proses minimiliasi yang dilakukan



Algoritma Minimalisasi

- LANGKAH 1: Pilih ukuran jarak R^p . Tentukan c dan m . Inisialisasi nilai $U^{(0)} \in M$. Tentukan $l=0$.
- LANGKAH 2: Hitung class center $v^{(l)}$ sehingga

$$J(U^{(l)}, v^{(l)}) = \min_{v \in R^{cp}} J(U^{(l)}, v)$$

- LANGKAH 3: Update U : Hitung $U^{(l+1)}$ sehingga

$$J(U^{(l+1)}, v^{(l)}) = \min_{U \in M} J(U, v^{(l)})$$

- LANGKAH 4: Bandingkan $U^{(l)}$ and $U^{(l+1)}$ melalui penggunaan matriks: jika $||U^{(l+1)} - U^{(l)}|| \leq \varepsilon$ maka berhenti, jika tidak $l=l+1$ kembali ke LANGKAH 2




Rumus Perhitungan

- Perhitungan Class Center

$$v_i^{(l)} = \frac{\sum_{k=1}^N x_{ik}}{N}$$

- Perhitungan Membership Function

$$u_{ik}^{l+1} = \sum_{j=1}^c \left(\frac{d(x_k, v_i)}{d(x_k, v_j)} \right)^{-\frac{2}{m-1}}$$




Mixture Modelling

- Memodel data di dalam suatu dataset dengan jumlah kelas yang belum diketahui menjadi model dengan jumlah kelas tertentu
- Merupakan metode clustering berbasis probabilitas
- Menggunakan distribusi statistic sebagai model dari masing-masing class
- Menggunakan metode tambahan untuk menentukan jumlah class yang paling sesuai seperti metode MML, AIC, BIC atau yang lainnya
- Proses untuk menentukan jumlah class dan menemukan karakteristik dari masing-masing class dapat dilakukan secara bersamaan



Parameter Setting

- Fungsi tujuan (objective function)

$$f(x | M, \pi_1, \dots, \pi_M, \vec{\theta}_1, \dots, \vec{\theta}_M) = \sum_{m=1}^M \pi_m \times f_m(x | \theta_m)$$

- Parameters:

- M jumlah class
- π_m relative weight dari class m
- $f_m(x/\theta_m)$ distribusi probabilitas dari class m
- θ_m parameter yang tercakup di dalam distribusi probabilitas class



Proses Klasifikasi

- Mencakup dua tipe proses:
 - Estimasi Parameter: proses untuk mengestimasi parameter dari distribusi statistik dari setiap kelas
 - Pemilihan model: proses memilih model yang paling sesuai untuk dataset yang dianalisa. Bagian utamanya adalah memilih jumlah class yang paling sesuai
- Kedua proses dapat dilakukan secara simultan



Metode Estimasi Parameter

- Maximum Likelihood (ML): metode yang paling banyak digunakan untuk estimasi parameter
- Minimum Message Length (MML): metode yang paling lengkap dalam melakukan estimasi
- Minimum Expected Kullback-Leibler Distance (MinEKL)
- Markov Chain Monte Carlo (MCMC)
- Maximum A Posterior (MAP): mirip dengan MML tetapi sensitive terhadap jumlah data yang dianalisa



Metode Pemilihan Model

- Minimum Message Length (MML): metode yang paling lengkap dalam pemilihan model
- Minimum Description Length (MDL): secara konsep sama dengan MML
- Bayesian Information Criterion (BIC): secara rumus, sama dengan MDL
- Akaike Information Criterion (AIC): Metode yang tidak terlalu bagus dalam pemilihan model
- Bootstrapped likelihood ratio criterion: Metode pemilihan model berbasis statistik



Kombinasi Kedua Metode

- Estimasi Parameter ML dengan Pemilihan Model AIC
- Estimasi Parameter ML dengan Pemilihan Model BIC
- Estimasi Parameter ML dengan Pemilihan Model Bootstrapped
- Estimasi Parameter MML dengan Pemilihan Model MML
- Estimasi Parameter MAP dengan Pemilihan Model Bayesian Networks Tanpa Supervisi



Minimisation Procedure

- LANGKAH 1: Menentukan parameter awal seperti jumlah class
- LANGKAH 2: Memilih model dan mengestimasi parameters
 - LANGKAH 2.1: Menentukan data untuk masing-masing class secara random
 - LANGKAH 2.2: Mengestimasi parameters untuk masing-masing class
 - LANGKAH 2.3: Menghitung probabilitas setiap data masuk ke dalam masing-masing class
 - LANGKAH 2.4: Jika ada perubahan nilai probabilitas ulang ke LANGKAH 2.2, jika tidak dilanjutkan ke LANGKAH 3
- LANGKAH 3: Bandingkan nilai fungsi tujuan (objective function) model yang sedang dihitung dengan model yang tersimpan. Kalau nilainya lebih kecil dari model yang tersimpan, update model yang baru sebagai model terpilih. Lanjutkan dengan jumlah class yang lain.



Parameter Estimation for Class Model

- Untuk data continuous: Luas Lantai, Jumlah ART, Umur, Pendapatan dan yang sejenisnya
 - Menggunakan asumsi Distribusi Normal
 - Data continuous (luas lantai, jumlah ART, umur atau pendapatan) dibentuk sebagai model distribusi Normal dengan rata-rata dan standar deviasi sebagai parameter
- Untuk data categorical: Jenis Lantai, Jenis WC, Jenis Kelamin, Pendidikan dan yang sejenisnya
 - Menggunakan asumsi Distribusi Multinomial
 - Data kategori (jenis lantai, jenis WC, jenis kelamin atau pendidikan) dibentuk sebagai model distribusi Multinomial dengan probabilitas per kategori data digunakan sebagai parameter – probabilitas semen, probabilitas tanah untuk jenis lantai, probabilitas WC sendiri, probabilitas WC bersama untuk jenis WC, probabilitas laki-laki, probabilitas perempuan untuk data jenis kelamin atau probabilitas S1, probabilitas S2, dan probabilitas tingkat pendidikan lainnya untuk data pendidikan



Distribusi Normal

- Fungsi Likelihood

$$f(x | \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Estimasi Parameter

$$\mu = \frac{\sum_{i=1}^N x_i}{N} \quad \sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \mu)^2}{(N-1)}}$$



Distribusi Multinomial

- Fungsi Likelihood

$$f(n_1, n_2, \dots, n_M \mid p_1, p_2, \dots, p_M) = p_1^{n_1} p_2^{n_2} \dots p_M^{n_M}$$

- Estimasi Parameter

$$p_m = \frac{(n_m + 1/2)}{(N + M/2)}$$



Pemilihan: MML

- MML: Minimum Message Length
- MML merupakan teknik pengestimasiian suatu titik dan pemilihan model berbasis teori Bayesian dan teori informasi serta bersifat invariance.
- MML menggabungkan, tidak hanya data yang dimiliki (likelihood) yang disusun berbasis model yang dipilih, tetapi parameter/model yang dipilih sebagai modelnya (prior probability)
- Dalam penjelasan yang lebih umum, dilihat dari data yang dimiliki untuk tujuan pemodelan, jumlah data yang tersedia umumnya belum tentu mewakili semua data yang seharusnya ada. Untuk keperluan tersebut kombinasi antara likelihood dan prior probability diperlukan
 - Prior probability bisa diartikan sebagai wilayah parameter yang umumnya berasal dari pengetahuan dalam suatu kepakaran



Minimum Message Length

- Ide dasar dari MML adalah memilih sebuah model yang meminimalkan total dari message length yang terdiri dari dua bagian:
 - Bagian pertama berisikan encoding untuk model yang dipilih
 - Bagian kedua berisikan encoding untuk data yang dikompresi berdasarkan model yang dipilih tersebut
- Untuk tujuan tersebut, kondisi dalam teori Bayesian direpresentasikan dalam bentuk message length sebagai berikut

$$MessLen = -\log_2(P(H)) - \log_2(P(D | H))$$



Metode Clustering Lainnya

- Self Organizing Map
- Quality Threshold Clustering
- Locality Sensitive Hashing
- Algoritma Rock
- Hierarchical Frequent-Term Base Clustering
- Suffix Tree Clustering
- Single Pass Clustering
- Neighborhood Clustering
- Sequence Clustering
- Spectral Clustering
- Clustering on Frequent Tree



Latihan

	Var1	Var2
X1	13	13
X2	45	58
X3	9	80
X4	45	14
X5	34	90
X6	46	88
X7	15	93
X8	15	94
X9	42	39
X10	30	56

- Bagi data menjadi dua kelompok secara random
- Lakukan iterasi pertama k-means ($m=2$)
- Jelaskan apakah ada perpindahan data dari kelompok yang satu ke kelompok yang lainnya



Contoh Perhitungan

	Var1	Var2
X1	13	13
X2	45	58
X3	9	80
X4	45	14
X5	34	90
X6	46	88
X7	15	93
X8	15	94
X9	42	39
X10	30	56

- Bagi data menjadi dua kelompok secara random
- Kelompok 1: X1, X3, X4, X7, X9
- Kelompok 2: X2, X5, X6, X8, X10

Contoh Perhitungan

	Var1	Var2
X1	13	13
X2	45	58
X3	9	80
X4	45	14
X5	34	90
X6	46	88
X7	15	93
X8	15	94
X9	42	39
X10	30	56

- Lakukan iterasi pertama k-means ($m=2$)
- Kelompok 1:
 - Cluster Center:
 - $v11: (13+9+45+15+42)/5=24.8$
 - $v12: (13+80+14+93+39)/5=47.8$
 - Membership Function:
 - X1:
 - $d11 = \text{SQRT}((x11-v11)^2+(x12-v12)^2)=35.75$
 - $d12 = \text{SQRT}((x11-v21)^2+(x12-v22)^2)=67.55$
 - $u11 = 1/((d11/d11)^2+(d11/d12)^2)=0.77$
 - $u12 = 1/((d12/d11)^2+(d12/d12)^2)=0.23$
 - Karena $u11 > u12$ maka X1 tetap di Kelompok 1
 - X3: ...