

BAB 2

LANDASAN TEORI

2.1. Konsep Dasar Pembangunan Perangkat Lunak

Proses pembangunan perangkat lunak meliputi analisis dan desain perangkat lunak, implementasi, pengujian perangkat lunak.

2.1.1. Analisis dan Desain Perangkat Lunak

Analisis dan desain perangkat lunak adalah proses yang digunakan untuk membuat (*automated*) sistem perangkat lunak. Analisis dan desain perangkat lunak juga dapat diartikan sebagai bagian (*awal*) dari proses panjang rekayasa perangkat lunak yang memiliki tujuan antara lain:

1. Menghasilkan perangkat lunak yang bebas kesalahan (*fault-free software*)
2. Selesai tepat waktu, sesuai dengan anggaran
3. Memenuhi kebutuhan pemakai

Analisis digunakan untuk mengerti *business problem* termasuk tujuan, peranan dan kebutuhan. Desain akan menghasilkan sebuah "*blueprint*" solusi atas *problem* yang dinyatakan dalam analisis.

2.1.1.1. Pendekatan-Pendekatan Analisis Sistem

Secara mendasar, analisis sistem adalah mengenai pemecahan masalah. Beberapa pendekatan analisis sistem yang populer adalah analisis terstruktur (*structured analysis*), teknik informasi (*information engineering*), *discovery prototyping*, dan analisis berorientasi objek (*object-oriented analysis*) (Whitten, 2004).

Analisis terstruktur adalah salah satu pendekatan formal pertama untuk analisis sistem informasi. Analisis terstruktur fokus pada aliran data melalui proses-proses bisnis dan perangkat lunak. Analisis ini disebut *process centered* karena pemakaian teknik ini adalah blok pembangunan proses di dalam kerangka kerja sistem informasi. Analisis terstruktur sederhana dalam konsep. Para analis sistem menggambar serangkaian model proses yang disebut diagram aliran data (*data flow diagram*) yang mengilustrasikan proses-proses yang ada dan atau diusulkan dalam sebuah sistem bersama dengan input, output, dan file mereka. Model-model tersebut menunjukkan aliran data diantara dan melalui proses-proses dan menunjukkan tempat-tempat data disimpan. Pada akhirnya model-model proses ini berperan sebagai cetak biru (*blue print*) bagi proses-proses bisnis untuk diimplementasikan dan perangkat lunak untuk dibeli atau dikonstruksi.

Teknik informasi fokus pada struktur data tersimpan dalam sebuah sistem. Teknik Informasi dan analisis terstruktur mencoba mensinkronkan model-model data dan proses. Dua pendekatan tersebut berbeda hanya pada model mana yang digambar dulu. Teknik Informasi menggambar model-model data dahulu, sedangkan analisis terstruktur menggambar model-model proses dahulu.

Analisis berorientasi objek adalah sesuatu yang terbaru. Teknologi-teknologi objek muncul untuk menghilangkan pemisahan *artificial* data dan proses. Sebaliknya, data dan proses-proses spesifik yang membuat, membaca, memperbaharui dan menghapus data itu diintegrasikan ke dalam konstruksi yang disebut *object/objek*. Satu-satunya cara untuk membuat, membaca,

memperbaharui atau menghapus data objek (disebut properti) adalah melalui salah satu dari proses pelekatan (*embedded*) yang disebut metode. Tren terhadap penggunaan teknologi objek untuk membangun aplikasi-aplikasi baru menciptakan kebutuhan akan pendekatan analisis sistem pelengkap, *object-oriented analysis* (OOA)/analisis berorientasi objek. OOA telah menjadi populer sehingga standar pemodelan berkembang disekitarnya. *Unified Modelling Language* (UML) menyediakan sintaks grafis untuk keseluruhan rangkaian model-model objek (*object models*). UML mendefinisikan beberapa tipe diagram yang berbeda yang secara kolektif memodelkan sebuah sistem informasi atau aplikasi dalam artian objek-objek.

2.1.2. Implementasi

Implementasi mengkontruksi sistem informasi baru dan menempatkannya ke dalam operasi. Selama implementasi perangkat keras dan perangkat lunak yang baru diinstal. Dalam kontruksi hal yang perlu diperhatikan adalah gap antara rancangan (konseptual) dengan implementasi (teknis). Membuat kode yang mudah dimengerti karena kode ini (kemungkinan) akan digunakan orang lain atau akan dikembangkan.

2.1.3. Pengujian Perangkat Lunak

Pengujian Perangkat Lunak (PL) dilakukan dengan tujuan untuk menemukan kesalahan (*fault*) sebanyak mungkin dari PL yang diuji, membuat PL yang diperbaiki setelah diuji menjadi PL yang berkualitas, mengumpulkan kesalahan yang terjadi dan menggunakannya untuk tindakan preventif.

2.2. Konsep Dasar Basis Data

Basis data dapat didefinisikan dengan beberapa cara, antara lain adalah sebagai berikut:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Tujuan dibangunnya sebuah basis data, yaitu:

1. Kecepatan dan kemudahan (*Speed*)
Hal ini berkaitan dengan menyimpan, memanipulasi dan menampilkan kembali data dengan cepat dan mudah.
2. Efisiensi ruang penyimpanan (*Space*)
Penekanannya adalah mengurangi terjadinya redundansi data atau kerangkapan data. Dengan semakin sedikit redundansi maka kebutuhan akan media penyimpanan juga akan semakin sedikit.
3. Keakuratan (*Accuracy*)
Pemanfaatan pengkodean/pembentukan relasi adalah untuk menghindari ketidakakuratan. Masalah ini harus diperhatikan dalam melakukan perancangan sebuah basis data.
4. Ketersediaan (*Availability*)
Pertumbuhan data yang membutuhkan *space* yang besar, maka data harus dipilah sebagai data

master/referensi/transaksi/history. Data bisa diset *on-line*, *off-line*, disebar di banyak tempat, atau direplikasi.

5. Kelengkapan (*Completeness*)

Lengkap adalah relatif, sehingga penambahan data dan atau struktur data dapat diakomodasi.

6. Keamanan (*Security*)

Penentuan siapa yang berhak mengakses obyek-obyek tertentu dan operasi apa yang diperbolehkan.

7. Kebersamaan pemakai (*Shareability*)

Pemakai basis data tidak terbatas pada satu pemakai saja, sehingga perlu diperhatikan bagaimana dapat menciptakan mekanisme agar sebuah basis data dapat digunakan oleh banyak orang tanpa menyebabkan kekacauan pada basis data.

Secara fisik pengelolaan basis data dilakukan oleh sebuah perangkat lunak yang disebut DBMS yang akan menentukan bagaimana data diorganisir, disimpan, diubah dan diambil kembali.

2.3. Logika Fuzzy

Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Sebagai contoh:

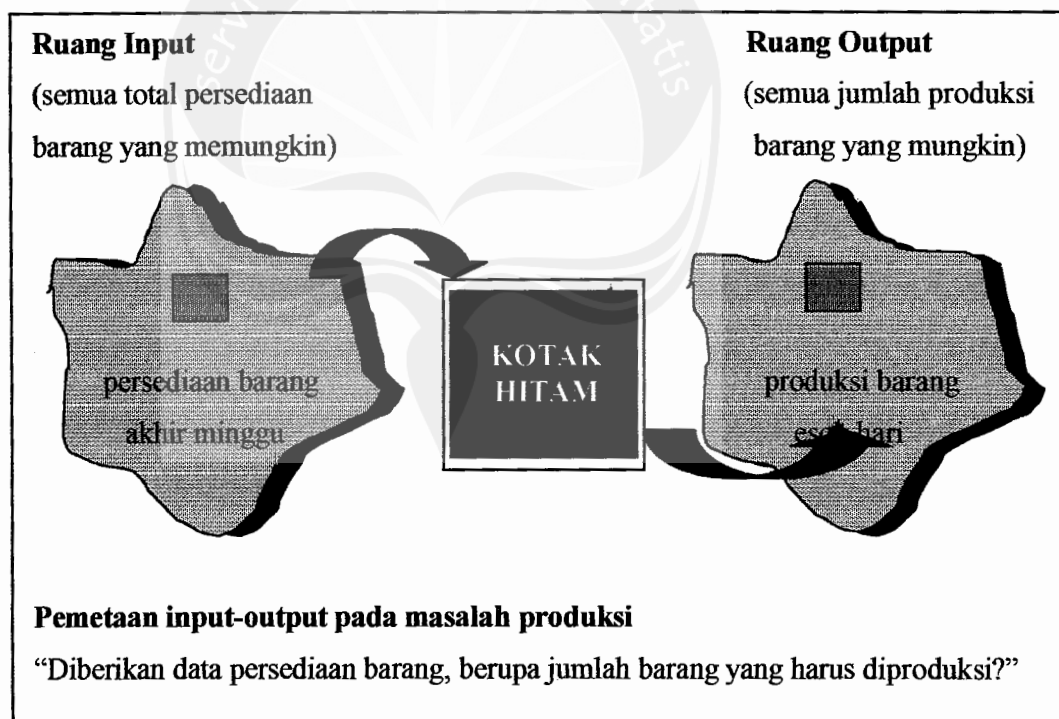
1. Manajer pergudangan mengatakan pada manajer produksi seberapa banyak persediaan barang pada akhir minggu ini, kemudian manajer produksi akan menetapkan jumlah barang yang harus diproduksi esok hari. Inputnya adalah banyaknya persediaan barang pada akhir minggu

ini dan outputnya adalah jumlah barang yang harus diproduksi esok hari;

2. Pelayan restoran memberikan pelayanan terhadap tamu, kemudian tamu akan memberikan tip yang sesuai atas baik tidaknya pelayanan yang diberikan. Inputnya adalah pelayanan dan outputnya tip;

3. Anda mengatakan pada saya seberapa sejuk ruangan yang anda inginkan, saya akan mengatur putaran kipas yang ada pada ruangan ini. Inputnya adalah keinginan seberapa sejuk ruangan dan outputnya adalah pengaturan kipas.

Salah satu contoh pemetaan suatu input-output dalam bentuk grafis adalah:



Gambar 2.1. Contoh pemetaan input-output (Kusumadewi, 2004)

Ada beberapa alasan mengapa orang menggunakan logika fuzzy, antara lain:

1. Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran fuzzy sangat sederhana dan mudah dimengerti.
2. Logika fuzzy sangat fleksibel.
3. Logika fuzzy memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika fuzzy mampu memodelkan fungsi-fungsi non linier yang sangat kompleks.
5. Logika fuzzy dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika fuzzy dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. Logika fuzzy didasarkan pada bahasa alami.

2.4. Data Mining

Data mining pada dasarnya adalah perolehan pengetahuan dari data yaitu ekstraksi informasi atau pola yang menarik (tidak sepele, implisit, tak diketahui sebelumnya, mungkin bermanfaat) dari data di dalam database yang besar (Han, 2001)

Data-data yang diperoleh melalui pemrosesan awal data (*data preprocessing*). Tugas utama pemrosesan awal data adalah:

1. Pembersihan Data

Pembersihan data merupakan usaha untuk memenuhi nilai-nilai yang hilang, menghaluskan *noise* sambil mengidentifikasi *outliers* dan membetulkan ketidakkonsistensian dalam data. *Missing value* merupakan nilai-nilai yang tidak bisa terdeteksi karena tidak terpenuhinya nilai atribut yang bersangkutan.

Noisy data adalah kesalahan yang acak (*random error*) atau varian di dalam variabel yang terukur.

Inconsistent data (data yang tidak konsisten): ketidakkonsistenan data dapat terjadi selama pemasukan data, pengintegrasian data atau adanya *redundancies*.

2. Integrasi Dan Transformasi Data

1) Integrasi data

Menyatukan data dari berbagai sumber data ke dalam bentuk *data store* yang koheren seperti pada *warehousing data*.

Hal-hal yang perlu dipertimbangkan dalam pengintegrasian data:

a. *Schema integration*: tahap ini bisa dimanipulasi berdasarkan pengidentifikasian masalah yang sebenarnya.

b. *Redundancy*: disebabkan ketidakkonsistenan dalam penamaan. Beberapa *redundancy* dapat diketahui dalam analisa korelasi. Duplikasi dalam atribut harus dapat dideteksi pada *tuple level*.

c. Mendeteksi dan meresolusi konflik nilai data: nilai dari sebuah atribut dari sumber yang berbeda harus dibedakan. Hal ini dapat dilakukan pada pembedaan representasi, *scaling* dan *encoding*.

Pengintegrasian data dari sumber yang berbeda dapat membantu dalam mengurangi dan mencegah kelebihan dan ketidakkonsistenan dalam menghasilkan data set.

2) Transformasi data

Data ditransformasi atau disatukan dalam formulir yang bertujuan untuk *mining*. Transformasi data meliputi:

- a. *Smoothing*: menghilangkan *noise* dari data, dengan cara *binning*, klastering, dan regresi.
- b. Agregasi: data diringkas. Langkah ini dilakukan dalam pembuatan konstruksi kubus data untuk menganalisis data dari titik-titik yang banyak.
- c. Generalisasi data: dilakukan ketika data primitif digantikan oleh tingkatan konsep yang lebih tinggi
- d. Normalisasi: dilakukan ketika atribut data diskala dalam range yang lebih kecil.
- e. Kontruksi atribut: dilakukan ketika ada atribut-atribut baru disusun dan ditambahkan dari atribut tambahan yang digunakan untuk membantu dalam proses *mining*.

3. Reduksi Data

Dilakukan untuk mengurangi representasi dari data set sehingga dihasilkan volume yang lebih kecil.

Teknik mereduksi data:

- 1) Agregasi kubus data: diterapkan pada saat data diagregasi dalam pembuatan kubus data.
- 2) Reduksi dimensi: dilakukan ketika data tidak relevan, atribut dan dimensi yang

berlebihan terdeteksi dan bisa dihilangkan.

- 3) Kompresi data: dilakukan pada saat mekanisme *encoding* untuk mengurangi ukuran data set.
- 4) Reduksi jumlah: dilakukan ketika data dapat digantikan oleh alternatif lain berupa representasi data yang lebih kecil seperti model parametrik atau menggunakan *nonparametric method* seperti klustering, sampling, dan menggunakan histogram.
- 5) Pendiskritasian dan generasi konsep hirarki: ketika nilai data awal untuk atribut digantikan oleh range atau konsep yang mempunyai level lebih tinggi. Konsep hirarki memungkinkan untuk melakukan *mining* pada banyak level dari berbagai abstraksi dan menggunakan teknik yang lebih tinggi

Salah satu fungsionalitas dari data mining adalah Analisis klaster (analisis pengelompokan). Klustering adalah klasifikasi tanpa supervisi: tidak ada kelas yang didefinisikan sebelumnya. Tujuan klaster adalah mengelompokkan objek-objek berdasarkan kesamaan karakteristik di antara objek-objek tersebut dan ketaksamaan pada objek-objek dalam klaster lainnya. Analisis klaster adalah pengelompokan sekumpulan objek data ke dalam klaster-klaster.

Suatu metode klustering yang baik akan menghasilkan klaster kualitas tinggi dengan kemiripan intra-kelas tinggi (homogenitas antar klaster tinggi). Kualitas dari suatu hasil klustering bergantung kepada ukuran kesamaan yang digunakan oleh metode dan

implementasinya. Kualitas dari suatu metode klastering adalah juga diukur dengan kemampuannya untuk menemukan beberapa atau seluruh pola tersembunyi.

Pendekatan Klastering utama adalah:

1. Algoritma pemisahan: membangun berbagai partisi dan kemudian mengevaluasi partisi tersebut dengan beberapa kriteria
2. Algoritma hierarki: membuat suatu penguraian secara hierarkikal dari himpunan data (atau objek-objek) menggunakan beberapa kriteria
3. Berbasis densitas: berdasarkan konektivitas dan fungsi densitas
4. Berbasis grid: berdasarkan suatu struktur granularitas multi-level
5. Berbasis model: suatu model dihipotesakan untuk masing-masing kluster dan ide untuk mencari *best fit* dari model tersebut untuk masing-masing yang lain

Contoh dari aplikasi klastering:

1. *Marketing*: membantu marketer menemukan grup berbeda dalam basis *customer* mereka, dan kemudian menggunakan pengetahuan ini untuk mengembangkan target program marketing.
2. *Land use*: mengenal daerah *land use* yang sama dalam suatu *database* pengamatan bumi
3. *Insurance*: mengenali kelompok pemegang polis asuransi motor dengan suatu klaim rata-rata biaya tinggi
4. *City-planning*: mengenali kelompok perumahan menurut tipe, nilai, dan lokasi geografisnya.

5. *Earth-quake studies*: pengamatan pusat gempa bumi harus dikelompokkan bersama-sama dengan bencana alam

2.5. Fuzzy Klustering

Teori himpunan fuzzy akan memberikan jawaban terhadap suatu masalah yang mengandung ketidakpastian. Pada beberapa kasus khusus, seperti nilai keanggotaan yang kemudian akan menjadi 0 atau 1, teori dasar tersebut akan identik dengan teori himpunan biasa, dan himpunan fuzzy akan menjadi himpunan crisp tradisional.

Yang menjadi pertanyaannya adalah: "Seberapa besar kekaburan suatu himpunan fuzzy?", atau "Seberapa sama antara 2 himpunan fuzzy?" Pertanyaan pertama akan melahirkan suatu konsep ukuran fuzzy (*fuzzy measure*), dan pertanyaan yang kedua akan melahirkan konsep ukuran kesamaan (*similarity measure*).

2.5.1. Ukuran Fuzzy

Ukuran fuzzy menunjukkan derajat kekaburan dari himpunan fuzzy. Secara umum ukuran kekaburan dapat ditulis sebagai suatu fungsi:

$$F: P(X) \rightarrow R$$

Dengan $P(X)$ adalah himpunan semua subset dari X . $f(A)$ adalah suatu fungsi yang memetakan subset A ke karakteristik dengan kekaburannya.

Dalam mengukur nilai kekaburan, fungsi f harus mengikuti hal-hal berikut ini:

1. $f(A) = 0$ jika dan hanya jika A adalah himpunan *crisp*.
2. jika $A \subset B$, maka $f(A) \leq f(B)$. Disini, $A \subset B$ berarti B lebih kabur dibanding A (atau A lebih tajam

dibanding B). Relasi ketajaman $A < B$ didefinisikan dengan:

$\mu_A[x] \leq \mu_B[x]$, jika $\mu_B[x] \leq 0,5$; dan

$\mu_A[x] \geq \mu_B[x]$, jika $\mu_B[x] \geq 0,5$

3. $f(A)$ akan mencapai maksimum jika dan hanya jika A benar-benar kabur secara maksimum. Tergantung pada interpretasi derajat kekaburan, nilai fuzzy maksimal biasanya terjadi pada saat $\mu_A[x] = 0,5$ untuk setiap x .

2.5.2. Indeks Kekaburan

Indeks kekaburan adalah jarak antara suatu himpunan fuzzy A dengan himpunan *crisp* C yang terdekat. Himpunan *crisp* C terdekat dari himpunan fuzzy A dinotasikan sebagai $\mu_C[x] = 0$, jika $\mu_A[x] \leq 0,5$, dan $\mu_C[x] = 1$, jika $\mu_A[x] \geq 0,5$. ada 3 kelas yang paling sering digunakan dalam mencari indeks kekaburan, yaitu:

1. Hamming distance.

$$f(A) = \sum |\mu_A[x] - \mu_C[x]| \text{ atau}$$

$$f(A) = \sum \min [\mu_A[x], 1 - \mu_A[x]]$$

2. Euclidean distance.

$$f(A) = \{\sum [\mu_A[x] - \mu_C[x]]^2\}^{1/2}$$

3. Minkowski distance.

$$f(A) = \{\sum [\mu_A[x] - \mu_C[x]]^w\}^{1/w}$$

dengan $w \in [1, \infty]$.

2.6. Fuzzy C-Means (FCM)

Fuzzy klastering adalah salah satu teknik untuk menentukan klaster optimal dalam suatu ruang vektor yang didasarkan pada bentuk normal Euclidian untuk jarak antar vektor. Fuzzy klastering sangat berguna

bagi pemodelan fuzzy terutama dalam mengidentifikasi aturan-aturan fuzzy (Kusumadewi, 2004).

Ada beberapa algoritma klastering data seperti *K-Mean*, *K-Medoids*, *Clustering Large Application*, klastering berbasis densitas, *Fuzzy C-Means* (FCM). *Fuzzy C-Means* (FCM) adalah suatu teknik pengklasteran data yang mana keberadaan tiap-tiap titik data dalam klaster ditentukan oleh derajat keanggotaan. Teknik ini pertama kali diperkenalkan oleh Jim Bezdek pada tahun 1981.

Konsep dasar FCM, pertama kali adalah menentukan pusat klaster, yang akan menandai lokasi rata-rata untuk tiap-tiap klaster. Pada kondisi awal, pusat klaster ini masih belum akurat. Tiap-tiap titik data memiliki derajat keanggotaan untuk tiap-tiap klaster. Dengan cara memperbaiki pusat klaster dan derajat keanggotaan tiap-tiap data secara berulang, maka akan dapat dilihat bahwa pusat klaster akan bergerak menuju lokasi yang tepat. Perulangan ini didasarkan pada minimasi fungsi obyektif yang menggambarkan jarak dari titik data yang diberikan ke pusat klaster yang terbobot oleh derajat keanggotaan titik data tersebut.

Output dari FCM bukan merupakan *fuzzy inference system*. Namun merupakan deretan pusat klaster dan beberapa derajat keanggotaan untuk tiap-tiap titik data. Informasi ini dapat digunakan untuk membangun suatu *fuzzy inference system*.

2.6.1. Algoritma FCM

Algoritma *Fuzzy C-Means* (FCM) adalah sebagai berikut:

1. Input data yang akan diklaster X , berupa matriks berukuran $n \times m$ (n = jumlah sampel data, m =

atribut setiap data). X_{ij} = data sampel ke-I
($i=1,2,\dots,n$), atribut ke-j ($j=1,2,\dots,m$).

2. Tentukan:

- o Jumlah klaster = c ;
- o Pangkat = w ;
- o Maksimum iterasi = MaxIter ;
- o Error terkecil yang diharapkan = ξ
- o Fungsi obyektif awal = $P_0 = 0$;
- o Iterasi awal = $t = 1$;

3. Bangkitkan bilangan random μ_{ik} , $i = 1,2,\dots,n$; $k = 1,2,\dots,c$; sebagian elemen-elemen matriks partisi awal U.

Hitung jumlah setiap kolom (atribut):

$$Q_j = \sum_{k=1}^c \mu_{ik}$$

dengan $j=1,2,\dots,m$.

Hitung:

$$\mu_{ik} = \frac{\mu_{ik}}{Q_j}$$

4. Hitung pusat klaster ke-k: V_{kj} , dengan $k = 1,2,\dots,c$; dan $j = 1,2,\dots,m$.

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w * X_{ij})}{\sum_{i=1}^n (\mu_{ik})^w}$$

5. Hitung fungsi obyektif pada iterasi ke-t, P_t :

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left(\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right] (\mu_{ik})^w \right)$$

6. Hitung perubahan matriks partisi:

$$\mu_{ik} = \frac{\left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right]^{\frac{-1}{w-1}}}{\sum_{k=1}^c \left[\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right]^{\frac{-1}{w-1}}}$$

dengan $i = 1, 2, \dots, n$; dan $k = 1, 2, \dots, c$.

7. Cek kondisi berhenti:

- o Jika: $(|P_t - P_{t-1}| < \xi)$ atau $(t > \text{MaxIter})$
maka berhenti;
- o Jika tidak: $t = t+1$, ulangi langkah ke-4

2.7. Total Within Cluster Variation (TWCV)

Pada klasterisasi terdapat beberapa algoritma untuk menentukan jumlah klaster valid. Klaster valid adalah nilai minimum klaster yang bernilai valid. Salah satunya adalah *Total Within Cluster Variation* (TWCV), yang dirumuskan sebagai berikut:

$$S^{(k)}(W) = \sum_{i=1}^n W_{ik} \sum_{j=1}^d (x_{ij} - c_{kj})^2$$

Dengan $S^{(k)}(W)$: variasi antar klaster

n : jumlah data

W_{ik} : matriks 0 & 1 yang menunjukkan letak klaster

d : jumlah atribut

x_{ij} : tabel data

C_{kj} : titik pusat klaster

Nilai *Total Within Cluster Variation* (TWCV) dirumuskan sebagai berikut:

$$S(W) = \sum_{k=1}^K S^{(k)} = \sum_{k=1}^K \sum_{i=1}^n W_{ik} \sum_{j=1}^d (x_{ij} - c_{kj})^2$$

Dengan $S(W)$: *Total Within Cluster Variation*

K : jumlah klaster

- n : jumlah data
- W_{ik} : matriks 0 & 1 yang menunjukkan letak klaster
- d : jumlah atribut
- x_{ij} : tabel data
- C_{kj} : titik pusat klaster

2.8. Statistika

Statistika dapat diartikan sebagai ilmu pengetahuan yang mempelajari tentang cara-cara pengumpulan fakta, pengolahan serta analisis pembuatan keputusan dan penarikan kesimpulan yang cukup beralasan berdasarkan fakta dan pengolahan data yang dilakukan (Supangat, M.Si, 2007). Pengertian statistik sebagai metode menurut definisi Croxton dan Cowden: "*statistics may be defined as the collection, presentation, analysis and interpretation of numerical data*". Statistik dalam pengertian metode diperlukan untuk menyusun data yang telah dikumpulkan.

Statistik dalam pengertian sebagai metode dapat dibedakan menjadi 2 macam, yakni:

1. Statistik Deskriptif (*Descriptive Statistics*)

Adalah metode statistik yang menggambarkan sifat-sifat data. Kegiatan statistik di sini berupa kegiatan pengumpulan data, penyusunan data dan penyajian data dalam bentuk-bentuk tabel, grafik-grafik maupun diagram-diagram.

2. Statistik Induktif (*Inductive Statistics*)

Kegiatan statistik yang lebih berupa penggunaan data untuk peramalan maupun penarikan kesimpulan

Analisa data dapat dilakukan dengan ukuran pemusatan dan ukuran penyebaran data.

1. Ukuran pemusatan data menunjukkan di mana suatu kumpulan data memusat. Ukuran pemusatan data antara lain: rata-rata (mean), median, modus. Untuk mean rumus yang digunakan adalah

$$\bar{x} = \frac{\sum_{i=1}^n X_i}{n}$$

Dengan $\bar{x}$: rata-rata

n : jumlah data

X_i : data

2. Ukuran penyebaran data adalah perserakan data individual terhadap nilai rata-rata. Data yang bersifat homogen akan mempunyai penyebaran yang kecil, sedang data yang bersifat heterogen penyebarannya akan besar. Ukuran penyebaran antara lain: jangkauan (*range*), simpangan rata-rata (*mean deviation*), variansi (*variance*), simpangan baku (*standard deviation*)

Ukuran penyebaran yang sering digunakan adalah *standard deviation*. Secara statistik *standard deviation* dirumuskan:

$$S = \sqrt{\frac{\sum n.(X_i - \bar{X})^2}{n-1}}$$

Dengan X_i : Nilai data

$\bar{X}$: Rata-rata nilai

n : jumlah data