

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 Definisi *Artificial Intelligence*

Kecerdasan buatan atau *artificial intelligence* (AI) merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industri dan rumah tangga. Buku ini akan memaparkan berbagai pandangan model dan hasil riset terkini yang perlu dikuasai oleh para akademisi, mahasiswa, pelajar dan praktisi. (Budihartono & Suhartono, 2014:2)

Kecerdasan buatan berasal dari bahasa Inggris “*Artificial intelligence*” atau disingkat *AI*, yaitu *intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan, kecerdasan buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. (Sutojo, Mulyanto, & Suhartono, 2011:1)

Jika dibandingkan dengan kecerdasan alami (kecerdasan yang dimiliki oleh manusia). Kecerdasan buatan memiliki keuntungan komersial, antara lain (Turban): (Sutojo et al., 2011:10)

1. Kecerdasan buatan lebih bersifat permanen. Kecerdasan alami akan cepat mengalami perubahan. Kemampuan kecerdasan buatan tidak akan pernah berubah selama programnya tidak diubah oleh *programmer*. Berbeda dengan kecerdasan alami. Karena sifat manusia yang subjektif, pelupa, dan makin lama makin tua hingga kemampuan berpikirnya berkurang seiring bertambahnya waktu, kemampuan kecerdasan alami cenderung tidak permanen.
2. Kecerdasan buatan lebih mudah diduplikasi dan disebar. Misalnya saja pemerintah membutuhkan 10.000 orang pakar penyakit jantung untuk ditempatkan di seluruh Indonesia. Bayangkan kalau pemerintah harus menyekolahkan anak bangsa sejumlah 10.000 orang, mulai dari SD sampai lulus sarjana kedokteran spesialis penyakit jantung. Waktu yang dibutuhkan minimal 20 tahun. Jika biaya pendidikan 1 orang Rp 100 juta, maka untuk 10.000 orang biaya yang diperlukan adalah Rp 1 triliun. belum lagi kendala jika orangnya meninggal sebelum menjadi sarjana. Kalau sudah jadi sarjana, pemerintah akan kesulitan untuk mendistribusikan mereka ke pelosok-pelosok Indonesia. Hal ini sangat tidak efisien. Sementara itu, untuk kecerdasan buatan, pemerintah cukup membuat 1 sistem pakar penyakit jantung dengan waktu yang *relative* lebih cepat dan biaya yang jauh lebih murah. Proses duplikasi dan pendistribusiannya ke seluruh pelosok tanah air pun sangat mudah.
3. Kecerdasan buatan lebih murah dibandingkan kecerdasan alami. Hal ini tergambar seperti dijelaskan nomor 2.
4. Kecerdasan buatan bersifat konsisten. Misalnya saja telah dibuat sistem pakar hakim pengadilan untuk mengadili kasus-kasus pidana di Indonesia. Untuk

kasus yang sama, solusi dan keputusan yang dibuat oleh kecerdasan buatan tidak pernah berubah. Berbeda dengan kecerdasan alami. Arti kata hakim bisa berubah menjadi “Hubungi Aku Kalau Ingin Menang”. Untuk kasus yang sama, solusi dan keputusan yang dibuat oleh kecerdasan alami bisa berubah-ubah tergantung orang yang terkena kasus telah menghubungi hakim atau tidak.

5. Kecerdasan buatan dapat didokumentasi. Solusi dan keputusan yang dibuat oleh kecerdasan buatan dapat didokumentasi dengan mudah karena disimpan di dalam *hard disk* dan pencarian datanya *relative* lebih mudah dilacak, sedangkan untuk kecerdasan alami, hal ini sangat sulit dilakukan.
6. Kecerdasan buatan dapat mengerjakan pekerjaan lebih cepat dibanding dengan kecerdasan alami. Tentu saja karena kecepatan berpikir dari sebuah prosesor jauh lebih cepat dibanding kecepatan berpikir dari otak manusia.
7. Kecerdasan buatan dapat mengerjakan pekerjaan lebih baik dibanding dengan kecerdasan alami.

Sementara itu, kecerdasan alami memberikan keuntungan sebagai berikut:

1. Kreatif pengetahuan seorang manusia selalu bertambah seiring dengan perkembangan waktu. Sifat bosan manusia pun mengakibatkan ia harus berpikir kreatif untuk mencari solusi-solusi terbaru. Berbeda dengan kecerdasan buatan, penambahan pengetahuan harus dilakukan pada sistem yang telah dibangun.

2. Kecerdasan alami memungkinkan orang menggunakan pengalaman secara langsung, sedang pada kecerdasan buatan harus bekerja dengan *input-input* simbolik.
3. Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

Adapun bidang ilmu yang ada pada kecerdasan buatan atau *artificial intelligence (AI)* sebagai berikut:

- a. Sistem Pakar

Sistem Pakar merupakan cabang dari *artificial intelligence (AI)* yang cukup tua karena sistem ini mulai dikembangkan pada pertengahan 1960. Sistem pakar adalah sebuah sistem yang menggunakan pengetahuan manusia di mana pengetahuan tersebut dimasukkan kedalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah yang biasanya membutuhkan kepakaran dan atau keahlian manusia. Sistem pakar terdiri dari beberapa komponen penting yaitu: Akuisisi Pengetahuan, Basis Pengetahuan (*Knowledge Base*), Mesin Inferensi (*Inference Engine*), Daerah Kerja (*Blackboard*), Antarmuka (*User Interface*), Subsistem Penjelasan (*Explanation Subsystem/ Justifier*), Sistem Perbaikan Pengetahuan (*Knowledge Refining System*), dan Pengguna (*User*). (Sutojo et al., 2011:159)

- b. Jaringan Saraf Tiruan (JST)

Jaringan saraf tiruan (JST) adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak manusia. Jaringan saraf tiruan terdiri dua metode pembelajaran yaitu algoritma

pembelajaran dengan *supervise* dan pembelajaran tanpa *supervise* (jaringan kohonen). (Sutojo et al., 2011:283)

c. Logika *Fuzzy*

Logika *fuzzy* adalah metodologi sistem *control* pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan *PC*, *multi-channel* atau *workstation* berbasis akuisisi *data*, dan sistem *control*. Logika *fuzzy* terdiri dari beberapa metode yaitu Metode Tsukamoto, Metode Mamdani, dan Metode Sugeno. (Sutojo et al., 2011:211)

2.1.2 Definisi Logika *Fuzzy*

Logika *fuzzy* merupakan salah satu komponen pembentuk *soft computing*. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dari penalaran dengan logika *fuzzy* tersebut. (Kusumadewi & Purnomo, 2013:1)

Konsep tentang logika *fuzzy* diperkenalkan oleh Prof. Lotfi Astor Zadeh pada 1962, Logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan *PC*, *multi-channel* atau *workstation* berbasis akuisisi *data*, dan sistem *control*. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya.

Dalam logika klasik dinyatakan bahwa segala sesuatu bersifat *biner*, yang artinya adalah hanya mempunyai dua kemungkinan, “Ya atau Tidak”, “Benar atau Salah”, “Baik atau Buruk” dan lain-lain. Oleh karena itu, sistem ini dapat mempunyai nilai keanggotaan 0 atau 1. Akan tetapi, dalam logika *fuzzy* memungkinkan nilai keanggotaan berada di antara 0 dan 1. Artinya, bisa saja suatu keadaan mempunyai dua nilai “Ya dan Tidak”, “Benar dan Salah”, “Baik dan Buruk” secara bersamaan, namun besar nilainya tergantung pada bobot keanggotaan yang dimilikinya. Bila dibandingkan dengan logika konvensional, kelebihan logika *fuzzy* adalah kemampuannya dalam proses penalaran secara bahasa sehingga dalam perancangannya tidak memerlukan persamaan matematik yang rumit. (Sutojo et al., 2011:211)

2.1.3 Alasan Penggunaan Metode *Fuzzy Logic*

Dalam penelitian ini, penulis menggunakan metode *fuzzy logic*. Adapun alasan penulis memilih menggunakan metode ini adalah antara lain sebagai berikut: (Kusumadewi & Purnomo, 2013:2)

- a. Konsep *fuzzy logic* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
- b. *Fuzzy logic* sangat fleksibel.
- c. *Fuzzy logic* memiliki toleransi terhadap *data-data* yang tidak tepat.
- d. *Fuzzy logic* mampu memodelkan fungsi-fungsi *nonlinear* yang sangat kompleks.

- e. *Fuzzy logic* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
- f. *Fuzzy logic* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
- g. *Fuzzy logic* didasarkan pada bahasa alami.

2.1.4 Dasar-Dasar Logika *Fuzzy*

Adapun beberapa dasar logika *fuzzy* yang harus dipahami dalam memahami logika *fuzzy* yaitu: (Sutojo et al., 2011:212)

- a. Variabel *fuzzy*, yaitu variabel yang akan dibahas dalam suatu sistem *fuzzy*.
- b. Himpunan *fuzzy*, yaitu suatu kelompok yang mewakili suatu keadaan tertentu dalam suatu variabel *fuzzy*. Himpunan *fuzzy* memiliki 2 atribut yaitu: Linguistik dan Numeris. Linguistik adalah nama suatu kelompok yang mewakili suatu keadaan tertentu dengan menggunakan bahasa alami, dan Numeris adalah suatu nilai yang menunjukkan ukuran dari suatu variabel.
- c. Semesta pembicaraan, yaitu seluruh nilai yang diizinkan untuk dioperasikan dalam suatu variabel *fuzzy*
- d. *Domain* himpunan *fuzzy*, yaitu seluruh nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*.

Adapun beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* yaitu: (Kusumadewi & Purnomo, 2013:7)

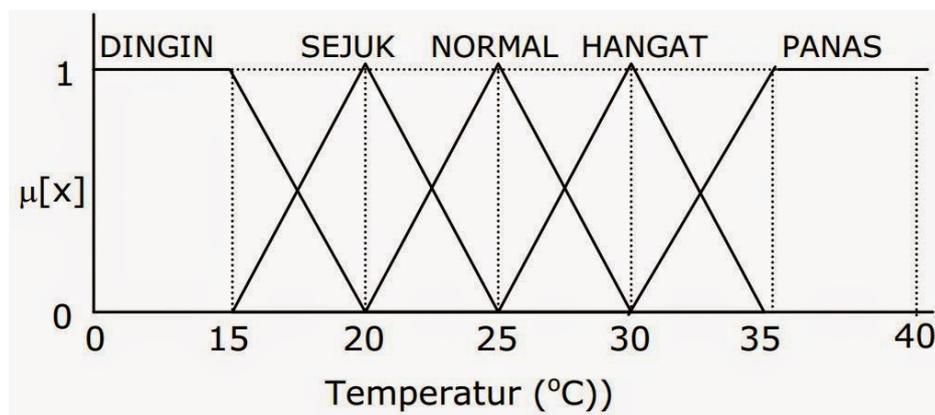
a. Variabel *fuzzy*

Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: umur, temperatur, permintaan, dsb.

b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Contoh:

- a. Variabel umur, terbagi menjadi 3 himpunan *fuzzy*, yaitu: muda, parobaya, dan tua.
- b. Variabel temperatur, terbagi menjadi 5 himpunan *fuzzy*, yaitu: dingin, sejuk, normal, hangat dan panas. Seperti pada gambar 2.1.



Gambar 2.1 Himpunan *Fuzzy* Pada Variabel Temperatur
Sumber: (Kusumadewi & Purnomo, 2013:7)

- c. Semesta Pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif

maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Contoh:

1. Semesta pembicaraan untuk variabel umur: $[0 +\infty]$
2. Semesta pembicaraan untuk variabel temperatur: $[0 40]$

d. *Domain* himpunan *fuzzy* adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, *domain* merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai *domain* dapat berupa bilangan positif dan bilangan negatif. Contoh *domain* himpunan *fuzzy*:

- a. Mudah = $[0 45]$
- b. Parobaya = $[35 55]$
- c. Tua = $[4 +\infty]$
- d. Dingin = $[0 20]$
- e. Sejuk = $[15 25]$
- f. Normal = $[20 30]$
- g. Hangat = $[25 35]$
- h. Panas = $[30 40]$

2.1.5 Fungsi Keanggotaan

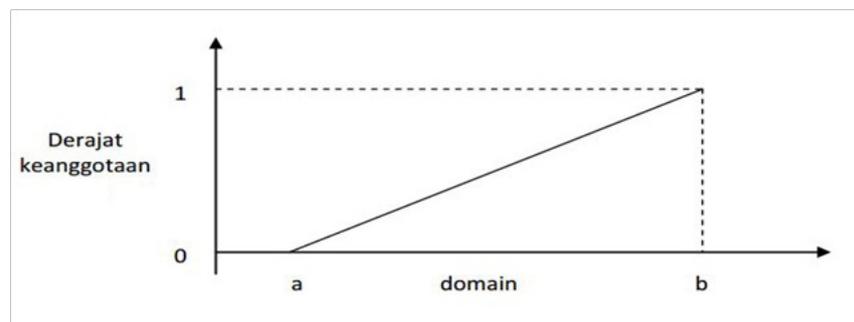
Fungsi keanggotaan adalah grafik yang mewakili besar dari derajat keanggotaan masing-masing variabel *input* berada dalam interval 0 dan 1. Derajat keanggotaan sebuah variabel x dilambangkan dengan simbol $\mu(x)$. *Rule-rule*

menggunakan nilai keanggotaan sebagai faktor bobot untuk menentukan pengaruh pada saat melakukan inferensi untuk menarik kesimpulan. Ada beberapa fungsi keanggotaan yang sering digunakan, di antaranya adalah: (Sutojo et al., 2011:213)

a. Grafik keanggotaan kurva *linear*

Pada grafik keanggotaan *linear*, sebuah variabel *input* dipetakan ke derajat keanggotaannya dengan digambarkan sebagai suatu garis lurus.

Ada 2 grafik keanggotaan *linear*. Pertama grafik keanggotaan linear kurva naik, yaitu kenaikan himpunan *fuzzy* dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi (Gambar 2.2).



Gambar 2.2 Grafik Keanggotaan Kurva *Linear* Naik
Sumber: (Sutojo et al., 2011:214)

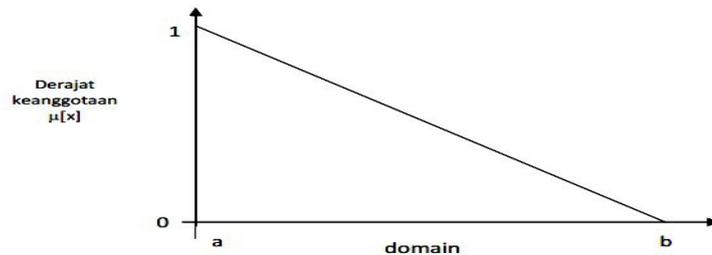
Berikut merupakan rumus fungsi keanggotaan *linear* naik di rumus 2.1

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & x = b \end{cases} \quad \text{Rumus 2.1}$$

Sumber: (Sutojo et al., 2011:214)

Kedua, grafik keanggotaan kurva *linear* turun, yaitu himpunan *fuzzy* dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri,

kemudian bergerak menurun ke nilai *domain* yang memiliki derajat keanggotaan lebih rendah (Gambar 2.2).



Gambar 2.3 Grafik Keanggotaan Kurva *Linear Turun*
Sumber: (Sutojo et al., 2011:215)

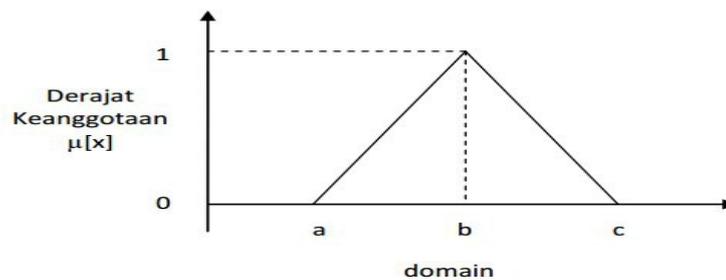
Berikut merupakan rumus fungsi keanggotaan linear turun di rumus 2.2

$$\mu[x] = \begin{cases} (b-x)/(b-a); & a \leq x \leq b \\ 0; & x \geq b \end{cases} \quad \text{Rumus 2.2}$$

Sumber: (Sutojo et al., 2011:216)

b. Grafik keanggotaan kurva segitiga

Grafik keanggotaan kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (*linear*) seperti terlihat pada gambar 2.4



Gambar 2.4 Grafik Keanggotaan Kurva Segitiga
Sumber: (Sutojo et al., 2011:217)

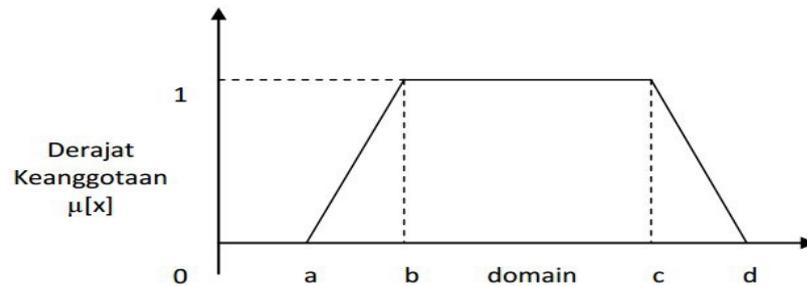
Berikut merupakan rumus fungsi keanggotaan *linear* turun di rumus 2.3

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x-a)/(b-a); & a \leq x \leq b \\ (b-x)/(c-b); & b \leq x \leq c \end{cases} \quad \text{Rumus 2.3}$$

Sumber: (Sutojo et al., 2011:217)

c. Grafik keanggotaan kurva Trapesium

Grafik keanggotaan kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1 (Gambar 2.5).



Gambar 2.5 Grafik Keanggotaan Kurva Trapesium
Sumber: (Sutojo et al., 2011:218)

Berikut merupakan rumus fungsi keanggotaan kurva trapesium di rumus 2.4

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d-x)/(d-c); & c \leq x \leq d \end{cases} \quad \text{Rumus 2.4}$$

Sumber: (Sutojo et al., 2011:218)

2.1.6 Operasi Himpunan *Fuzzy*

Operasi himpunan *fuzzy* diperlukan untuk proses inferensi atau penalaran. Dalam hal ini yang dioperasikan adalah derajat keanggotaannya. Derajat keanggotaan sebagai hasil dari operasi dua buah himpunan *fuzzy* disebut sebagai *fire strength* atau α -predikat. Berikut beberapa operasi dasar yang paling sering digunakan untuk menggabungkan dan memodifikasi himpunan *fuzzy*. (Sutojo et al., 2011:227)

a. Operasi Gabungan (*Union*)

Operasi gabungan (sering disebut operator *OR*) dari himpunan *fuzzy* A dan B dinyatakan sebagai $A \cup B$. dalam system logika *fuzzy*, operasi gabungan disebut sebagai *Max*. Operasi *max* ditulis dengan persamaan berikut.

$$\mu_{A \cup B}(x) = \max \{ \mu_A(x), \mu_B(x) \} \text{ untuk setiap } x \in X \quad \textbf{Rumus 2.5}$$

b. Operasi Irisan (*Intersection*)

Operasi irisan (sering disebut operator *AND*) dari himpunan *fuzzy* A dan B dinyatakan sebagai $A \cap B$. dalam sistem logika *fuzzy*, operasi irisan disebut sebagai *min*. Operasi *min* ditulis dengan persamaan berikut.

$$\mu_{A \cap B}(x) = \min \{ \mu_A(x), \mu_B(x) \} \text{ untuk setiap } x \in X \quad \textbf{Rumus 2.6}$$

c. Operasi Komplemen (*Complement*)

Bila himpunan *fuzzy* A pada himpunan universal X merupakan fungsi keanggotaan $\mu_A(X)$ maka komplemen dari himpunan *fuzzy* A (sering disebut NOT) adalah himpunan *fuzzy* A^c dengan fungsi keanggotaan untuk setiap x elemen X.

$$\mu_{A^c}(x) = 1 - \mu_A(x) \quad \textbf{Rumus 2.7}$$

2.1.7 Penalaran Monoton

Metode penalaran secara monoton digunakan sebagai dasar untuk teknik implikasi *fuzzy*. Meskipun penalaran ini sudah jarang sekali digunakan, namun terkadang masih digunakan untuk penskalaan *fuzzy*. Jika 2 daerah *fuzzy*

direlasikan dengan implikasi sederhana sebagai berikut: (Kusumadewi & Purnomo, 2013:25)

IF x is A THEN y is B

transfer fungsi:

$$y = f((x,A),B)$$

Maka sistem *fuzzy* dapat berjalan tanpa harus melalui komposisi dan dekomposisi *fuzzy*. Nilai *output* dapat diestimasi secara langsung dari nilai keanggotaan yang berhubungan dengan antesedennya.

2.1.8 Fungsi Implikasi

Tiap-tiap aturan (proposisi) pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Bentuk umum dari aturan yang digunakan dalam fungsi implikasi adalah: (Kusumadewi & Purnomo, 2013:28-29)

IF x is A THEN y is B

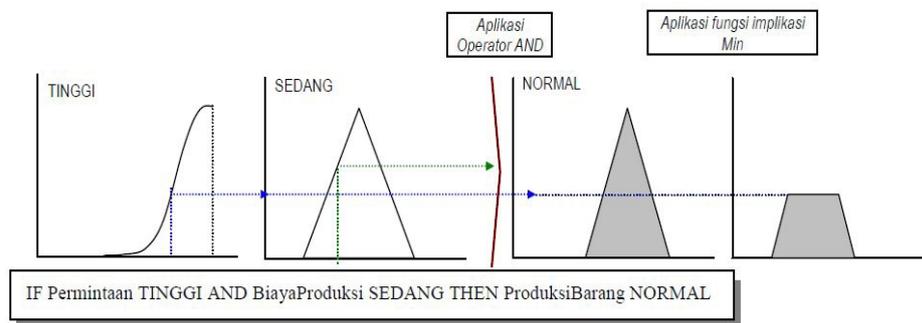
dengan x dan y adalah skalar, dan A dan B adalah himpunan *fuzzy*. Proposisi yang mengikuti *IF* disebut sebagai anteseden, sedangkan proposisi yang mengikuti *THEN* disebut sebagai konsekuen. Proposisi ini dapat diperluas dengan menggunakan operator *fuzzy*, seperti:

IF (x1 is A1) • (x2 is A2) • (x3 is A3) • • (xN is AN) THEN y is B

dengan \bullet adalah operator (misal: *OR* atau *AND*). Secara umum, ada 2 fungsi implikasi yang dapat digunakan, yaitu:

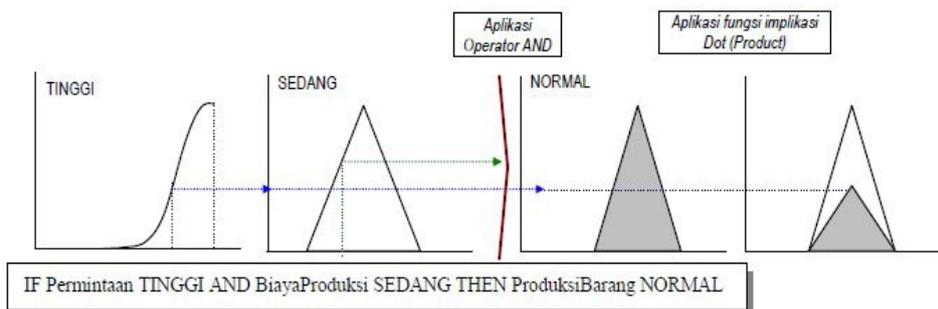
a. *Min (minimum)*. Fungsi ini akan memotong *output* himpunan *fuzzy*. Gambar

2.6 menunjukkan salah satu contoh penggunaan fungsi *min*.



Gambar 2.6 Fungsi Implikasi *MIN*
 Sumber: (Kusumadewi & Purnomo, 2013:29)

- b. *Dot (product)*. Fungsi ini akan menskala *output* himpunan *fuzzy*. Gambar 2.7 menunjukkan salah satu contoh penggunaan fungsi *dot*.



Gambar 2.7 Fungsi Implikasi *DOT*
 Sumber: (Kusumadewi & Purnomo, 2013:29)

2.1.9 Metode Sugeno

Metode Sugeno hampir sama dengan penalaran Mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan *linear*. Metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985, sehingga metode ini sering juga dinamakan dengan metode TSK. Metode TSK terdiri dari 2 jenis yaitu: (Kusumadewi & Purnomo, 2013:46)

a. Model *Fuzzy* Sugeno Orde-Nol

Bentuk Secara Umum:

$$\text{IF } (X_1 \text{ is } A_1) \cdot (X_2 \text{ is } A_2) \cdot (X_3 \text{ is } A_3) \cdot \dots \cdot (X_n \text{ is } A_n) \text{ THEN } z = k$$

Dengan A_1 adalah himpunan *fuzzy* ke - I sebagai anteseden, dan k adalah konstanta (tegas) sebagai konsekuen.

b. Model *Fuzzy* Sugeno Orde - Satu

Bentuk Secara Umum:

$$\text{IF } (X_1 \text{ is } A_1) \cdot \dots \cdot (X_n \text{ is } A_n) \text{ THEN } z = P_i \cdot x_i + \dots + P_n \cdot X_n + q$$

Dengan A_1 adalah himpunan *fuzzy* ke - i sebagai anteseden, dan P_i adalah suatu konstanta ke - i dan q merupakan konstanta dalam konsekuen.

2.2 Variabel

Variabel dalam suatu penelitian merupakan objek yang difokus dalam suatu penelitian.

Variabel merupakan pengelompokan secara logis dari dua atau lebih atribut dari objek yang diteliti. (Julyansyah, 2012:47)

2.2.1 Reparasi Kapal

Reparasi merupakan suatu kegiatan memperbaiki kapal yang harus dilakukan oleh pemilik kapal (*owner*) untuk menjaga operasional kapal tetap optimal serta kondisi konstruksi maupun peralatan yang terdapat di dalam kapal tetap baik dan sesuai dengan persyaratan yang telah ditetapkan oleh *class* atau Brio Klasifikasi yang digunakan oleh kapal tersebut.

Reparasi kapal adalah “proses memperbaiki atau mengganti bagian-bagian kapal yang sudah tidak layak dan tidak memenuhi standar minimal kelayakan untuk berlayar baik dari peraturan *statutory* maupun *class*”. (Nurwanti, Wuruk, & Sc, 2016:G42)

2.2.2 Berat Kapal (*Lightweight*)

Berat Kapal (*lightweight*) merupakan berat kosong atau bobot mati suatu kapal yang terdiri dari berat baja kapal, berat *outfit*, berat mesin, dan akomodasi, biasanya berat kapal dinyatakan dalam satuan ton (*tonnage*).

Berat kapal (*lightweight*) yang sering disingkat menjadi LWT adalah Berat kapal kosong dan terdiri dari berat baja kapal, berat konstruksi lambung kapal, berat permesinan, dan peralatan yang digunakan (meliputi berat solar panel, motor listrik, dan baterai). (Bashori & Anita, 2013:4)

Jangkauan semesta pembicaraan tentang berat kapal (LWT) di PT Amnor *Shipyards* adalah diantara 0 sampai dengan 2.850 ton. Himpunan *fuzzy* berat kapal dalam penelitian ini di kelompokkan menjadi 3 himpunan sebagai berikut.

a. Sangat berat

Jika berat kapal diantara 1.900 ton sampai dengan 2.850 ton.

b. Berat

Jika berat kapal diantara 950 ton sampai dengan 1.900 ton.

c. Ringan

Jika berat kapal diantara 0 ton sampai dengan 950 ton.

2.2.3 Sarat Air Kapal (*Draft*)

Sarat air kapal (*draft*) adalah tinggi lambung kapal yang terendam di dalam air yang diukur dari garis muat sampai bagian terendah kapal.

Sarat air kapal (*draft*) adalah “jarak vertikal antara garis air (*load water line*) atas pada garis muat dengan garis dasar (*base line*)” bagian terendah kapal. (Nahdyah, Farhum, & Jaya, 2014:83)

Jangkauan semesta pembicaraan tentang sarat air kapal (*draft*) di PT Amnor *Shipyards* adalah diantara 0 sampai dengan 4,2 meter. Himpunan *fuzzy* sarat air kapal dalam penelitian ini di kelompokkan menjadi 3 himpunan sebagai berikut.

a. Tinggi

Jika sarat air kapal diantara 2,8 meter sampai dengan 4,2 meter.

b. Sedang

Jika sarat air kapal diantara 1,4 meter sampai dengan 2,8 meter.

c. Rendah

Jika sarat air kapal diantara 0 meter sampai dengan 1,4 meter.

2.2.4 Panjang Kapal (*Length Overall*)

Panjang kapal adalah panjang keseluruhan kapal yang dihitung dan diukur dari paling depan kapal hingga paling belakang kapal.

Panjang kapal (*length overall*) adalah panjang keseluruhan kapal yang diukur dari ujung buritan hingga ujung haluan. (Hamzah & Manfaat, 2016:G-287)

Jangkauan semesta pembicaraan tentang panjang kapal (LOA) di PT Amnor *Shipyards* adalah diantara 0 sampai dengan 150 meter. Himpunan *fuzzy*

panjang kapal dalam penelitian ini di kelompokkan menjadi 3 himpunan sebagai berikut.

a. Sangat Panjang

Jika panjang kapal diantara 100 meter sampai dengan 150 meter.

b. Panjang

Jika panjang kapal diantara 50 meter sampai dengan 100 meter.

c. Pendek

Jika panjang kapal diantara 0 meter sampai dengan 50 meter.

2.3 Matrix Laboratory (MATLAB)

2.3.1 Definisi MATLAB

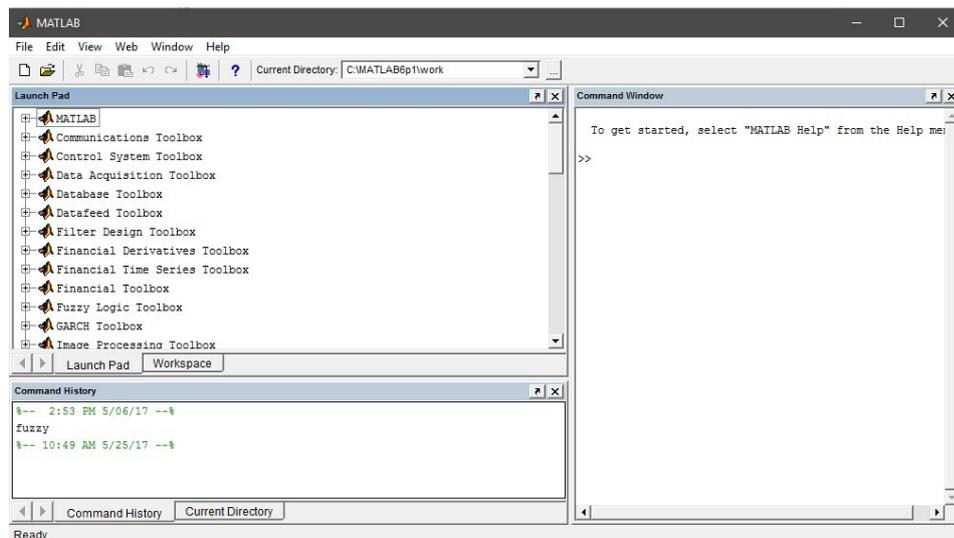
Fakta MATLAB adalah bahasa pemrograman tingkat tinggi dimana arti perintah dan fungsi-fungsinya bisa dimengerti dengan mudah, meskipun bagi seorang pemula. Hal ini dikarenakan di dalam MATLAB, masalah dan solusi bisa diekpresikan dalam notasi-notasi matematis yang biasa dipakai. MATLAB singkatan dari *Matrix Laboratory*. (Naba, 2009:39)

2.3.2 Memulai dan Mengakhiri MATLAB

Ada beberapa tahap dalam memulai dan mengakhiri MATLAB adalah sebagai berikut: (Naba, 2009:40-43)

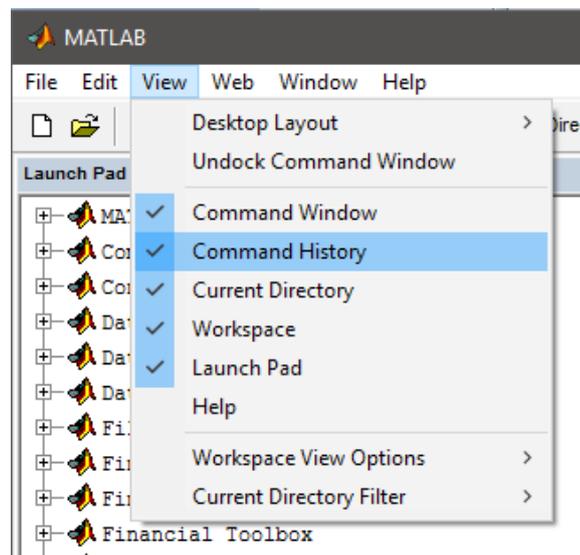
1. Pada sistem operasi *windows*, mulailah MATLAB dengan mengklik dua kali *shortcut* ikon MATLAB pada *Window Desktop* atau klik *menu*

MATLAB dari *Start Menu*. Pada sistem operasi *Linux* atau *UNIX*, mulai MATLAB dengan mengetikkan MATLAB pada *prompt* sistem operasi. MATLAB *desktop* seperti diperlihatkan dalam Gambar 2.8 akan muncul ketika menjalankan MATLAB.



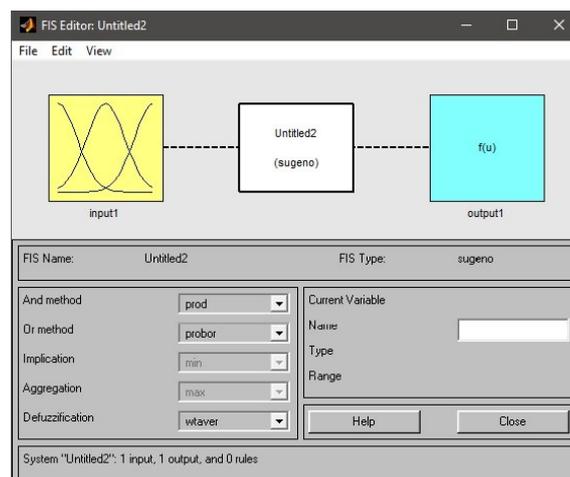
Gambar 2.8 MATLAB Desktop
Sumber: MATLAB 6.1

2. MATLAB menyediakan beberapa *windows*, antara lain *Command Window*, *Current Directory Window* dan *Command History Window*. Untuk menyembunyikan atau memunculkan masing-masing *window*, klik menu *View* lalu klik jenis *window* yang diinginkan. Untuk memunculkan suatu *window*, pastikan muncul tanda *checkbox* disebelah kiri menu jenis *window* yang diinginkan, dan sebaliknya untuk menyembunyikan seperti Gambar 2.9.



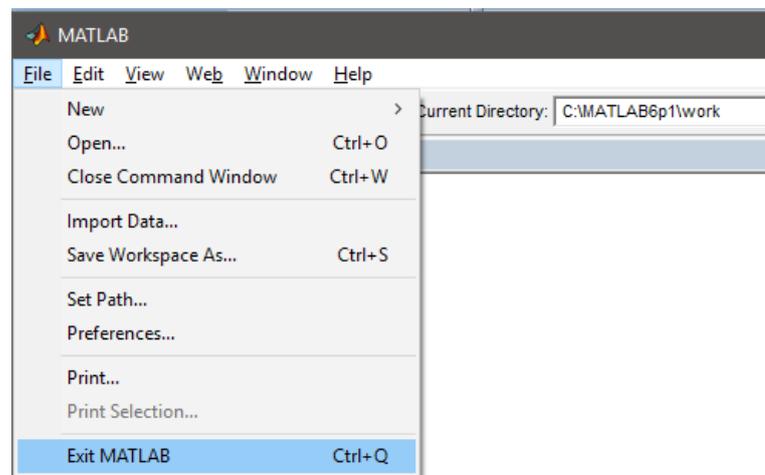
Gambar 2.9 Menu untuk Memilih *Windows* dalam MATLAB
Sumber: MATLAB 6.1

- Untuk memulai *fuzzy logic* dalam program MATLAB, pada MATLAB *Prompt*, ketik *fuzzy* dan klik *enter*. Maka akan muncul *FIS Editor* dengan sebuah variabel masukan dengan label *input1* dan sebuah *output* dengan label *output1*.



Gambar 2.10 Menu untuk Memulai *Fuzzy Logic* dalam MATLAB
Sumber: MATLAB 6.1

- Dan untuk mengakhiri MATLAB, pilih *menu File* lalu pilih pilihan *Exit* MATLAB, dapat dilihat seperti yang ditampilkan Gambar 2.11.



Gambar 2.11 Menu untuk mengakhiri MATLAB
Sumber: MATLAB 6.1

2.4 Penelitian Terdahulu

- a. Menurut penelitian (Pinha & Ahluwalia, 2014:52), *DECISION SUPPORT SYSTEM FOR PRODUCTION PLANNING IN THE SHIP REPAIR INDUSTRY*, All ships and offshore platforms, however large or small, undergo scheduled or unscheduled repair and maintenance. The bidding process for a ship repair job is highly competitive and global in scope. The ship repair industry is also prone to significant risks due to high levels of capital investment in skilled labor, specialized equipment, and facilities such as dry docks. Tradition decision support tools have been utilized by this industry for mid to long-term planning. These tools organize the system as a collection of cost centers and attempt to minimize cost at each center. This paper proposes a decision support system for short term planning. It is oriented towards day to day decision making, with an objective of maximizing system through put and minimizing total project cost. Such an

approach a voids unnecessary internal completion between cost centers, resulting in fewer delays and resource overloading. The proposed decision support system utilizes a common corporate database to share information between stake holders.

- b. Menurut penelitian (**Apriliani, Wisudo, Iskandar, & Novita, 2014:88**), **JARINGAN KERJA DAN EFEKTIVITAS PERBAIKAN KAPAL DI GALANGAN KPNDP DKI JAKARTA, MUARA ANGKE** menyimpulkan bahwa: perbaikan kapal ringan maupun perbaikan kapal berat memiliki persamaan jenis pekerjaan, yaitu: sekrap, pemakalan dan pendempulan serta pengecatan. Perbedaan kedua jenis kegiatan ini terlihat dari kegiatan yang membutuhkan proses dan lama waktu yaitu perbaikan mesin kapal yang terdapat pada perbaikan kapal berat. Jalur kritis pada perbaikan kapal ringan pada kegiatan B-C-D-H-I-J-K-L-P-Q sedangkan jalur kritis untuk perbaikan kapal berat adalah kegiatan B-C-D-G-R-T-U-V. Perbaikan kapal ringan seharusnya dapat diselesaikan dalam waktu kerja efektif 5 hari, waktu ini memiliki efektivitas sebesar 71,43% dibandingkan dengan waktu yang dialokasikan pihak galangan yaitu 7 hari. Perbaikan kapal berat dapat diselesaikan berdasarkan waktu kerja efektif selama 10 hari.
- c. Menurut Penelitian (**Putri & Effendi, 2016:58**), **FUZZY LOGIC UNTUK MENENTUKAN LOKASI KIOS TERBAIK DI KEPRI MALL DENGAN MENGGUNAKAN METODE SUGENO** menyimpulkan bahwa Berdasarkan perhitungan manual dengan hasil 0,85 dan pengujian sistem dengan hasil 0,86 maka dapat disimpulkan bahwa hasil perhitungan manual dengan metode Sugeno dinyatakan *valid* dimana hasil perhitungan

manual dengan metode Sugeno dan MATLAB menunjukkan lokasi kios terbaik memiliki nilai yang tepat

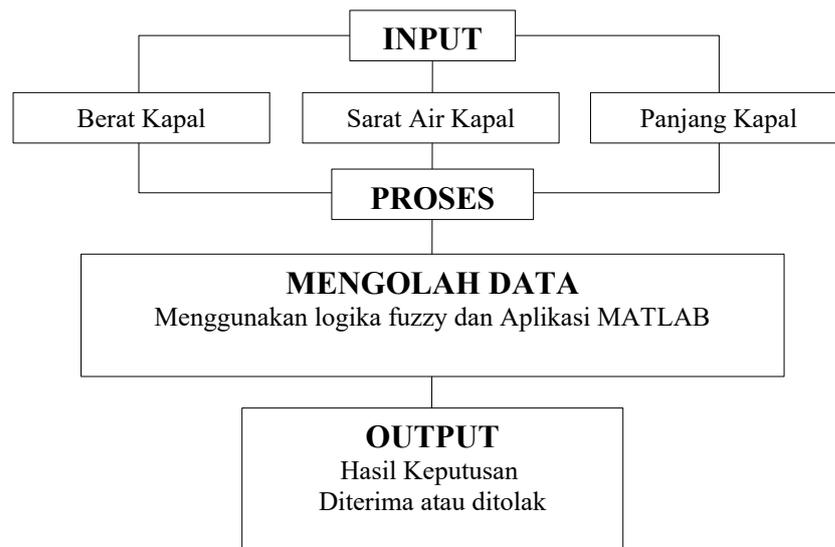
- d. Menurut penelitian (**Agustin, Gandhiadi, & Oka, 2016:182**), PENERAPAN METODE *FUZZY* SUGENO UNTUK MENENTUKAN HARGA JUAL SEPEDA MOTOR BEKAS menyimpulkan bahwa Berdasarkan pembahasan mengenai penerapan metode *fuzzy* Sugeno untuk menentukan harga jual sepeda motor bekas, maka dapat disimpulkan bahwa metode *fuzzy* Sugeno dapat diterapkan dalam menentukan harga jual sepeda motor bekas dengan variabel-variabel *input*, yaitu: tahun motor, kondisi fisik motor, plat nomor, dan harga beli motor. Perhitungan hasil dilakukan dengan menggunakan program MATLAB R2009a, kemudian dilakukan pengujian untuk mengetahui keakuratan dari hasil yang diperoleh dengan menggunakan metode *Mean Absolute Percentage Error* (MAPE). Nilai MAPE yang diperoleh sebesar 5,64%. Hal ini menunjukkan bahwa tingkat kesalahan dibawah 10%, sehingga dapat dikatakan hasil perhitungan tersebut sangat bagus. Dengan demikian, hasil dari perhitungan ini dapat digunakan oleh pemilik *dealer* sebagai alat bantu dalam menentukan harga jual sepeda motor bekas.
- e. Menurut penelitian (**Zulkarami, Sulistiowati, & Lemantara, 2016:6**), RANCANG BANGUN PENJADWALAN DAN MONITORING PERBAIKAN LAMBUNG KAPAL PADA PT TAMBANGAN RAYA PERMAI menyimpulkan bahwa:

- a. Aplikasi dapat meningkatkan kinerja *administrator*, kontraktor BKI dan kordinator *replating* dalam proses pengerjaan perbaikan lambung kapal. Hal ini terlihat pada pengawasan kerja dan proses pengerjaan yang terjadwal dengan baik setelah menggunakan aplikasi dari penelitian.
 - b. Aplikasi sudah berjalan sesuai dengan proses sistem kegiatan di perusahaan. Dibuktikan dengan hasil uji coba aplikasi pada pengguna dengan hasil presentase akhir yaitu 86,42% dengan jumlah responden sebanyak tujuh (7) orang.
- f. Menurut penelitian **(Shokouhifar & Jalali, 2017:17)**, *OPTIMIZED SUGENO FUZZY CLUSTERING ALGORITHM FOR WIRELESS SENSOR NETWORK, propose a new centralized cluster-based routing protocol based on Sugeno fuzzy inference system (named LEACH-SF). In the cluster-based protocols in the literature, CHs are typically selected among all sensor nodes, and then, clusters are formed by simply assigning each node to the nearest CH. The main drawback is to generate an inappropriate distribution of CHs over the network. However, in LEACH-SF, an overall clustering is firstly performed on all nodes to form balanced clusters, and then, proper CHs are selected.*

2.5 Kerangka Pemikiran

Kerangka pemikiran adalah suatu diagram yang menjelaskan secara garis besar alur logika pemikiran dibuat berdasarkan pernyataan penelitian (*research question*).

Berdasarkan teori-teori yang telah diperoleh dan dijelaskan, maka kerangka berpikir dari penelitian, digambarkan pada kerangka pemikiran yang disajikan pada gambar 2.12:



Gambar 2.12 Kerangka Pemikiran

Sumber: Pengolahan Data Penelitian (2018)

Berdasarkan gambar 2.12 kerangka pemikiran terdapat *input* yang terdiri dari berat kapal, sarat air kapal, dan panjang kapal kemudian diproses dengan menggunakan logika *fuzzy* dan aplikasi MATLAB menghasilkan *output* yaitu keputusan diterima atau ditolak.