

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

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 disini adalah kemampuan mesin (komputer) untuk berpikir, menimbang tindakan yang akan diambil, dan melakukan pengambilan keputusan seperti halnya manusia. Alan Turing merupakan seorang ahli matematika berkebangsaan Inggris yang juga dijuluki bapak komputer modern dan beliau merupakan orang yang berhasil membongkar sandi Nazi dalam era Perang Dunia II 1950, mengungkapkan definisi *Artificial Intelligence* “Jika komputer tidak dapat dibedakan dengan manusia saat berbincang melalui terminal komputer, maka bisa dikatakan komputer itu cerdas, mempunyai kecerdasan” (Sutojo, Mulyanto, & Suhartono, 2011). John McCarthy dari Stanford mendefinisikan kecerdasan sebagai “kemampuan untuk mencapai sukses dalam menyelesaikan suatu permasalahan” (Sutojo et al., 2011).

Selain beberapa pengertian kecerdasan buatan di atas, berikut merupakan pengertian kecerdasan buatan dari beberapa ahli dan juga literatur:

1. Herbert Alexander Simon (Juni 15, 1916 – Februari 9, 2001).

Kecerdasan buatan (*Artificial Intelligence*) merupakan kawasan penelitian, aplikasi, dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang dalam pandangan manusia adalah cerdas.

2. Rich dan Knight (1991).

Kecerdasan buatan (AI) merupakan sebuah pembelajaran bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia.

3. Encyclopedia Britannica.

Kecerdasan buatan (AI) merupakan cabang ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan dan memproses informasi berdasarkan metode *heuristics* atau dengan berdasarkan sejumlah aturan.

4. Menurut Winston dan Prendergast, terdapat tiga tujuan dari kecerdasan buatan adalah:

- a) Membuat mesin menjadi lebih pintar (tujuan utama)
- b) Memahami apa itu kecerdasan (tujuan ilmiah)
- c) Membuat mesin lebih bermanfaat (tujuan *entrepreneurial*)

Berdasarkan pengertian di atas, maka kecerdasan buatan menawarkan media maupun uji teori tentang kecerdasan. Teori-teori ini nantinya dapat dinyatakan dalam bahasa pemrograman dan eksekusinya dapat dibuktikan pada komputer nyata. Dari sini dapat dikatakan bahwa: cerdas adalah memiliki pengetahuan, pengalaman, dan penalaran untuk membuat keputusan dan mengambil tindakan.

Jadi, agar mesin menjadi cerdas (bertindak seperti manusia) maka harus diberi bekal pengetahuan dan diberi kemampuan untuk menalar (Sutojo et al., 2011)

Jika dibandingkan dengan kecerdasan alami (kecerdasan yang dimiliki oleh manusia), kecerdasan buatan memiliki keuntungan komersial, antara lain Turban (1992) dalam (Sutojo et al., 2011).

1. Kecerdasan buatan lebih bersifat permanen.
2. Kecerdasan buatan lebih mudah diduplikasi dan disebar.
3. Kecerdasan buatan lebih murah dibandingkan kecerdasan alami.
4. Kecerdasan buatan lebih bersifat konsisten.
5. Kecerdasan buatan dapat didokumentasi.
6. Kecerdasan buatan dapat mengerjakan pekerjaan lebih cepat dibanding dengan kecerdasan alami.
7. Kecerdasan buatan dapat mengerjakan pekerjaan lebih baik dibandingkan dengan kecerdasan alami.

Persoalan-persoalan yang ditangani oleh kecerdasan buatan makin lama makin berkembang sehingga memungkinkan bagi kecerdasan buatan untuk merambah ke bidang ilmu yang lain. Hal ini disebabkan karakteristik cerdas sudah mulai dibutuhkan di berbagai disiplin ilmu dan teknologi (Sutojo et al., 2011).

Berikut beberapa lingkup kecerdasan buatan (Sutojo et al., 2011).

2.1.1 Jaringan Saraf Tiruan (JST)

Jaringan Saraf Tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak

manusia. Elemen kunci dan paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang saling berhubungan (*neuron*), bekerja serentak untuk menyelesaikan masalah tertentu. Cara kerja JST seperti cara kerja manusia, yaitu belajar melalui contoh. Sebuah JST dikonfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau klasifikasi data, melalui proses pembelajaran. Belajar dalam sistem biologis melibatkan penyesuaian terhadap koneksi *synaptic* yang ada antara *neuron*. Hal ini berlaku juga untuk JST (Sutojo et al., 2011).

Jaringan syaraf tiruan sederhana pertama kali diperkenalkan oleh McCulloch dan Pitts di tahun 1943. McCulloch dan Pitts menyimpulkan bahwa kombinasi beberapa *neuron* sederhana menjadi sebuah sistem neural akan meningkatkan kemampuan komputasinya. Fungsi aktivasi yang dipakai adalah fungsi *threshold*. Jaringan Syaraf Tiruan (JST) adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi (Siang, 2009), JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi dengan asumsi bahwa (Sutojo et al., 2011).

1. Pemrosesan informasi terjadi pada banyak elemen sederhana (*neuron*).
2. Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung-penghubung.
3. Penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal.

4. Untuk menentukan *output*, setiap *neuron* menggunakan fungsi aktivasi yang dikarenakan pada jumlahan input yang diterima. Besarnya *output* ini selanjutnya *dibandingkan* dengan suatu batas ambang.

2.1.1.1 Kelebihan JST

Jaringan Saraf Tiruan (JST) memiliki beberapa kelebihan, yang antara lain sebagai berikut:

1. *Belajar Adaptive*

Kemampuan untuk mempelajari bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan atau pengamatan awal.

2. *Self-Organisation*

Sebuah JST dapat membuat organisasi sendiri atau representasi dari informasi yang diterimanya selama waktu belajar.

3. *Real Time Operation*

Perhitungan JST dapat dilakukan secara paralel sehingga perangkat keras yang dirancang dan diproduksi secara khusus dapat mengambil keuntungan dari kemampuan ini.

2.1.1.2 Kelemahan JST

Selain memiliki kelebihan JST juga memiliki kelemahan yang antara lain sebagai berikut:

1. Tidak efektif jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.

2. Tidak efisien jika digunakan untuk melakukan operasi algoritma aritmatika, operasi logika, dan simbolis.
3. Untuk beroperasi JST butuh pelatihan sehingga bila jumlah datanya besar, waktu yang digunakan untuk proses pelatihan sangat lama.

2.1.2 Sistem Pakar

Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) yang cukup tua karena sistem ini sudah dikembangkan pada pertengahan 1960. Sistem pakar yang pertama kali muncul adalah *General-purpose problem solver* (GPS) yang dikembangkan oleh Newell dan Simon. Sampai saat ini sudah banyak sistem pakar yang dibuat, seperti MYCIN untuk diagnosis penyakit, DENDRAL untuk mengidentifikasi struktur molekul campuran yang tak dikenal, XCON & XSEL untuk membantu konfigurasi sistem komputer besar, SOPHIE untuk analisis sirkuit elektronik, *Prospector* digunakan di bidang geologi untuk membantu mencari dan menemukan deposit, FOLIO digunakan untuk membantu memberikan keputusan bagi seorang manager dalam *stock* dan *investasi*, DELTA dipakai untuk pemeliharaan *lokomotif listrik diesel*, dan sebagainya (Sutojo et al., 2011).

Istilah sistem pakar berasal dari istilah *knowledge-based expert system*. Istilah ini muncul karena untuk memecahkan suatu masalah, sistem pakar menggunakan pengetahuan dari seorang pakar yang dimasukkan ke dalam komputer. Seseorang yang bukan pakar menggunakan sistem pakar untuk

meningkatkan kemampuan pemecahan masalah, sedangkan seorang pakar menggunakan sistem pakar untuk *knowledge assistant* (Sutojo et al., 2011).

2.1.2.1 Manfaat Sistem Pakar

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat.
2. Membuat seorang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Andal, sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit
8. Meningkatkan kapabilitas sistem komputer.
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti. Berbeda dengan sistem komputer konvensional. Sistem pakar dapat bekerja dengan informasi yang tidak lengkap. Pengguna dapat merespon dengan “tidak tahu” atau “tidak yakin” pada satu atau lebih pertanyaan selama konsultasi dan sistem pakar tetap akan memberikan jawabannya.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.

2.1.2.2 Kekurangan Sistem Pakar

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% benar.

2.1.2.3 Ciri-ciri Sistem Pakar

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami.
4. Bekerja berdasarkan kaedah/*rule* tertentu.
5. Mudah dimodifikasi.
6. Basis pengetahuan dan mekanisme inferensi terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

2.1.3 Logika *Fuzzy*

Dalam kamus *Oxford*, Istilah *Fuzzy* didefinisikan sebagai *blurred* (kabur atau remang-remang), *indistinct* (tidak jelas), *imprecisely defined* (didefinisikan secara tidak presisi), *confused* (membingungkan), *vague* (tidak jelas) (Agus Naba, 2009). Istilah logika *Fuzzy* yang didasarkan pada logika *Boolean* yang umum digunakan dalam komputasi. Secara ringkas, teorema *Fuzzy* memungkinkan komputer

“berpikir” tidak hanya dalam skala hitam-putih (0 dan 1, mati atau hidup) tetapi juga dalam skala abu-abu. Dalam logika *Fuzzy* suatu preposisi dapat direpresentasikan dalam derajat kebenaran (*truthfulness*) atau kesalahan (*falsehood*) tertentu.

Secara umum, *Fuzzy logic* adalah sebuah metodologi “berhitung” dengan variabel kata-kata (*linguistic variable*), sebagai pengganti berhitung dengan bilangan. Kata-kata yang digunakan dalam *Fuzzy logic* memang tidak sepresisi bilangan, namun kata-kata jauh lebih dekat dengan intuisi manusia. Manusia bisa langsung “merasakan” nilai dari variabel kata-kata yang sudah dipakainya sehari-hari. Demikianlah *Fuzzy logic* memberi ruang dan bahkan mengeksploitasi toleransi terhadap ketidakpresisian. *Fuzzy logic* membutuhkan “ongkos” yang lebih murah dalam memecahkan berbagai masalah yang bersifat *Fuzzy*. *Fuzzy logic* telah menjadi area riset yang mengagumkan karena kemampuannya dalam menjembatani bahasa mesin yang serba presisi dengan bahasa manusia yang cenderung tidak presisi, yaitu hanya dengan menekankan pada makna atau arti (Agus Naba, 2009).

Ada beberapa alasan mengapa orang menggunakan logika *Fuzzy* (Kusumadewi & Purnomo, 2010) antara lain:

1. Konsep logika *Fuzzy* mudah dimengerti. Karena logika *Fuzzy* menggunakan dasar teori himpunan, maka konsep matematis yang mendasari penalaran *Fuzzy* tersebut cukup mudah untuk dimengerti.
2. Logika *Fuzzy* sangat fleksibel, artinya mampu beradaptasi dengan

perubahan-perubahan, dan ketidakpastian yang menyertai permasalahan.

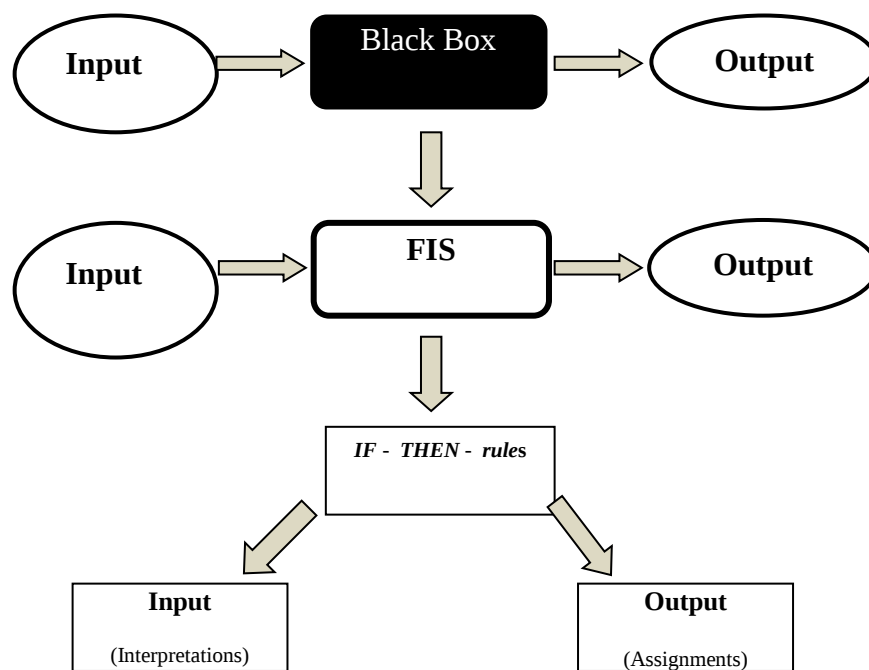
3. Logika *Fuzzy* memiliki toleransi terhadap data yang tidak tepat. Jika diberikan kelompok data yang cukup homogen, dan kemudian ada beberapa data yang “*eksklusif*”, maka logika *Fuzzy* memiliki kemampuan untuk menangani data *eksklusif* tersebut.
4. Logika *Fuzzy* mampu memodelkan fungsi-fungsi *nonlinier* yang sangat kompleks.
5. Logika *Fuzzy* dapat membangun dan mengaplikasikan pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan. Dalam hal ini, sering dikenal dengan nama *Fuzzy Expert System*.
6. Logika *Fuzzy* dapat bekerjasama dengan teknik-teknik kendali secara konvensional. Hal ini umumnya terjadi pada aplikasi dibidang teknik mesin maupun teknik elektro.
7. Logika *Fuzzy* didasarkan pada bahasa alami. Logika *Fuzzy* menggunakan bahasa sehari-hari sehingga mudah dimengerti.

2.1.3.1 Konsep *Fuzzy Logic*

Motivasi utama teori *Fuzzy logic* adalah memetakan sebuah ruang input ke dalam ruang output dengan menggunakan *IF-THEN rules*. Pemetaan dilakukan dalam *Fuzzy Inference System* (FIS). Urutan *rule* bisa sembarang. FIS mengevaluasi semua *rule* secara simultan untuk menghasilkan kesimpulan. Oleh karenanya, semua *rule* harus didefinisikan lebih dahulu sebelum membangun FIS yang akan digunakan untuk menginterpretasikan semua *rule* tersebut. Mekanisme dalam FIS bisa dirangkum seperti ini: FIS adalah sebuah metode yang

menginterpretasikan harga-harga dalam *vektor input*, menarik kesimpulan berdasarkan *IF-THEN rules* yang diberikan, dan kemudian menghasilkan *vektor output* (Agus Naba, 2009).

Kronologi proses rancang bangun FIS diilustrasikan dalam gambar berikut:



Gambar 2.1 Konsep Umum Kronologi Proses Pembangunan FIS

Sumber: Naba (2009)

2.1.3.2 Himpunan *Fuzzy*

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki dua kemungkinan (Kusumadewi & Purnomo, 2010), yaitu:

1. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
2. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Terkadang kemiripan antara keanggotaan *Fuzzy* dengan *probabilitas* menimbulkan kerancuan. Keduanya memiliki nilai pada *interval* $[0,1]$, namun interpretasi nilainya sangat berbeda antara kedua kasus tersebut. Keanggotaan *Fuzzy* memberikan suatu ukuran terhadap pendapat atau keputusan, sedangkan probabilitas mengindikasikan proporsi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Misalnya, jika nilai keanggotaan bernilai suatu himpunan *Fuzzy* USIA adalah 0,9 maka tidak perlu dipermasalahkan berapa seringnya nilai itu diulang secara individual untuk mengharapkan suatu hasil yang hampir pasti muda. Di lain pihak, nilai probabilitas 0,9 usia berarti 10% dari himpunan tersebut diharapkan tidak muda. Himpunan *Fuzzy* memiliki 2 atribut (Kusumadewi & Purnomo, 2010) yaitu:

1. *Linguistik*, penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA
2. *Numeris*, suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 40, 25, 50, dsb.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *Fuzzy* (Kusumadewi & Purnomo, 2010), yaitu:

1. Variable *Fuzzy*

Variable *Fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *Fuzzy*.

Contoh: umur, temperature, permintaan, dsb.

2. Himpunan *Fuzzy*

Himpunan *Fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *Fuzzy*.

3. Semesta Pembicaraan

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. Ada kalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh:

a. Semesta pembicaraan untuk *variable* umur: [0 65]

b. Semesta pembicaraan untuk *variable* temperatur: [0 360]

4. Domain

Domain himpunan *Fuzzy* adalah keseluruhan nilai yang diijinkan 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.

2.1.3.3 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data kedalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki *interval* antara 0 sampai dengan 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi (Kusumadewi & Purnomo, 2010).

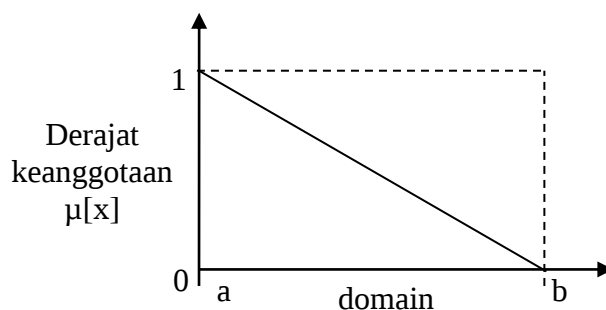
Ada beberapa fungsi yang bisa digunakan (Kusumadewi & Purnomo, 2010) yaitu:

1. Grafik Keanggotaan Kurva Linier

Pada grafik keanggotaan linier, sebuah variabel input dipetakan ke derajat keanggotaannya dengan di gambarkan sebagai suatu garis lurus. Ada 2 grafik keanggotan linier yaitu :

a. Grafik Keanggotaan Linier Turun

Kurva linier 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. (Sutojo et al., 2011)



Gambar 2.2 Grafik Keanggotaan Kurva Linier Turun

Sumber: Sutojo et.al.,(2011)

Fungsi keanggotaan :

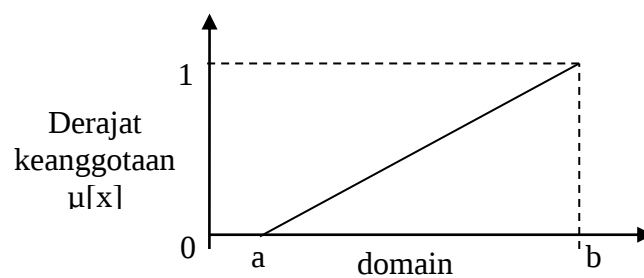
$$\mu [x] = \begin{cases} (b - x) / (b - a) & a \leq x \leq b \\ 0 & \text{lainnya} \end{cases}$$

Rumus 2.1 Kurva *Linier* Turun

Sumber: (Sutojo et al., 2011)

b. Grafik Keanggotaan *Linier* Naik

Kurva *linier* 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 2.3 Grafik Keanggotaan Kurva Linier Naik

Sumber: Sutojo et al., (2011)

Fungsi keanggotaan :

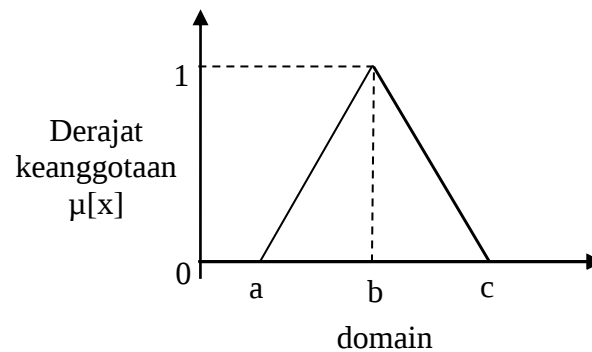
$$\mu[x] = \begin{cases} (x - a) / (b - a); & a \leq x \leq b \\ 1; & x = b \end{cases}$$

Rumus 2.2 Kurva *Linier* Naik

Sumber: (Sutojo et al., 2011)

c. Grafik Keanggotaan Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (*linier*) seperti terlihat gambar dibawah ini. (Sutojo et al., 2011)



Gambar 2.4 Grafik Keanggotaan Kurva Segitiga

Sumber: Sutojo et.al.,(2011)

Fungsi keanggotaan:

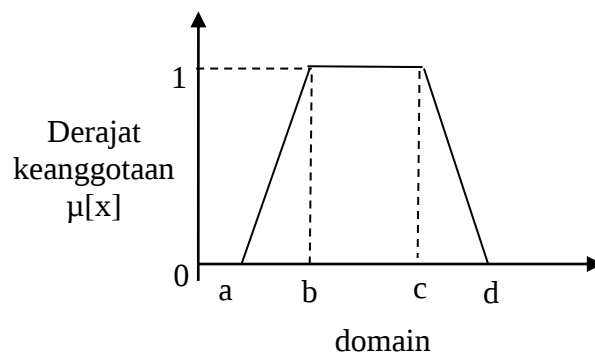
$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x - a) / (b - a); & a \leq x \leq b \\ (c - x) / (c - b) & b \leq x \leq c \end{cases}$$

Rumus 2.3 Kurva Segitiga

Sumber: (Sutojo et al., 2011)

d. Grafik Keanggotaan Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1. (Sutojo et al., 2011).



Gambar 2.5 Grafik Keanggotaan Kurva Trapesium

Sumber: Sutojo et.al., (2011)

Fungsi keanggotaan:

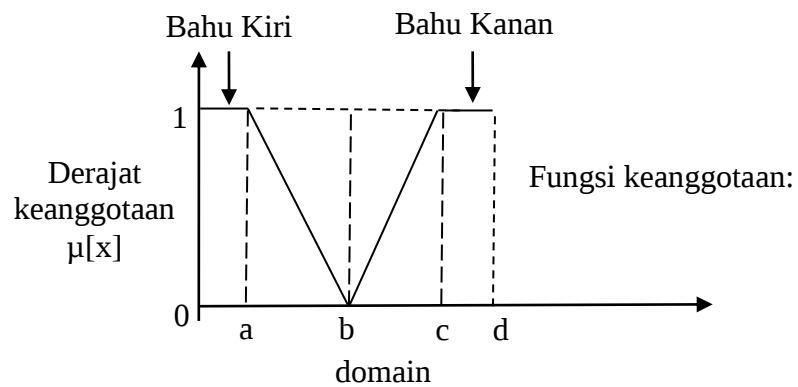
$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x - a) / (b - a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (c - x) / (c - b) & c \leq x \leq d \end{cases}$$

Rumus 2.4 Kurva Trapesium

Sumber: Sutojo et.al., (2011)

e. Grafik Keanggotaan Kurva Bentuk Bahu

Kurva “bahu” digunakan untuk mengakhiri variabel suatu daerah Fuzzy yang nilai derajat keanggotaannya adalah konstan (biasanya 1). (Sutojo et al., 2011)



Gambar 2.6 Grafik Keanggotaan Kurva Bahu

Sumber : Sutojo et.al., (2011)

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

Rumus 2.5 Kurva Bahu

Sumber: (Sutojo et al., 2011)

2.1.3.4 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 strenght atau α -predikat (Sutojo et al., 2011). Berikut beberapa operasi dasar yang paling sering digunakan untuk mengombinasi dan memodifikasi himpunan *Fuzzy* (Sutojo et al., 2011):

1. Operasi Gabungan (*Union*).

Operasi gabungan (sering disebut operator OR) dari himpunan *Fuzzy* A dan B dinyatakan sebagai $A \cup B$. Dalam sistem logika *Fuzzy*, operasi gabungan disebut sebagai *Max*. Derajat keanggotaan setiap unsur himpunan *Fuzzy* $A \cup B$ adalah derajat keanggotaannya pada himpunan *Fuzzy* A atau B yang memiliki nilai terbesar (Sutojo et al., 2011)

2. 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*. Derajat keanggotaan setiap unsur himpunan *Fuzzy* $A \cap B$ adalah derajat keanggotaannya pada himpunan *Fuzzy* A dan B yang memiliki nilai terkecil (Sutojo et al., 2011)

3. Operasi Komplemen (*Complement*).

Bila himpunan *Fuzzy* A pada himpunan universal X mempunyai 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 (Sutojo et al., 2011)

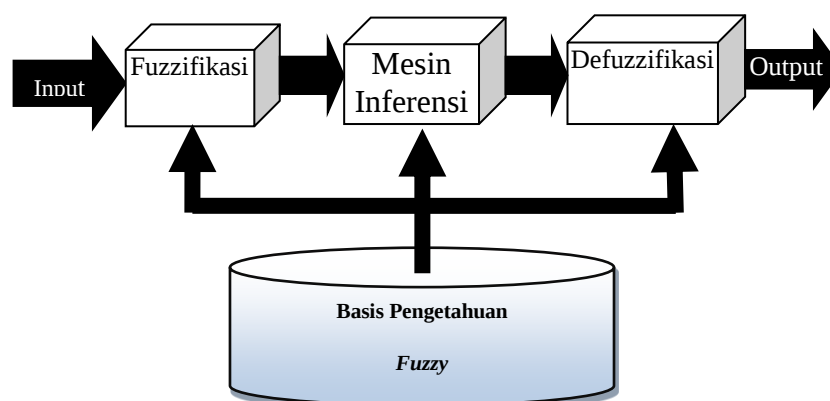
2.1.3.5 *Fuzzy Inference System*

Biasanya seorang operator/pakar memiliki pengetahuan tentang cara kerja dari sistem yang bisa dinyatakan dalam sekumpulan *IF-THEN rule*. Dengan melakukan *Fuzzy inference*, pengetahuan tersebut bisa ditransfer ke dalam perangkat lunak yang selanjutnya memetakan suatu *input* menjadi *output* berdasarkan *IF-THEN rule* yang diberikan. Sistem *Fuzzy* yang dihasilkan disebut *Fuzzy Inference System* (FIS). FIS telah berhasil diaplikasikan dalam berbagai bidang, seperti kontrol otomatis, klasifikasi data, analisis keputusan, dan sistem pakar. Karena kemampuannya yang fleksibel untuk bisa diterapkan di berbagai bidang, FIS sering disebut dengan nama lain, seperti *Fuzzy-rule-based system*, *Fuzzy expert system*, *Fuzzy modelling*, *Fuzzy logic controller*, dan tidak jarang cukup dengan *Fuzzy system* (Agus Naba, 2009).

Fuzzy Inference System (FIS) dapat dilakukan dengan tiga metode, yaitu dengan metode Mamdani, Metode Sugeno dan Metode Tsukamoto (Kusumadewi dan Purnomo, 2010: 31-59). Diantara ketiga metode FIS, metode Mamdani adalah metode yang paling sering dijumpai ketika membahas metodologi-metodologi *Fuzzy*. Hal ini mungkin karena metode ini merupakan metode yang pertama kali dibangun dan berhasil diterapkan dalam rancang bangun *system* kontrol menggunakan teori himpunan *Fuzzy*. Adalah Ebrahim Mamdani yang pertama kali mengusulkan metode ini di tahun 1975 ketika membangun sistem kontrol mesin uap dan boiler. Mamdani menggunakan sekumpulan *IF-THEN rule* yang diperoleh dari operator/pakar yang berpengalaman. Karya Mamdani ini sebenarnya didasarkan pada artikel “*The Father of Fuzzy, Lotfi A. Zadeh: Fuzzy algorithms for complex systems and decisiin processes*” (Agus Naba, 2009).

Keluaran tipe Mamdani berupa *Fuzzy set* dan bukan sekedar inversi dari fungsi keanggotaan output. Dengan kata lain, untuk menghitung harga keluaran dari suatu *IF-THEN rule*, metode Mamdani harus menghitung luas di bawah kurva *Fuzzy set* pada bagian keluaran (*THEN part*). Selanjutnya dalam proses defuzzifikasi, metode Mamdani harus menghitung rata-rata (*centroid*) luas yang diboboti dari semua *Fuzzy set* keluaran dari semua *rule*, kemudian mengisikan rata-rata tersebut ke variabel keluaran FIS. Namun dalam banyak kasus, akan jauh lebih efisien jika menghindari penghitungan luas di bawah kurva *Fuzzy set* keluaran. Sebagai gantinya bisa menggunakan *single spike* sebagai fungsi keanggotaan keluaran. Fungsi keanggotaan keluaran demikian dikenal dengan fungsi keanggotaan *singleton* dan bisa dianggap sebagai sebuah *pre-defuzzified Fuzzy set*. Pendekatan demikian jauh menghemat waktu komputasi daripada metode Mamdani standar yang mengharuskan penentuan centroid sebelum proses defuzzifikasi. Pendekatan ini didukung dalam FIS tipe Sugeno. Secara umum, FIS tipe Sugeno dapat diaplikasikan pada sembarang model *inference system* di mana fungsi keanggotaan keluaran adalah konstan atau linier (Agus Naba, 2009).

Untuk memahami cara kerja logika *Fuzzy*, perhatikan struktur elemen dasar sistem inferensi *Fuzzy* berikut (Sutojo et al., 2011).



Gambar 2.7 Struktur *Fuzzy Inference System*

Sumber: (Sutojo et al., 2011)

Keterangan:

1. Basis Pengetahuan *Fuzzy*: kumpulan *rule-rule Fuzzy* dalam bentuk pernyataan *IF-THEN*.
2. *Fuzzyfikasi*: proses untuk mengubah input sistem yang mempunyai nilai tegas menjadi variabel linguistik menggunakan fungsi keanggotaan yang disimpan dalam Basis pengetahuan *fuzzy*.
3. Mesin Inferensi: Proses untuk mengubah *input Fuzzy* menjadi *output Fuzzy* dengan cara mengikuti aturan-aturan (*IF-THEN rules*) yang telah ditetapkan pada Basis pengetahuan *Fuzzy*.
4. *Defuzzifikasi*: mengubah output *Fuzzy* yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan *fuzzifikasi*.

Cara kerja logika *Fuzzy* meliputi beberapa tahapan berikut (Sutojo, dkk, 2011: 233):

1. *Fuzzyfikasi*
2. Pembentukan Basis pengetahuan *Fuzzy* (*rule* dalam bentuk *IF...THEN*)
3. Mesin inferensi (fungsi implikasi *Max-Min* atau *Dot-Product*)
4. *DeFuzzyfikasi*. Banyak cara untuk melakukan *deffuzzyfikasi*, diantaranya

metode rata-rata (*Average*) dan metode titik tengah (*Centre of Area*).

Fuzzy Logic sendiri memiliki beberapa metode dalam penyelesaiannya, seperti metode Tsukamoto, Sugeno dan Mamdani. (Kusumadewi & Purnomo, 2010)

1. Metode Tsukamoto

Metode Tsukamoto merupakan perluasan dari penalaran monoton. Pada metode Tsukamoto, setiap konsekuen pada aturan yang berbentuk *IF-THEN* harus direpresentasikan dengan suatu himpunan *Fuzzy* dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil inferensi dari tiap-tiap aturan diberikan secara tegas (*crisp*) berdasarkan α -predikat (*fire strength*). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot (Kusumadewi & Purnomo, 2010)

Dalam inferensinya, metode Tsukamoto menggunakan tahapan berikut (Sutojo et al., 2011)

- a. *Fuzzyfikasi*
- b. Pembentukan Basis pengetahuan *Fuzzy* (*Rule* dalam bentuk *IF-THEN*)
- c. Mesin inferensi. Menggunakan fungsi implikasi MIN untuk mendapatkan nilai α -predikat tiap-tiap *rule* ($\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n$).
Kemudian masing-masing nilai α -predikat ini digunakan untuk menghitung keluaran hasil inferensi tegas (*crisp*) masing masing *rule* ($Z_1, Z_2, Z_3, \dots, Z_n$)
- d. *DeFuzzyfikasi*. Menggunakan metode rata-rata (*Average*)

2. Metode Sugeno

Penalaran dengan metode Sugeno hampir sama dengan penalaran Mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *Fuzzy*, melainkan berupa konstanta atau persamaan linier. Metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985, sehingga metode ini sering juga dinamakan dengan Metode TSK (Sutojo et al., 2011).

Dalam inferensinya, metode Sugeno menggunakan tahapan berikut (Sutojo et al., 2011).

- a. *Fuzzyfikasi*
- b. Pembentukan Basis pengetahuan *Fuzzy* (*Rule* dalam bentuk *IF-THEN*)
- c. Mesin inferensi. Menggunakan fungsi implikasi MIN untuk mendapatkan nilai α -predikat tiap-tiap *rule* ($\alpha_1, \alpha_2, \alpha_3, \dots \alpha_n$). Kemudian masing-masing nilai α -predikat ini digunakan untuk menghitung keluaran hasil inferensi tegas (*crisp*) masing masing *rule* ($Z_1, Z_2, Z_3, \dots Z_n$)
- d. *DeFuzzyfikasi*. Menggunakan metode rata-rata (*Average*)

3. Metode Mamdani

Metode Mamdani sering dikenal dengan nama Metode *Max-Min*. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan output, diperlukan 4 tahapan (Kusumadewi & Purnomo, 2010):

- a. Pembentukan Himpunan *Fuzzy*

Pada Metode Mamdani, baik variabel input maupun variabel *output* dibagi menjadi satu atau lebih himpunan *Fuzzy*.

b. Aplikasi Fungsi Implikasi.

Tiap-tiap aturan (proposisi) pada Basis pengetahuan *Fuzzy* akan berhubungan dengan suatu relasi *Fuzzy*. Bentuk umum aturan dari aturan yang digunakan dalam fungsi implikasi adalah:

$$IF\ x\ is\ A\ THEN\ y\ is\ B$$

dengan x dan y adalah skalar, dan A dan B adalah himpunan *Fuzzy*.

Proposisi yang mengikuti *IF* disebut *anteseden*, sedangkan proposisi yang mengikuti *THEN* disebut sebagai konsekuensi. Proposisi ini dapat diperluas dengan menggunakan operator *Fuzzy*, seperti (Kusumadewi & Purnomo, 2010):

$$IF\ (x_1\ is\ A_1)\ o\ (x_2\ is\ A_2)\ o\ (x_3\ is\ A_3)\ o\ \dots\ o\ (x_N\ is\ A_N)\ THEN\ y\ is\ B$$

Dengan o 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*.
- b. Dot (*product*). Fungsi ini akan menskala output himpunan *Fuzzy*.

Pada metode Mamdani, fungsi implikasi yang digunakan adalah **Min**.

c. Komposisi Aturan

Apabila sistem terdiri dari beberapa aturan, maka inferensi diperoleh dari gabungan antar aturan. Ada tiga metode yang digunakan dalam melakukan inferensi sistem *Fuzzy*, yaitu: *max*, *additive* dan *probabilistik OR* (probor).

- a) Metode *Max* (*Maximum*)

Pada metode ini, solusi himpunan *Fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah *Fuzzy*, dan mengaplikasikannya ke output dengan menggunakan operator *OR(union)*. Jika semua proposisi telah dievaluasi, *Then output* akan berisi suatu himpunan *Fuzzy* yang merefleksikan kontribusi dari tiap-tiap proposisi.

b) Metode *Additive (Sum)*

Pada metode ini, solusi himpunan *Fuzzy* diperoleh dengan cara melakukan *bounded-sum* terhadap semua *output* daerah *Fuzzy*.

c) Metode *Probabilistik OR (probor)*

Pada metode ini, solusi himpunan *Fuzzy* diperoleh dengan cara melakukan *product* terhadap semua output daerah *Fuzzy*.

d. Penegasan (*deFuzzy*)

Input dari proses *defuzzifikasi* adalah suatu himpunan *Fuzzy* yang diperoleh dari suatu komposisi aturan – aturan *Fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada himpunan *Fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *Fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai output.

Ada beberapa metode *defuzzifikasi* pada komposisi aturan Mamdani, antara lain:

a. Metode *Centroid (Composite Moment)*

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat (Z^*) daerah *Fuzzy*.

b. Metode Bisektor

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai pada domain *Fuzzy* yang memiliki nilai keanggotaan setengah dari jumlah total nilai keanggotaan daerah *Fuzzy*.

c. Metode *Mean of Maximum* (MOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode *Largest of Maximum* (LOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. Metode *Smallest of Maximum* (SOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

Pada tahap *defuzzifikasi* metode Mamdani menggunakan metode *Centroid*. (Sutojo et al., 2011).

2.2 Variabel

Adapun langkah-langkah yang dilakukan dalam penerapan metode *Fuzzy* Mamdani salah satunya adalah menentukan variable input dan output. Berikut adalah tabel variabel input dan output dalam penentuan optimasi *Inventory* dalam pengadaan barang.

Table 2.1 Variabel dan Himpunan *Fuzzy*

Fungsi	Variabel	Himpunan <i>Fuzzy</i>	Domain
<i>Input</i>	Persediaan	Sedikit	0 – 19
		Sedang	15 – 45
		Banyak	40 – 59
	Pemakaian	Sedikit	1 – 16
		Sedang	10 – 36
		Banyak	34 – 50
<i>Input</i>	Lama Pemesanan	Cepat	3 – 11
		Sedang	8 – 16
		Lama	14 – 21
	Harga	Murah	70000 – 750000
		Sedang	500000 – 1200000
		Mahal	1100000 – 1875000
<i>Output</i>	Pemesanan Suku Cadang	Sedikit	1 – 16
		Sedang	14 – 40
		Banyak	30 – 50

Sumber : Data Penelitian, (2018)

Pada tabel di atas terdapat dua buah fungsi yaitu *input* sebagai masukan dan *output* sebagai keluaran. Fungsi *input* dan *output* mempunyai variabel – variabel tersendiri. Untuk lebih jelasnya akan diberikan penjelasan di bawah ini:

1. Variabel *Input*

a. Persediaan

Persediaan yang dimaksud disini adalah jumlah barang yang tersedia di dalam gudang PT Rodamas Makmur Motor.

b. Pemakaian

Yang dimaksud dengan pemakaian adalah jumlah suku cadang yang digunakan/dijual dalam periode waktu tertentu.

c. Lama Pemesanan

Lama pemesanan adalah waktu yang dibutuhkan dalam proses pengadaan sparepart, mulai dari waktu pesan sampai barang diterima.

d. Harga

Harga adalah jumlah uang yang harus dikeluarkan seorang pelanggan untuk dapat memiliki suku cadang yang dikehendaki.

2. Variabel *Output*

a. Pemesanan Suku Cadang

Jumlah pesanan yang dilakukan untuk memenuhi kebutuhan pelanggan dan juga bengkel.

2.3 Software Pendukung

Pada penelitian ini menggunakan bantuan MATLAB untuk simulasi grafik dan *rule* karena pada MATLAB menyediakan perkakas untuk membuat *Fuzzy Inference System* (FIS) yaitu *Fuzzy Logic Toolbox* yang didalamnya terdapat *Graphical User Interface* (GUI) untuk merancang FIS.

2.3.1 MATLAB

MATLAB merupakan perangkat lunak yang cocok dipakai sebagai alat komputasi yang melibatkan penggunaan matriks dan vektor. Fungsi-fungsi dalam *toolbox* MATLAB dibuat untuk memudahkan perhitungan tersebut. Sehingga Menentukan optimasi *Inventory* dan jumlah pesanan dengan menggunakan metode Mamdani dapat menggunakan *toolbox Fuzzy* yang terdapat di *software* MatLab.

Fuzzy logic toolbox memberikan fasilitas *Graphical User Interface* (GUI) untuk mempermudah dalam membangun suatu model *Fuzzy*.



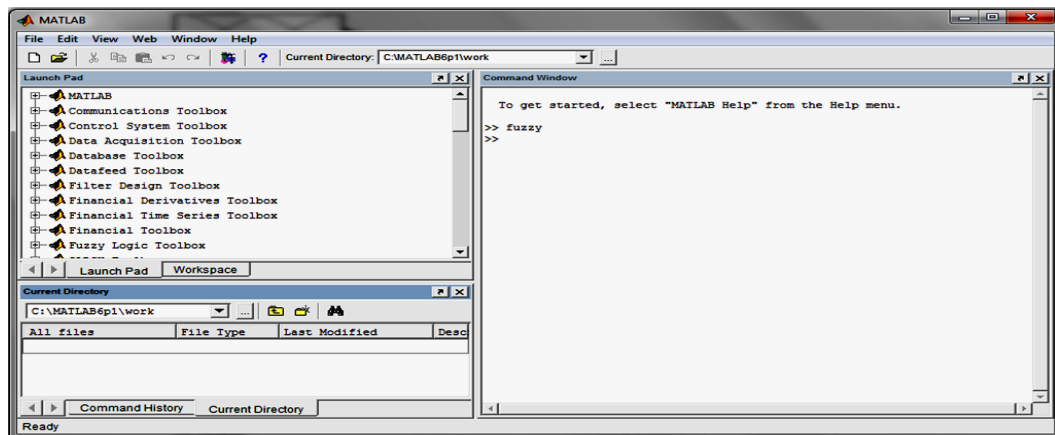
Gambar 2.8 Logo Matlab

Sumber: (Agus Naba, 2009)

MATLAB adalah bahasa pemrograman tingkat tinggi di mana arti perintah dan fungsi-fungsinya bisa dimengerti dengan mudah, meskipun bagi seorang pemula. Hal itu karena di dalam MATLAB, masalah dan solusi bisa diekspresikan dalam notasi-notasi matematis yang biasa dipakai. MATLAB singkatan dari *matrix laboratory*. Dalam dunia akademis, ia telah menjadi alat bantu standar instruksional dalam kuliah-kuliah pengenalan dan tingkat lanjut bidang matematik, teknik dan sains. Spektrum penggunaan MATLAB yang luas ini dimungkinkan karena MATLAB telah melengkapi diri dengan berbagai *toolbox*. Sebuah *toolbox* dalam MATLAB adalah koleksi berbagai fungsi MATLAB (M-Files, yaitu file berekstensi .m), yang merupakan perluasan MATLAB untuk memecahkan masalah-masalah khusus pada bidang tertentu. Oleh karenanya, dengan memakai *toolbox* dalam MATLAB, para pengguna bisa belajar dan menerapkan berbagai *specialized technology*. Beberapa bidang sudah tersedia *toolbox*-nya dalam MATLAB, meliputi *Fuzzy logic*, *neural network* (jaringan syaraf tiruan) *control system* (sistem kontrol), *signal processing* (pengolahan sinyal) dan *wavelet* (Agus Naba, 2009).

2.3.1.1 Memulai dan Mengakhiri MATLAB

Pada sistem operasi windows, mulailah MATLAB dengan mengklik dua kali *shortcut icon* MATLAB pada *Windows Dekstop* atau klik *menu* MATLAB dari *Start Menu*. Pada sistem operasi *Linux* atau *UNIX*, mulai MATLAB dengan mengetikkan MATLAB pada *prompt* sistem operasi. MATLAB desktop akan muncul ketika mulai menjalankan MATLAB. MATLAB menyediakan beberapa *window*, antara lain *Command Window*, *Current Directory Window*, *Workspace Window*, dan *Comman 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 *checkboxlist* di sebelah kiri menu jenis *window* yang diinginkan, dan sebaliknya untuk menyembunyikan (Agus Naba, 2009).



Gambar 2.9 MATLAB Dekstop

Sumber: (Agus Naba, 2009)

Untuk mengakhiri MATLAB, pilih menu *File Exit* MATLAB atau ketikkan *exit* atau *quit* pada MATLAB *prompt*. MATLAB akan selalu mencari dan

mengeksekusi *file finish*. sebelum benar-benar keluar. MATLAB mengijinkan membuat atau memodifikasi sendiri *file finish*. jika menginginkan MATLAB untuk melakukan sesuatu sebelum keluar (Agus Naba, 2009).

2.3.1.2 Dasar-dasar Pemrograman MATLAB

Dasar-dasar pemrograman dalam MATLAB meliputi (Agus Naba, 2009): *Flow Control: IF, switch, case, for, while, continue, break*. *Data Structure*: dipakai untuk mengani multidimensional *arrays, cell arrays, character, text data* dan *structures*. *Scripts*: sekumpulan perintah yang disimpan dalam *M-Files*, tidak memerlukan argumen input dan tidak memberikan suatu keluaran (*notreturning output argument*). *Funtcions*: *M-Files* yang memerlukan argumen input dan mennghasilkan suatu keluaran.

2.3.1.3 Fuzzy Logic Toolbox

Fuzzy Logic Toolbox adalah sekumpulan *tool* yang akan membantu peneliti merancang sistem *Fuzzy* untuk diaplikasikan dalam berbagai bidang, seperti *automatic control, signal processing, identification system, pattern recognition, time series prediction, data mining*, dan bahkan *financial applications*. Dengan *Fuzzy Logic Toolbox*, peneliti bisa membuat atau mengedit FIS dalam lingkungan kerja MATLAB. *Fuzzy Logic Toolbox* sangat *user friendly*, memungkinkan pengguna berkreasi dengan bebas dalam rancang bangun FIS (Agus Naba, 2009). Semua *tool* dalam *Fuzzy Logic Toolbox* dikelompokkan menjadi tiga kategori (Agus Naba, 2009).

1. *Command Lines*

Fungsi-fungsi *command lines Fuzzy Logic Toolbox* adalah fungsi-fungsi yang dapat dieksekusi langsung dari *MATLAB Prompt*. Sebagian besar fungsi ini ditulis dalam bentuk M-Files.

2. *Graphical User Interface (GUI)*

GUI memungkinkan pengguna mengakses banyak fungsi-fungsi yang tersedia dalam *Fuzzy Logic Toolbox*. Sebenarnya *Fuzzy Logic Toolbox* lebih banyak mengandalkan GUI dalam membantu penyelesaian kerja dalam rancang bangun FIS, meskipun bisa dilakukan dari *command lines*. GUI sangat cocok untuk pemula, sementara *command lines* ditujukan untuk pemakai yang sudah berpengalaman.

3. *Simulink Block*

Kategori ketiga adalah *tool* dalam bentuk blok-blok *Simulink*. Sebenarnya *tool* kategori ketiga ini dirancang khusus untuk aplikasi-aplikasi FIS dalam lingkungan *Simulink*.

2.3.1.4 *Graphical User Interface (GUI)*

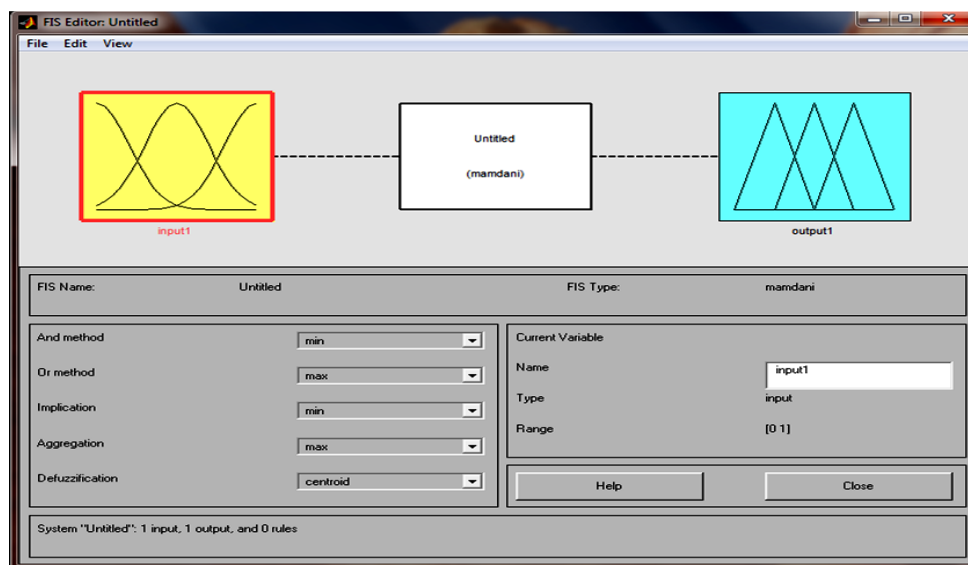
GUI memungkinkan pengguna mengakses banyak fungsi-fungsi yang tersedia dalam *Fuzzy Logic Toolbox*. Sebenarnya *Fuzzy Logic Toolbox* lebih banyak mengandalkan GUI dalam membantu penyelesaian kerja dalam rancang bangun FIS, meskipun bisa dilakukan dari *command lines*. GUI sangat cocok untuk pemula, sementara *command lines* ditujukan untuk pemakai yang sudah berpengalaman (Agus Naba, 2009).

Fuzzy Logic Toolbox menyediakan 5 jenis GUI untuk rancang bangun FIS, yaitu:

1. FIS Editor

Pada MATLAB *Prompt*, ketikkan *Fuzzy*, maka akan muncul FIS *editor*

dengan sebuah variabel masukan dengan label input1 dan sebuah output dengan label output1.



Gambar 2.10 Tampilan FIS Editor

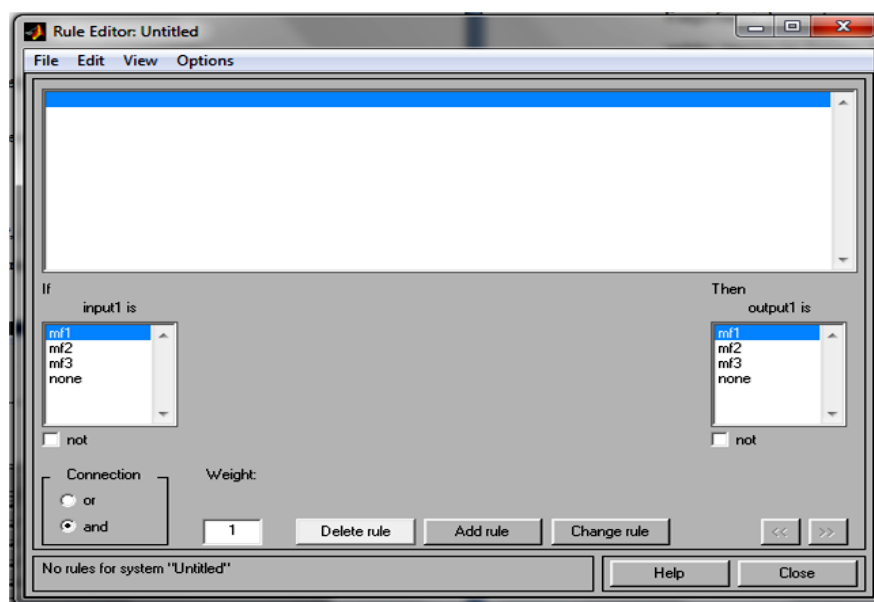
Sumber: (Agus Naba, 2009)

2. Membership Function Editor

Fungsi-fungsi keanggotaan variabel masukan dan keluaran didefinisikan melalui *Membership Function Editor*. Fitur-fitur dalam *Membership Function Editor* serupa dengan fitur-fitur dalam FIS editor, dan juga semua GUI FIS yang belum disebut sejauh ini. Dengan *Membership Function Editor*, maka bisa menampilkan dan mengedit semua fungsi keanggotaan dari variabel FIS masukan dan keluaran.

3. RuleEditor

Dengan GUI *Rule Editor*, maka dapat dengan mudah mendefinisikan *IF-THEN rule*. Berdasar deskripsi variabel-variabel masukan dan keluaran yang didefinisikan dalam *FIS editor*, *Rule Editor* memudahkan pengguna menyusun pernyataan-pernyataan *IF-THEN rule* secara otomatis, dengan mengklik sebuah item opsi nilai *linguistik* untuk tiap variabel FIS. Memilih opsi *none* untuk variabel tertentu berarti mengabaikan variabel tersebut dalam *rule* yang sedang dibuat. Memilih opsi *not* untuk variabel tertentu akan menegaskan sembarang harga variabel tersebut.



Gambar 2.11 Tampilan *Rule Editor*

Sumber: (Agus Naba, 2009)

4. *RuleViewer*

Rule Viewer menampilkan proses keseluruhan yang terjadi dalam FIS. Cara kerja *Rule Viewer* didasarkan pada diagram FIS yang dibahas dalam seksi sebelumnya.

5. *Surface Viewer*

Surface Viewer mempunyai kemampuan khusus yang sangat membantu dalam kasus dengan dua atau lebih masukan FIS dan sebuah keluaran.

2.3.2 Mendeley Desktop

Mendeley adalah sebuah perangkat lunak yang kelahirannya diilhami oleh sebuah upaya untuk mengintegrasikan “*citation & reference manager*” ke dalam sebuah jejaring sosial. Dengan jejaring semacam ini, peneliti di berbagai belahan dunia dapat berkolaborasi dan melakukan sharing data penelitian.

Perangkat lunak Mendeley saat ini tersedia dalam 2 (dua) versi, Desktop Edition dan Institutional Edition. Mendeley Desktop Edition (selanjutnya dalam panduan ini akan disebut Mendeley saja) adalah perangkat lunak “*citation & reference manager*” yang bisa didapatkan secara gratis (tidak berbayar) dan sangat kompatibel dengan program pengolah kata MS Word (2003,2007,2010), Mac Word (2008, 2011), Open Office/Libre Office (3.2), dan Bib Tex.

Beberapa fitur yang menjadi andalan Mendeley antara lain:

1. Dapat berjalan pada MS Windows, Mac, ataupun Linux.
2. Menampilkan metadata dari sebuah file PDF secara otomatis.
3. Backup dan sinkronisasi data dari beberapa komputer dengan akun online.
4. *Smart filtering* dan *tagging*.

5. PDF viewer dengan kemampuan anotasi dan *highlighting*.
6. Impor dokumen dan makalah penelitian dari situs-situs eksternal (misalnya PubMed, Google Scholar, arXiv, dll).
7. Integrasi dengan berbagai perangkat lunak pengolah kata seperti MS Word, Open Office, dan Libre Office.
8. Fitur jejaring sosial.
9. iPhone dan iPad app.
10. *Free web storage* sebesar 2 GB yang dapat dimanfaatkan sebagai *online backup*.

Disamping itu, pengaturan dan manajemen file hasil download sangat mudah dilakukan dengan “*drag and drop*”, sehingga terasa sangat *user friendly*. Namun demikian bukan berarti Mendeley tidak memiliki kelemahan. Mendeley akan mengunggah semua file yang ada di *hard disk* lokal ke web site Mendeley, sehingga apabila file-file ini kita dapatkan dari jurnal berbayar seperti ScienceDirect, ProQuest, ClinicalKey tentunya akan sedikit bermasalah khususnya dengan hak akses terhadap file-file tersebut.

2.4 Penelitian Terdahulu

Pada penelitian terdahulu yang memuat tentang beberapa penelitian yang sudah dilakukan oleh peneliti lain yang relevan, akan dibahas secara lengkap

jurnal yang mendukung penelitian yang akan dilakukan. Berikut ini dijabarkan penelitian terdahulu yang digunakan sebagai dasar dalam penelitian ini, yaitu:

1. Dalam penelitiannya Satria Fitra Widya Utama dan Helmie Ar Wibawa (Jurnal Dinamika Rekayasa Vol. 11 No. 2 Agustus 2015, ISSN 1858-3075) dengan judul: Implementasi Logika *Fuzzy* Mamdani Dalam *Game* Simulasi Memancing. Hasil penelitian yang dilakukan menunjukkan bahwa logika *fuzzy* Mamdani dengan memanfaatkan *agregasi or* dan metode *mean of maximum* dalam *defuzzifikasi* dapat dijadikan sebagai model penentuan jenis ikan dan berhasil diimplementasikan pada game simulasi memancing. Kedalaman umpan dan waktu pemancingan dijadikan sebagai masukan untuk perhitungan *fuzzy* Mamdani. Hasil dari perhitungan *fuzzy* Mamdani *dibandingkan* dengan jenis umpan yang digunakan untuk menentukan jenis ikan yang didapat. Penggunaan metode *mean of maximum* saat *defuzzifikasi* pada logika *fuzzy* Mamdani menyebabkan ikan-ikan tertentu yang memakan umpan yang spesifik tidak mungkin didapatkan, sehingga pemain tidak bisa melengkapi daftar ikan yang diperoleh (pencapaian Pemain). Hasil pembangunan game Mancing Nusantara memiliki nilai akurasi 86.9% dalam menentukan kelompok ikan yang didapat.
2. Penelitian selanjutnya yang dilakukan Wira Buana (Jurnal Edik Informatika V2.il 138-143, ISSN 2407-0491) dengan judul: Penerapan *Fuzzy* Mamdani Untuk Sistem Pendukung Keputusan Pemilihan Telepon Seluler. Dalam penelitiannya menyebutkan *fuzzy logic* merupakan salah satu pendekatan yang menggunakan beberapa tahapan tertentu. Beberapa model *fuzzy logic* banyak

diterapkan dalam menyelesaikan berbagai permasalahan salah satunya adalah *fuzzy Mamdani*. Banyaknya jenis telepon seluler yang tersedia dipasaran membuat para konsumen menjadi kesulitan untuk menentukan pilihannya. Permasalahan yang dipilih adalah membangun sistem pendukung pengambilan keputusan untuk membantu memberikan pilihan ponsel bagi para konsumen berdasarkan kriteria-kriteria yang diinginkan oleh konsumen tersebut. Kriteria yang digunakan dalam membantu menentukan pilihan ponsel yang diinginkan konsumen antara lain berdasarkan pada harga, ukuran layar ponsel, dan kapasitas memori. Dikarenakan kriteria-kriteria tersebut sifatnya relatif maka dibuat *fuzzy Mamdani* yang dapat digunakan model untuk mendapatkan pilihan yang tepat dari suatu yang samar. Hasil penelitian dalam pemilihan telepon seluler, berdasarkan data telepon seluler penulis melakukan beberapa pengujian yaitu pengujian dengan *fuzzy* dan pengujian menggunakan *software Matlab* yang dapat membantu mempercepat proses pengolahan data dan mendapatkan sebuah keputusan dengan cepat.

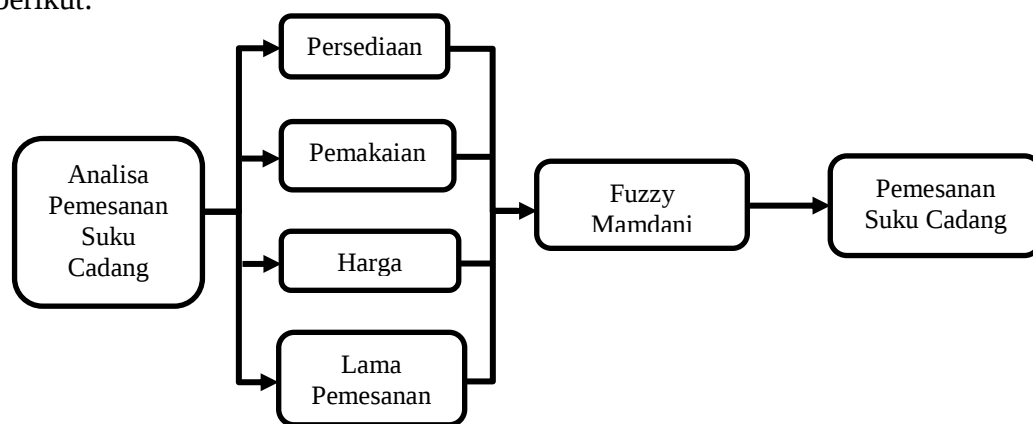
3. Penelitian selanjutnya dilakukan oleh Edy Victor Haryanto dan Vina Nasari (Jurnal STMIK Amikom Yogyakarta Februari 2015, ISSN 2302-3805) dengan judul: Penerapan Metode *Fuzzy Mamdani* Dalam Memprediksi Tingginya Pemakaian Listrik – Studi Kasus Kelurahan ABC. Dalam penelitian ini menggunakan *metode fuzzy Mamdani* untuk mendapatkan hasil factor apa saja yang mempengaruhi tingginya pemakaian listrik dirumah tangga. Penelitian ini menggunakan beberapa variabel yang digunakan, antara lain : luas rumah, tegangan, perlengkapan.

4. Penelitian yang dilakukan Novi Apriyanti dan Huzainsyahnoor Aksad (Jurnal Progresif Vol. 9 No. 1 Februari 2013, ISSN 0216-3284) dengan judul: Penerapan Metode *Fuzzy* Mamdani Dalam Perencanaan Produksi Roti. Penelitian ini menggunakan metode *fuzzy* Mamdani dengan empat variabel yaitu, jumlah permintaan, jumlah sisa, jumlah kekurangan dan jumlah produksi roti. Penelitian bertujuan untuk menentukan produksi roti di Nusqo Bakery.
5. Penelitian selanjutnya dilakukan oleh Minarmi dan Firman Aldyanto (Jurnal Teknoif Vol. 4 No. 2 Oktober 2016, ISSN 2338-2724) dengan judul: Prediksi Jumlah Produksi Roti Menggunakan Metode Logika Fuzzy – Studi Kasus: Roti Malabar Bakery. Penelitian ini membandingkan tiga metode *fuzzy* antara lain: Mamdani, *sugeno* dan *tsukamoto* dalam memprediksi jumlah produksi. Tujuannya adalah mencari metode mana yang paling unggul dalam memprediksi jumlah produksi roti. inputan permintaan dan persediaan. Dari ketiga metode pengujian metode Tsukamoto, Sugeno dan Mamdani, metode Tsukamoto memiliki nilai error yang paling sedikit yang artinya hasil prediksi jumlah produksi dengan metode tsukamoto tidak berbeda jauh dari jumlah produksi yang dilakukan perusahaan. Hal ini ditunjukkan dengan hasil perhitungan nilai error dari jumlah data produksi antara data sebenarnya dengan data prediksi tsukamoto sistem yang menggunakan data dari 01 februari sampai 29 februari 2016 yang menghasilkan nilai error sebanyak 1 %.

2.5 Kerangka Pemikiran

Kerangka berfikir/kerangka konsep merupakan dasar pemikiran pada penelitian yang dirumuskan dari fakta-fakta, observasi dan tinjauan pustaka. Kerangka konsep memuat teori, dalil atau konsep-konsep yang akan dijadikan dasar dan pijakan untuk melakukan penelitian. Uraian dalam kerangka konsep menjelaskan hubungan dan keterkaitan antar variabel penelitian. Variabel penelitian dijelaskan secara mendalam dan relevan dengan permasalahan yang diteliti, sehingga dapat dijadikan dasar untuk menyusun hipotesis dan menjawab permasalahan yang diteliti. (Saryono & Anggraeni, 2013).

Berdasarkan dukungan landasan Teoritis yang diperoleh dari eksplorasi teori yang dijadikan rujukan variabel, maka dapat disusun Kerangka Pemikiran sebagai berikut:



Gambar 2.11 Kerangka Pemikiran

Sumber: Data Penelitian (2018)

Dalam menentukan optimasi pengadaan barang, ditetapkan kriteria-kriteria yang akan digunakan sebagai variabel *input* yang akan dianalisis, kriteria yang ditetapkan yaitu kebutuhan, permintaan, harga, persediaan dan lama pemesanan. Dari data yang diperoleh dilakukan proses analisis dengan menggunakan *Fuzzy*

Inference System Metode Mamdani. Fuzzy Inference System Metode Mamdani digunakan untuk mengolah data sehingga dapat menghasilkan keputusan yang lebih tepat, cepat dan efisien. *Fuzzy Inference System Metode Mamdani* dapat diaplikasikan ke dalam program MATLAB. MATLAB adalah salah satu *software* aplikasi untuk matematika yang sangat *handal* untuk menyelesaikan berbagai masalah matematika, pemakaiannya sangat mudah dengan antarmuka yang sederhana. Hasil analisis adalah rekomendasi yang disusun berdasarkan pokok-pokok kesimpulan dan saran yang didapat dari pembahasan hasil penelitian.