

**IMPLEMENTASI KOMBINASI METODE *BAYESIAN NETWORK* DAN
FUZZY LOGIC UNTUK KOMPUTASI PREDIKSI CUACA DENGAN
BAHASA PEMROGRAMAN R**

SKRIPSI

Diajukan untuk memenuhi sebagian dari
Syarat Memperoleh Gelar Sarjana Komputer
Program Studi Ilmu Komputer



ARZAAQ FALAAHAN ALI

1301682

PROGRAM STUDI ILMU KOMPUTER

DEPARTEMEN PENDIDIKAN ILMU KOMPUTER

FAKULTAS PENDIDIKAN MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS PENDIDIKAN INDONESIA

2017

LEMBAR PENGESAHAN

**IMPLEMENTASI KOMBINASI METODE *BAYESIAN NETWORK* DAN
FUZZY LOGIC UNTUK KOMPUTASI PREDIKSI CUACA DENGAN
BAHASA PEMROGRAMAN R**

Oleh:

Arzaaq Falaahan Ali
1301682

DISETUJUI DAN DISAHKAN OLEH:

Pembimbing 1

Lala Septem Riza, MT. Ph.D.

NIP.197809262008121001

Pembimbing 2

Harsa Wara P., M.Pd

NIP.198008102009121003

Mengetahui,
Kepala Departemen Pendidikan Ilmu Komputer

Prof.Dr.Munir M.IT

NIP.196603252001121001

PERNYATAAN

Saya menyatakan bahwa skripsi yang berjudul “**Implementasi Kombinasi Metode Bayesian Network dan Fuzzy Logic untuk Komputasi Prediksi Cuaca dengan Bahasa Pemrograman R**” ini sepenuhnya karya sendiri. Tidak ada plagiat dari orang lain di dalamnya dan saya tidak melakukan penyalinan atau pengutipan dengan cara-cara yang tidak sesuai etika keilmuan yang berlaku dalam masyarakat keilmuan. Atas pernyataan ini, saya siap menanggung sanksi yang dijatuhkan kepada saya apabila ditemukan adanya pelanggaran terhadap etika keilmuan di karya ini atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Bandung, Mei 2017

Pembuat pernyataan,

Arzaaq Falaahan Ali

1301682

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah-Nya, penelitian yang berjudul “**Implementasi Kombinasi Metode *Bayesian Network* dan *Fuzzy Logic* untuk Komputasi Prediksi Cuaca dengan Bahasa Pemrograman R**” dapat diselesaikan.

Skripsi ini disusun untuk memenuhi salah satu syarat dalam menempuh gelar Sarjana Komputer di Program Studi Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam, Universitas Pendidikan Indonesia.

Penulis sadari bahwa skripsi yang disusun dengan segala usaha dari awal sampai selesai ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan berbagai kritik dan saran dari para pecinta ilmu pengetahuan yang bersifat positif supaya skripsi ini dapat dibangun dengan lebih baik. Penulis juga berharap skripsi ini dapat memberikan manfaat, baik untuk penulis sendiri, umumnya bagi para pengembang teknologi dan pecinta ilmu pengetahuan.

Bandung, Mei 2017

Penulis,

Arzaaq Falaahan Ali

UCAPAN TERIMA KASIH

Dan tentunya dalam penyusunan skripsi dan melakukan penelitian ini, penulis mendapatkan banyak dorongan, bimbingan, dan bantuan dari berbagai pihak. Maka dari itu, penulis mengucapkan terima kasih serta penghargaan setinggi-tingginya kepada :

1. Ayah dan Ibu tercinta yang selalu memberikan dorongan dan motivasi nya, juga yang selalu mendo'akan penulis dalam menyelesaikan skripsi ini.
2. Bapak Lala Septem Riza, MT. Ph.D. selaku pembimbing I, yang selalu memberikan bimbingan, arahan, serta masukkan dengan sabar kepada penulis selama menyusun skripsi ini.
3. Bapak Harsa Wara P., M.Pd. selaku pembimbing II, yang dengan sabarnya membimbing, memberikan arahan serta masukkan dengan sabar kepada penulis selama menyelesaikan skripsi ini.
4. Bapak Muhammad Iid selaku peneliti cuaca dan iklim di BMKG, yang senantiasa meluangkan waktunya untuk membantu penulis dalam melakukan penelitian di skripsi ini.
5. Bapak Herbert Siregar, MT. selaku pembimbing akademik kelas C1, yang selalu memberikan dukungan dan dorongan selama penyusunan skripsi ini.
6. Teman-teman kelas C1 angkatan 2013 yang sama-sama berjuang dan saling memotivasi untuk menyelesaikan skripsi.

Semoga segala bentuk kebaikan yang selama penelitian ini diberikan kepada penulis menjadikan pahala yang besar dan semoga Tuhan meridhoi segala aktivitas yang dilakukan oleh kita. *Aamiin*.

Bandung, Mei 2017

Penulis,

Arzaaq Falaahan Ali

ABSTRAK

Prediksi cuaca merupakan hal yang penting dalam kehidupan manusia, salah satu manfaatnya adalah dapat mengantisipasi orang ketika hendak melakukan aktivitas diluar rumah. Metode Bayesian network menjadi salah satu metode yang tidak asing untuk digunakan dalam mengatasi masalah prediksi. Namun ditemukan masalah ketika data yang digunakan untuk memprediksi adalah bersifat kontinu, karena metode Bayesian network hanya mendukung data-data diskrit saja. Akhirnya metode *fuzzy* digunakan sebagai salah satu solusi untuk mengatasi data-data kontinu tersebut, yakni dengan mendiskritkannya pada tahap fuzzifikasi. Namun ada kerumitan yang muncul ketika menggunakan metode *fuzzy* yaitu harus membangun fungsi keanggotaan, di mana fungsi ini harus dibangun berdasarkan pakar. Model *fuzzy* Wang Mendel memberikan solusi untuk masalah tersebut, yakni dengan membangun fungsi keanggotaan secara otomatis. Dan hasil eksperimen dari model yang terbangun menunjukkan akurasi serta kecepatan komputasi yang baik.

Kata Kunci : Prediksi cuaca, Bayesian network, Metode Fuzzy, Fuzzy Wang Mendel

ABSTRACT

Weather prediction is an important thing in human life, one of the benefits is to warn people when they want to do activities outside home. The Bayesian network is one of the familiar method to solve prediction problems. But the problem is found when the data that used to predict is continuous, because Bayesian network methode only support with discrete data. Finally the fuzzy method is used as one of the solutions to overcome the continuous data, ie by discriminating data at the fuzzification stage. But there are complexities that showed when using the fuzzy method of having to build a membership function, where this function must be built based on expert. Fuzzy model Wang Mendel provides solutions to the problem, ie by automatically building membership functions. And the experimental results of the built model show a good computational accuracy and speed.

Keywords : *Weather prediction, Bayesian network, Fuzzy Method, Fuzzy Wang Mendel*

DAFTAR ISI

PERNYATAAN.....	i
KATA PENGANTAR	ii
UCAPAN TERIMA KASIH.....	iii
ABSTRAK	iv
ABSTRACT.....	v
DAFTAR ISI.....	vi
DAFTAR TABEL.....	viii
DAFTAR GAMBAR	x
BAB I.....	1
PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan.....	4
1.4. Batasan Masalah.....	4
1.5. Sistematika Penulisan.....	4
BAB II.....	6
KAJIAN TEORI	6
2.1. Definisi dan Prediksi Cuaca	6
2.2. Definisi dan Algoritma Naïve Bayes.....	8
2.3. Definisi dan Algoritma Bayesian Network	14
2.4. Teori Fuzzy Set	21
2.5. Definisi dan Algoritma Fuzzy Logic	22
2.5.1. Fuzzification.....	23
2.5.2. Fuzzyfication Wang Mendel	28
2.5.3. Inference System.....	33
2.5.4. Defuzzification.....	34
2.5. Fuzzy Bayesian Network.....	37
2.6. Bahasa R.....	42
BAB III	50
METODOLOGI PENELITIAN.....	50
3.1. Desain Penelitian	50
3.2. Alat dan Bahan Penelitian	53
3.3. Metode Penelitian.....	53

3.3.1.	Proses Pengumpulan Data.....	53
3.3.2.	Proses Pengembangan Perangkat Lunak.....	54
BAB IV	56
PENELITIAN DAN PEMBAHASAN	56
4.1.	Pengumpulan Data	56
4.2.	Perancangan Model Fuzzy dan Bayesian Network	61
4.2.1.	Implementasi Fuzzy Expert.....	62
4.2.2.	Implementasi Fuzzy Wang Mendel	67
4.2.3.	Implementasi Bayesian Network	73
4.2.4.	Implementasi Defuzzifikasi COG	76
5.3.	Model Fuzzy dan Bayesian Network Di R.....	79
5.3.1.	Fungsi Cek Diskrit Kontinu	79
5.3.2.	Fungsi Fuzzifikasi	81
5.3.3.	Fungsi Bayesian Network	83
5.3.4.	Fungsi Defuzzifikasi	86
5.4.	Mengaplikasikan Model Fuzzy dan Bayesian Network.....	87
5.5.	Rancangan Eksperimen	89
5.6.	Analisis dan Hasil Eksperimen.....	91
BAB V	97
KESIMPULAN DAN SARAN	97
5.1.	Kesimpulan.....	97
5.2.	Saran	97
DAFTAR PUSTAKA	98
LAMPIRAN	104

DAFTAR TABEL

Tabel 2.1 Hasil Penelitian Prediksi Cuaca	7
Tabel 2.2 Data Mahasiswa	11
Tabel 2.3 Hasil Probabilitas Sensus Populasi	39
Tabel 2.4 Hasil Probabilitas Biro Pajak	40
Tabel 2.5 Derajat Keanggotaan Sensus Populasi	40
Tabel 2.6 Derajat Keanggotaan Biro Pajak	40
Tabel 4.1 Detil Variabel	57
Tabel 4.2 Data Iklim Cuaca	62
Tabel 4.3 Nilai Derajat Keanggotaan Suhu	63
Tabel 4.4 Nilai Derajat Keanggotaan Kelembaban	64
Tabel 4.5 Nilai Derajat Keanggotaan CURAH HUJAN	64
Tabel 4.6 Nilai Derajat Keanggotaan Kecepatan Angin	65
Tabel 4.7 Data Dummy Cuaca Iklim	66
Tabel 4.8 Nilai Derajat Keanggotaan Wang Mendel SUHU	69
Tabel 4.9 Nilai Derajat Keanggotaan Wang Mendel KECEPATAN ANGIN	70
Tabel 4.10 Nilai Derajat Keanggotaan Wang Mendel CURAH HUJAN	70
Tabel 4.11 Nilai Derajat Keanggotaan Wang Mendel KELEMBABAN	71
Tabel 4.12 Data Dummy Wang Mendel Cuaca Iklim	71
Tabel 4.13 CPT CURAH HUJAN	73
Tabel 4.14 CPT KELEMBABAN	74
Tabel 4.15. CPT KECEPATAN ANGIN	75
Tabel 4.16 CPT SUHU	75
Tabel 4.17 Data Latih dan Data Uji	90
Tabel 4.18 Hasil Pengujian Manual Fuzzy – 1	91
Tabel 4.19 Hasil Pengujian Manual Fuzzy – 2	92
Tabel 4.20 Hasil Pengujian Manual Fuzzy – 3	92
Tabel 4.21 Hasil Pengujian Automatic Fuzzy – 1	93
Tabel 4.22 Hasil Pengujian Automatic Fuzzy – 2	93
Tabel 4.23 Hasil Pengujian Automatic Fuzzy – 3	94

Tabel 4.24 Hasil Pengujian Automatic Fuzzy - 4	94
Tabel 4.25 Hasil Pengujian Model.....	95

DAFTAR GAMBAR

Gambar 2.1 Pseudocode Naïve Bayes	10
Gambar 2.2 Contoh Diagram Acyclic Graph.....	16
Gambar 2.3 Pseudocode Bayesian Network	17
Gambar 2.4 Contoh Conditional Probability Table (CPT)	18
Gambar 2.5 Arsitektur Fuzzy Logic.....	23
Gambar 2.6 Contoh Diagram Fungsi Keanggotaan	24
Gambar 2.7 Pseudocode Fuzzifikasi Segitiga.....	25
Gambar 2.8 Derajat Keanggotaan TEMPERATUR SUHU	26
Gambar 2.9 Hasil Bagi Input dan Output Menjadi Daerah Fuzzy	30
Gambar 2.10 Pseudocode Generate Membership Function Wang Mendel	31
Gambar 2.11 Hasil Proses Generate Membership Function	32
Gambar 2.12 Proses Defuzzifikasi.....	34
Gambar 2.13 Pseudocode Defuzzification COG	35
Gambar 2.14 Daerah Fuzzy Kelembaban	36
Gambar 2.15 Model Proses Fuzzy Bayesian Network.....	37
Gambar 2.16 RStudio.....	45
Gambar 2.17 Operator Penugasan - 1	46
Gambar 2.18 Operator Penugasan - 2	46
Gambar 2.19 Operator Pertambahan.....	47
Gambar 2.20 Operator Pengurangan.....	47
Gambar 2.21 Operator Pembagian.....	47
Gambar 2.22 Operator Perkalian.....	47
Gambar 2.23 Fungsi Gerbang Logika AND	48
Gambar 2.24 Bangun Matriks	48
Gambar 2.25 Gabung Matriks.....	49
Gambar 2.26 Read File in R.....	49
Gambar 2.27 Jalankan File R	49
Gambar 3.1 Desain Penelitian.....	52
Gambar 3.2 Model Modern Waterfall.....	54
Gambar 4.1 Derajat Keanggotaan SUHU	58

Gambar 4.2 Derajat Keanggotaan KECEPATAN ANGIN	58
Gambar 4.3 Derajat Keanggotaan CURAH HUJAN	59
Gambar 4.4 Derajat Keanggotaan KELEMBABAN	59
Gambar 4.5 Diagram Acyclic Graph	60
Gambar 4.6 Diagram Kombinasi Fuzzy dan Bayesian Network	61
Gambar 4.7 Source Code Fuzzy Partition.....	66
Gambar 4.8 Derajat Keanggotaan Wang Mendel SUHU	67
Gambar 4.9 Derajat Keanggotaan Wang Mendel KECEPATAN ANGIN.....	68
Gambar 4.10 Derajat Keanggotaan Wang Mendel CURAH HUJAN	68
Gambar 4.11 Derajat Keanggotaan Wang Mendel KELEMBABAN	69
Gambar 4.12 Generate Fuzzy Function	72
Gambar 4.13 Daerah Kesimpulan Curah Hujan	78
Gambar 4.14 Source Code Defuzzification COG	78
Gambar 4.15. Demo Fungsi Bayesian Network.....	87
Gambar 4.16 Demo Fuzzy Partition Manual	88
Gambar 4.17 Demo Fuzzy Partition Otomatis.....	89
Gambar 4.18 Pembagian Data Dengan K-Folds	90

BAB I

PENDAHULUAN

1.1. Latar Belakang

Prediksi menjadi hal yang penting dalam kehidupan manusia. Karena pada tanggal 8 September 1900 di Amerika Serikat, penduduk Galveston berhasil diantisipasi dari munculnya badai besar berkat adanya alat prediksi cuaca (Rego & Li, 2010). Alat tersebut bekerja dengan mengobservasi tekanan atmosfer, tekanan angin di sepanjang pantai dan pasang surut air laut. Lalu hasil observasi akan digunakan untuk menghitung tekanan lintas batas yang membentuk akun gelombang badai. Setelah itu proses selanjutnya adalah menghitung luas terjadinya badai dan pasang surut air laut.

Prediksi merupakan sebuah pernyataan bahwa suatu peristiwa atau kejadian akan muncul. Prediksi sudah menjadi hal yang tidak asing dalam dunia penelitian, misalnya saja ada yang melakukan penelitian untuk memprediksi umur (Pasquier et al., 1999), memprediksi keberhasilan khasiat jangka panjang untuk hepatitis C kronis (Vandelli et al., 1999), prediksi eksitasi seismik untukantisipasi terjadinya gempa (Yamada, 1999), atau memprediksi terjadinya tsunami untuk mengantisipasi jatuhnya korban dari bencana tersebut (Sugimoto, Murakami, Kozuki, Nishikawa, & Shimada, 2002). Prediksi juga memberikan banyak manfaat dalam beberapa hal atau peristiwa di dunia terutama terhadap manusia, contohnya saja dengan adanya prediksi gempa, warga yang dekat dengan lokasi akan terjadinya gempa dapat dievakuasi lebih awal sehingga korban gempa dapat diminimalisir (Stiros, 1997), adapun dalam hal lainnya, dengan adanya prediksi lalu lintas jaringan dapat meningkatkan performa jaringan di suatu tempat sehingga melakukan *browsing* melalui internet menjadi lebih lancar (Yin, Lin, Sebastien, Li, & Min, 2005). Oleh karena itulah prediksi menjadi hal yang penting.

Munculnya prediksi cuaca memberikan banyak manfaat bagi manusia, terutama untuk para nelayan, karena berkat adanya prediksi cuaca, pelayaran ke laut dapat lebih terkontrol dan tidak membahayakan para nelayan. Cuaca sendiri

merupakan kondisi udara (bisa terkait temperatur, penyinaran matahari, kelembaban, kecepatan angin, perubahan arah angin, ataupun arah angin) dengan keadaan tertentu di suatu wilayah yang relatif sempit dalam jangka waktu yang singkat.

Cuaca telah banyak dijadikan topik prediksi untuk studi kasus dalam penelitian dengan tujuan yang bermacam-macam, beberapa diantaranya adalah untuk meningkatkan pendeskripsian uap air di atmosfer (Leiming, Fuhai, Qiqige, & Guangqiang, 2012), uji kelayakan sistem prediksi cuaca (Chaves, Ross, & Krishnamurti, 2005), merancang strategi dalam prediksi cuaca (Aisjah & Arifin, 2011), meningkatkan akurasi prediksi cuaca (Corne, Dissanayake, Peacock, Galloway, & Owens, 2014), dan membangun model baru dalam memprediksi cuaca (Nandar, 2009). Model atau metode yang digunakan dalam penelitiannya pun bermacam-macam, beberapa diantaranya ada yang menggunakan *numerical weather prediction model* (Sundqvist, Berge, & Kristjansson, 1989; Kimura, 2002; Tang, Capon, Forbes, & Clark, 2009), persamaan dasar atmosfer (Kasahara, 1974), dan algoritma *ORIGIN weather prediction* (Corne, Dissanayake, Peacock, Galloway, & Owens, 2014).

Dari berbagai metode yang ada, *Bayesian network* merupakan salah satu metode yang telah banyak digunakan dalam masalah prediksi, diantaranya untuk prediksi lokasi objek (Zhang et al., 2009), prediksi *error* pada mesin (Wang & Zhou, 2014), prediksi biaya listrik dalam satu hari ke depan (Vahidinasab & Jadid, 2008), dan prediksi ketahanan aplikasi berorientasi (Koten & Gray, 2005). Metode ini banyak digunakan dalam masalah prediksi karena efektif dalam mengevaluasi ketidakpastian dan memodelkan *hidden knowledge* (Tang & Liu, 2007; Nandar, 2009; D'Angelo et al., 2014; Wang & Zhou, 2014). Metode ini hanya bekerja pada data diskrit dan berbasis pada teori probabilitas juga graf (Hsu, Huang, & Chang, 2008; Zhao & Zhang, 2013; Roos, Bonnevey, & Gavin, 2016). Dikarenakan metode ini hanya bekerja dengan nilai diskrit, maka dibutuhkan proses diskritisasi terhadap data yang bersifat kontinu (Chai & Wang, 2011).

Fuzzy logic menjadi salah satu solusi untuk mendiskritkan nilai-nilai yang bersifat kontinu atau tak hingga (seperti umur yang memiliki nilai tak hingga, bisa

1, 20, atau 100 tahun), karena *fuzzy* akan membagi nilai-nilai kontinu kedalam beberapa kelompok menggunakan fungsi keanggotaan (Zadeh, Fuzzy Sets, 1965; Tang & Liu, 2007; Zhao & Zhang, 2013; Yao, Li, Li, & Wang, 2015). *Fuzzy* juga banyak digunakan oleh peneliti, karena dapat digunakan untuk mengatasi ketidakpastian (Battaini, Casciati, & Faravelli, 1998). Pada metode ini memiliki tahap membangun fungsi keanggotaan, di mana fungsi tersebut dibangun berdasarkan seorang ahli (Zadeh, Communication Fuzzy Algorithms, 1968). Karena fungsi keanggotaan diperoleh dari ahli, proses pembagian nilai dari data kedalam beberapa kelompok (*fuzzy partition*) menjadi kurang efisien (Chai & Wang, 2011). Maka dari itu dibutuhkan sebuah metode atau cara agar *fuzzy partition* dapat dihasilkan secara otomatis.

Li-Xin Wang dan Jerry M. Mendel membuat sebuah metode yang dapat digunakan untuk menghasilkan aturan *fuzzy* atau fungsi keanggotaan dari data numerik secara otomatis (Wang & Mendel, 1991). Sehingga proses *fuzzy partition* dapat dilakukan tanpa harus melakukan validasi terhadap ahli untuk membangun fungsi keanggotaan. Oleh karena itu, metode ini penulis gunakan kedalam fungsi *fuzzy partition*.

Berdasarkan segala penjelasan yang telah disampaikan, penulis ingin membangun sebuah model kombinasi metode *fuzzy* dan Bayesian network kedalam bentuk aplikasi yang dapat digunakan untuk melakukan prediksi cuaca. Mulai dari merancang metode, mengimplementasikan metode kedalam program, hingga menguji kombinasi metode *Fuzzy* dan Bayesian network.

1.2. Rumusan Masalah

Berdasarkan permasalahan yang muncul pada latar belakang, maka beberapa masalah yang ingin diselesaikan yaitu :

1. Bagaimana metode *fuzzy* dan *Bayesian network* dapat menghitung prediksi cuaca ?
2. Bagaimana penggunaan metode *fuzzy* dan *Bayesian network* agar dapat dilakukan secara efektif ?

3. Bagaimana tingkat presisi serta kecepatan dari metode *fuzzy* dan *Bayesian network* terhadap prediksi cuaca ?

1.3. Tujuan

Berikut adalah beberapa tujuan yang ingin dicapai dalam penelitian ini :

1. Merancang model prediksi cuaca dengan menggunakan kombinasi metode *Fuzzy* dan *Bayesian network*
2. Implementasi kombinasi metode *Fuzzy* dan *Bayesian network* dengan memanfaatkan bahasa pemrograman R
3. Melakukan eksperimen kombinasi metode *Fuzzy* dan *Bayesian network*

1.4. Batasan Masalah

Terdapat beberapa batasan masalah dari penelitian ini, diantaranya :

1. Data diambil dari salah satu *website* yang dibangun oleh BMKG yakni **dataonline.bmkg.go.id** dan bersifat terbuka (siapa pun bisa mengunduh data BMKG secara gratis)
2. Data yang digunakan meliputi suhu rata-rata, kelembaban rata-rata, curah hujan, dan kecepatan angin rata-rata
3. Data diambil dari awal bulan Januari 2010 – akhir Desember 2016 dengan alokasi Kota Bandung
4. Aplikasi digunakan untuk memprediksi cuaca dalam rentang satu hari kedepan
5. Penelitian dilakukan untuk melakukan analisis akurasi dan nilai *error* dari gabungan metode *bayesian network* dan *fuzzy logic*

1.5. Sistematika Penulisan

Berikut ini adalah sistematika penulisan yang dilakukan dalam menyusun skripsi :

BAB 1 PENDAHULUAN

Bab ini berisikan masalah yang diangkat dalam penelitian meliputi latar belakang (menceritakan tentang prediksi secara umum, tentang prediksi cuaca, *Bayesian network* sebagai metode untuk mengatasi masalah prediksi, dan metode *fuzzy* untuk mengatasi masalah yang ada pada *Bayesian network*), rumusan masalah, batasan masalah, tujuan penelitian, dan sistematika penulisan.

BAB 2 KAJIAN TEORI

Berisi tentang kajian teori yang digunakan dalam penelitian meliputi teori tentang Naïve Bayes, Bayesian Network, *fuzzy logic*, *fuzzy Wang Mendel* Bayesian Network dan Bahasa R.

BAB 3 METODOLOGI PENELITIAN

Berisikan dasar teori mengenai metodologi yang digunakan dalam melakukan penelitian meliputi desain rancangan penelitian, subjek penelitian, alat dan bahan penelitian.

BAB 4 HASIL PENELITIAN DAN PEMBAHASAN

Bab ini berisi tentang hasil dan perancangan aplikasi dari penelitian yang dilakukan, hasil penelitian terdiri dari nilai akurasi, tingkat *error*, dan probabilitas terhadap prediksi cuaca.

BAB 5 KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran berdasarkan penelitian yang dilakukan dari mulai perumusan masalah hingga selesai.

LAMPIRAN

Berisi data cuaca iklim yang digunakan untuk penelitian, panduan instalasi program R, panduan menggunakan *package* cvTools untuk pembagian data menggunakan *k-Folds*, dan identitas penulis.

BAB II

KAJIAN TEORI

2.1. Definisi dan Prediksi Cuaca

Cuaca merupakan salah satu elemen yang ada dalam kehidupan sehari – hari pada planet bumi yang selalu hadir dan berubah – ubah (LaFontaine, 2004). Begitu melangkah keluar, kondisi cuaca dapat langsung terlihat dari keadaan suhu, kelembaban, awan, dan angin. Studi ilmiah yang mempelajari tentang cuaca disebut dengan *meteorology* (meteorologi). Sebutan meteorologi itu sendiri diambil dari kata *measure* (mengukur) dalam bahasa Yunani.

Cuaca dapat terbentuk karena adanya unsur – unsur yang mempengaruhi. Unsur – unsur tersebut adalah sinar matahari, suhu, kelembaban udara, tekanan udara, angin, curah hujan, dan awan (LaFontaine, 2004). Setiap unsur akan saling mempengaruhi unsur yang lainnya, beberapa diantaranya adalah, lamanya penyinaran matahari ke bumi, ketinggian suatu tempat, keadaan awan, keadaan tumbuhan, dan sudut penyinaran matahari akan mempengaruhi suhu di permukaan bumi, atau tingkat tekanan pada suatu daerah akan mempengaruhi kecepatan dan arah angin yang berhembus di daerah tersebut.

Kita semua terpengaruh oleh cuaca, namun sebagian besar waktu dari cuaca yang dialami terbilang biasa (Allaby & Garratt, 2003). Bisa jadi cuaca yang muncul dari waktu ke waktu adalah baik – baik saja atau basah maupun hangat atau kering. Secara detil, prediksi cuaca merupakan hal yang sulit, namun secara umum dapat dilakukan. Rata – rata kondisi cuaca dalam setiap periode dari beberapa dekade, cuaca dalam hari – hari biasa, mengandung iklim dari suatu daerah.

Prediksi cuaca menjadi hal yang sudah tidak asing lagi bagi manusia terutama dalam lingkup penelitian. Memprediksi perubahan kondisi cuaca dengan akurasi yang wajar telah menjadikan prediksi cuaca sebagai kontribusi berharga dalam kehidupan sehari – hari, sebagai contohnya, pemberitahuan cuaca harian memberikan pengaruh terhadap memilih pakaian dan perlengkapan, ataupun

peramalan badai yang berbahaya dapat menyelamatkan nyawa dan harta benda (LaFontaine, 2004). Banyak penelitian yang dilakukan dalam memprediksi cuaca, pada Tabel 2.1 terdapat beberapa contoh penelitian prediksi cuaca dengan hasil yang bermacam – macam.

Tabel 2.1 Hasil Penelitian Prediksi Cuaca

No.	Judul Penelitian	Nama Peneliti	Metode yang Digunakan	Hasil	Sumber
1.	Accurate Localized Short Term Weather Prediction for Renewables Planning	David Corne; Manula Dissanayake; Andrew Peacock; Stuart Galloway Eddie Owens	Numerical Weather Prediction (NWP) dan ORIGIN Weather Prediction Algorithm (WPA)	Gabungan NWP dan WPA lebih baik dari optimasi-lokal MetOffice-berbasis prediksi.	(Corne, Dissanaya ke, Peacock, Galloway, & Owens, 2014)
2.	Bayesian Network Probability Model for Weather Prediction	Aye Nandar	Bayesian Network	Akurasi dari metode dapat diterima, namun untuk analisis selanjutnya dibutuhkan efisiensi	(Nandar, 2009)
3.	Condensation and Cloud Parameterization Studies with a Mesoscale Numerical Weather Prediction Model	Hilding Sundqvist; Erik Berge; Jon Egill Kristjanssson	Mesoscale Model	Jumlah dari akumulasi curah hujan terprediksi secara realistik	(Sundqvist , Berge, & Kristjansson, 1989)
4.	Energy and Numerical Weather Prediction	Edward N. Lorenz	<i>Potential Energy Equation, Gross static stability, dan Jacobian</i>	Adanya kemungkinan model ini tidak dapat digunakan untuk peramalan jangka pendek	(Lorenz, 1960)
5.	Numerical Weather Prediction in	Ma Leiming;	<i>Short-range Numerical Model</i>	3DVAR dan <i>Global Positioning</i>	(Leiming, Fuhai, & Qiqige, &

Yangtze River Delta Region with Assimilation of AWS and GPS/PWV data	Wuyun Qiqige; Geng Fuhai; Zhou Guangqiang	<i>Systems</i> (SSNS) dan <i>Three-dimensional Variational</i> (3DVAR)	<i>System</i> (GPS) dapat mengurangi <i>Root Mean Square Error</i> (RMSE) dari perhitungan suhu	Guangqiang, 2012)
--	---	--	---	-------------------

Berdasarkan segala hasil penelitian yang dilakukan dalam hal prediksi cuaca seperti ditampilkan pada Tabel 2.1, dapat disimpulkan bahwa penelitian memberikan banyak pengaruh positif dalam hal prediksi cuaca, salah satunya adalah untuk meningkatkan tingkat akurasi model prediksi. Dan dengan meningkatkan tingkat akurasi, maka hasil prediksi cuaca yang akan terjadi akan lebih baik. Lalu baiknya hasil akurasi prediksi cuaca akan sangat bermanfaat bagi manusia, salah satunya adalah antisipasi warga terhadap munculnya badai bisa dilakukan dengan lebih baik, sehingga dapat meminimalisir jatuhnya korban yang diakibatkan oleh badai.

2.2. Definisi dan Algoritma Naïve Bayes

Naïve Bayes merupakan metode klasifikasi berbasiskan teorema Bayes dan teorema tersebut menyatakan seberapa jauh derajat kepercayaan subjektif harus berubah secara rasional ketika ada petunjuk baru dan penemu dari teorema ini adalah Thomas Bayes (Leroy, Miller, Rosemblat, & Browne, 2008). Naïve Bayes telah banyak digunakan karena metode ini sangat efisien dan mampu menggabungkan bukti atau nilai dari jumlah fitur yang sangat banyak (Zhang H. , 2004; Leroy, Miller, Rosemblat, & Browne, 2008; Hristea, 2013). Metode ini mudah untuk dipahami dan memiliki induksi (penarikan kesimpulan berdasarkan keadaan tertentu) yang sangat cepat, hanya membutuhkan sebuah sinyal melalui data (Hsu, Huang, & Chang, 2008). Namun, dari segi algoritmanya, metode ini hanya bekerja pada data yang bersifat kategorikal atau diskrit, misalnya data jenis kelamin hanya memiliki nilai laki-laki atau perempuan, lalu data suhu hanya memiliki nilai dingin, hangat, atau panas (Chai & Wang, 2011). Oleh karena itu, metode ini tidak bisa digunakan terhadap data campuran, yakni ada data numerik dan diskrit.

Metode Naïve Bayes menggunakan teorema Bayes dengan bentuk persamaan sebagai berikut (Leroy, Miller, Rosembat, & Browne, 2008) :

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad (2.1)$$

Keterangan :

- $P()$ adalah probabilitas
- A adalah variabel yang probabilitasnya akan dihitung
- B adalah variabel yang menjadi kondisi dari variabel yang akan dicari probabilitasnya

Dari persamaan (2. 1), misalkan A adalah tinggi_badan = tinggi dan B adalah ukuran_badan = besar maka,

- $P(A|B)$ adalah probabilitas A di mana B, maka $P(A|B)$ adalah probabilitas tinggi_badan bernilai tinggi di mana ukuran_badan nya besar.
- $P(A \cap B)$ adalah probabilitas tinggi_badan nya tinggi dan ukuran_badan nya besar. Nilainya adalah jumlah data yang tinggi_badan nya tinggi dan ukuran_badan nya besar.

$P(B)$ adalah probabilitas ukuran_badan nya besar dan nilainya adalah jumlah data yang mempunyai variabel ukuran_badan nya adalah besar.

Pseudocode Naïve Bayes**Input** : varA, varB, resA, resB, dataInput**Output** : probA**Proses** :

accomA.B = 0

accomB = 0

```

for( sum(dataInput)){
    if(dataInput[varA]=resA and dataInput[varB]=resB){
        accomA.B + 1
    }
    if(dataInput[varB]=resB){
        accomB + 1
    }
}

probA = accomA.B / accomB
return(probA)

```

Gambar 2.1 Pseudocode Naïve Bayes

Contoh pseudocode Naïve Bayes ditampilkan pada Gambar 2.1. Dapat dijelaskan bahwa metode ini terdiri dari beberapa masukan yaitu varA, varB, resA, resB, dan dataInput. Di mana varA adalah variabel yang akan dicari probabilitasnya, varB adalah variabel yang menjadi keadaan, resA adalah nilai dari varA, resB adalah nilai dari varB, dan dataInput berisikan data yang digunakan untuk menghitung probabilitas berbentuk matriks. Keluaran dari metode ini adalah probA yaitu nilai probabilitas dari variabel yang dicari kemungkinannya. Dan metode ini memiliki beberapa proses, yang pertama, tampung jumlah varA dengan keadaan varB kedalam accomA.B; kedua, tampung jumlah munculnya varB kedalam accomB; ketiga, tampung hasil kemungkinannya kedalam probA.

Tabel 2.2 Data Mahasiswa

No	Jenis Kelamin	Status Mahasiswa	Status Pernikahan	IPK Semester 1-6	Status Kelulusan
1	Laki – laki	Mahasiswa	Belum	3.17	Tepat
2	Laki – laki	Bekerja	Belum	3.30	Tepat
3	Perempuan	Mahasiswa	Belum	3.01	Tepat
4	Perempuan	Mahasiswa	Menikah	3.25	Tepat
5	Laki – laki	Bekerja	Menikah	3.20	Tepat
6	Laki – laki	Bekerja	Menikah	2.50	Terlambat
7	Perempuan	Bekerja	Menikah	3.00	Terlambat
8	Perempuan	Bekerja	Belum	2.70	Terlambat
9	Laki – laki	Bekerja	Belum	2.40	Terlambat
10	Perempuan	Mahasiswa	Menikah	2.50	Terlambat
11	Perempuan	Mahasiswa	Belum	2.50	Terlambat
12	Perempuan	Mahasiswa	Belum	3.50	Tepat
13	Laki – laki	Bekerja	Menikah	3.30	Tepat
14	Laki – laki	Mahasiswa	Menikah	3.25	Tepat
15	Laki – laki	Mahasiswa	Belum	2.30	Terlambat

Adapun contoh kasusnya, misal diberikan sebuah data mahasiswa yang terdiri dari informasi jenis kelamin, status mahasiswa, status pernikahan, IPK dari semester 1 sampai 6, dan status kelulusan (Tabel 2.2). Lalu dari data tersebut kita dimintai untuk menghitung probabilitas status kelulusan dengan keadaan status mahasiswa nya tidak sedang bekerja, status pernikahan nya belum, jenis kelamin nya laki-laki, dan IPK semester 1-6 adalah 2,70. Berikut ini cara untuk mendapatkan nilai probabilitasnya :

1. Diketahui bahwa

Jenis Kelamin = Laki-laki

Status Mahasiswa = Mahasiswa

Status Pernikahan = Belum
 IPK = 2.70

2. Proses hitung

Hitung jumlah probabilitas status kelulusan yang tepat dan terlambat, asumsikan SK adalah Status Kelulusan, maka hasilnya

$$\begin{aligned} P(\text{SK} = \text{Tepat}) &= \frac{\text{Jumlah SK bernilai Tepat}}{\text{Jumlah Seluruh Data}} \\ &= \frac{8}{15} \\ &= 0.53 \end{aligned}$$

$$\begin{aligned} P(\text{SK} = \text{Terlambat}) &= \frac{\text{Jumlah SK bernilai Terlambat}}{\text{Jumlah Seluruh Data}} \\ &= \frac{7}{15} \\ &= 0.47 \end{aligned}$$

Setelah mendapatkan nilai probabilitas SK, hitung probabilitas variabel lainnya namun mengikuti kondisi SK, asumsikan JK untuk Jenis Kelamin, SM untuk Status Mahasiswa, lalu SP untuk Status Pernikahan,

$$\begin{aligned} P(\text{JK} = \text{Laki} - \text{laki} | \text{SK} = \text{Tepat}) &= \frac{\text{Jumlah JK} = \text{Laki} - \text{laki} \text{ dan SK} = \text{Tepat}}{\text{Jumlah SK} = \text{Tepat}} \\ &= \frac{5}{8} \\ &= 0.625 \end{aligned}$$

$$P(\text{JK} = \text{Laki} - \text{laki} | \text{SK} = \text{Terlambat}) = \frac{3}{7} = 0.43$$

$$P(\text{SM} = \text{Mahasiswa} | \text{SK} = \text{Tepat}) = \frac{5}{8} = 0.625$$

$$P(\text{SM} = \text{Mahasiswa} | \text{SK} = \text{Terlambat}) = \frac{3}{7} = 0.43$$

$$P(\text{SP} = \text{Belum} | \text{SK} = \text{Tepat}) = \frac{4}{8} = 0.5$$

$$P(\text{SP} = \text{Belum} | \text{SK} = \text{Terlambat}) = \frac{4}{7} = 0.57$$

$$P(\text{IPK} = 2.70 | \text{SK} = \text{Tepat}) = \frac{0}{8} = 0$$

$$P(\text{IPK} = 2.70 | \text{SK} = \text{Terlambat}) = \frac{1}{7} = 0.143$$

Selanjutnya menyimpulkan nilai probabilitas SK dengan kondisi JK = Laki-Laki, SM = Mahasiswa, SP = Belum, dan IPK = 2.70. Karena SK memiliki dua keluaran yaitu TEPAT dan TERLAMBAT, maka probabilitas SK = TEPAT dan SK = TERLAMBAT harus dihitung dan berikut ini cara menghitung probabilitasnya masing – masing,

a) Hitung probabilitas SK (Status Kelulusan) = Tepat

$$\begin{aligned}
 &P(\text{SK=Tepat} \mid \text{JK = Laki-laki, SM = Mahasiswa, SP = Belum, IPK = 2.70}) \\
 &= P(\text{JK = Laki-laki} \mid \text{SK = Tepat}) * \\
 &\quad P(\text{SM = Mahasiswa} \mid \text{SK = Tepat}) * \\
 &\quad P(\text{SP = Belum} \mid \text{SK = Tepat}) * \\
 &\quad P(\text{IPK = 2.70} \mid \text{SK = Tepat}) * \\
 &\quad P(\text{SK = Tepat}) \\
 &= 0.625 * 0.625 * 0.5 * 0 * 0.53 \\
 &= 0
 \end{aligned}$$

Untuk menghitung probabilitas SK = Tepat, kalikan seluruh nilai probabilitas yang keadaan SK = Tepat, mulai dari probabilitas JK = Laki-laki, SM = Mahasiswa, SP = Belum, IPK = 2.70 di mana keadaan SK = Tepat.

b) Hitung probabilitas SK (Status Kelulusan) = Terlambat

$$\begin{aligned}
 &P(\text{SK=Terlambat} \mid \text{JK = Laki-laki, SM = Mahasiswa, SP = Belum, IPK = 2.70}) \\
 &= P(\text{JK = Laki-laki} \mid \text{SK = Terlambat}) * \\
 &\quad P(\text{SM = Mahasiswa} \mid \text{SK = Terlambat}) * \\
 &\quad P(\text{SP = Belum} \mid \text{SK = Terlambat}) * \\
 &\quad P(\text{IPK = 2.70} \mid \text{SK = Terlambat}) * \\
 &\quad P(\text{SK = Terlambat}) \\
 &= 0.43 * 0.43 * 0.57 * 0.143 * 0.47 \\
 &= 0.0071
 \end{aligned}$$

Untuk menghitung probabilitas SK = Terlambat, kalikan seluruh nilai probabilitas yang keadaan SK = Terlambat, mulai dari probabilitas JK = Laki-laki, SM = Mahasiswa, SP = Belum, IPK = 2.70 di mana keadaan SK = Terlambat.

Berdasarkan proses perhitungan yang telah dilakukan, maka nilai probabilitas mahasiswa akan lulus dengan tepat (SK = Tepat) adalah 0 dan terlambat (SK = Terlambat) adalah 0.0071, dengan keadaan mahasiswa tersebut adalah seorang laki-laki (JK = Laki - laki), statusnya tidak sedang bekerja (SM = Mahasiswa), belum menikah (SP = Belum), dan nilai IPK semester 1 – 6 adalah 2.70 (IPK = 2.70).

2.3. Definisi dan Algoritma Bayesian Network

Bayesian Network merupakan salah satu *Probabilistic Graphical Model* yang sederhana dan dibangun dari teori probabilitas juga teori graf (Koten & Gray, 2005; Ben-Gal, 2008; Nandar, 2009; Harahap, Sakamoto, & Nishi, 2010; Roos, Bonnevey, & Gavin, 2016). Teori probabilitas adalah cara untuk menyampaikan bukti bahwa suatu peristiwa telah terjadi sedangkan teori graf adalah cabang ilmu yang mempelajari himpunan benda-benda atau variabel-variabel (biasa disebut dengan node) yang terhubung oleh sisi atau panah seperti ditampilkan pada Gambar 2.2. Teori probabilitas berhubungan langsung dengan data, sedangkan teori graf berhubungan langsung dengan bentuk representasi yang ingin didapatkan. Bayesian Network sudah lama menjadi populer untuk digunakan dalam merepresentasikan ketidakpastian di sistem pakar (Zheng & Liu, 2010), diantaranya digunakan untuk membangun sistem ahli dalam pengkategorian morfologi (Morrison et al., 2002), untuk mendiagnosis kegagalan (Wang & Zhou, 2014), dan estimasi arus lalu lintas (Weber, Medina-Oliva, Simon, & Iung, 2010). Ada beberapa hal yang diperlukan dalam proses analisis menggunakan metode Bayesian Network, diantaranya :

1. Analisis awal untuk permasalahan

Menganalisa permasalahan yang akan diselesaikan dengan metode Bayesian network, salah satunya adalah memeriksa adanya data kontinu atau diskrit pada data yang akan digunakan

2. Pengumpulan data

Mengumpulkan data yang akan digunakan untuk menghitung probabilitas, sumber data bisa dari *website* atau dari instansi tertentu terkait permasalahan yang diangkat

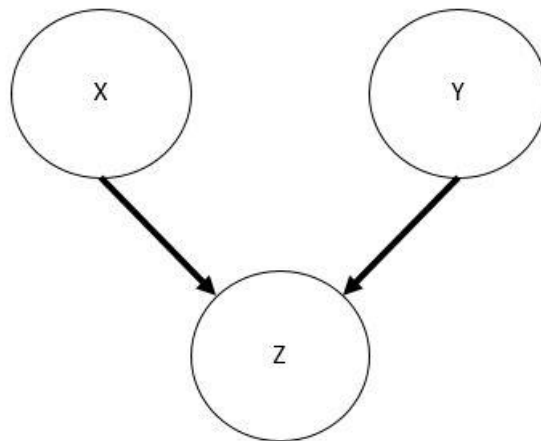
3. Pembentukan diagram sebab akibat (diagram dapat dibuat berdasarkan data yang sudah ada atau berdasarkan ahli)

Membangun *diagram acyclic graph* dari data yang digunakan, bisa diagram bisa didapatkan dari pakar atau menggunakan diagram yang dibangun sendiri dan telah tervalidasi oleh pakar terkait masalah yang diangkat

4. Proses perhitungan

Setelah *diagram acyclic graph* terbangun, langkah selanjutnya adalah menghitung probabilitas dari variabel yang dicari berdasarkan masalah yang diangkat

Probabilistic graphical model adalah alat yang digunakan untuk memodelkan karakteristik spasial dan temporal dari proses dinamis, contohnya adalah *hidden Markov model* dan Bayesian Network yang telah banyak digunakan untuk memodelkan fenomena dinamis seperti suara dan aktivitas (Corso, Ye, & Hager, 2005; Liu, Yue, & Gao, 2011). *Probabilistic graphical model* merupakan sebuah kerangka yang efisien untuk menginterpretasikan distribusi kompleks kedalam ruang dimensi ganda menggunakan representasi berbasis graf (Chen & Zabaras, 2014). Dalam Bayesian Network, *probabilistic graphical model* disebut juga dengan *directed acyclic graph* (DAG). DAG merupakan diagram berisikan node yang merepresentasikan peristiwa dan peristiwa tersebut digambarkan menggunakan garis terarah yang memiliki maksud bahwa adanya hubungan antar node (Koten & Gray, 2005; Nandar, 2009; Weber, Medina-Oliva, Simon, & Iung, 2010). Gambar 2.2 merupakan salah satu contoh sederhana DAG.



Gambar 2.2 Contoh *Diagram Acyclic Graph*

Dapat dijelaskan dari Gambar 2.2, bahwa X, Y, dan Z adalah node yang biasanya diisi dengan nama dari data atau variabel, misal asumsikan bahwa X adalah air, Y adalah api, dan Z adalah tanah, maka dapat disimpulkan bahwa air memiliki hubungan dengan tanah tapi tidak memiliki hubungan dengan api dan sebaliknya bahwa api memiliki hubungan dengan tanah tapi tidak memiliki hubungan dengan air.

Lalu dalam Bayesian Network ini, hubungan antar node seperti yang ditampilkan pada Gambar 2.4 didefinisikan sebagai *conditional probability* atau biasa lebih dikenal dengan sebutan *conditional probability table* (CPT) (Koten & Gray, 2005; Weber, Medina-Oliva, Simon, & Iung, 2010). CPT didefinisikan sebagai deret variabel acak bersifat diskrit yang didemonstrasikan kedalam probabilitas marjinal dari satu variabel ke variabel terhubung seperti yang ditampilkan pada Gambar 2.4. Sama halnya dengan Naïve Bayes, nilai CPT itu sendiri didapatkan berdasarkan teorema atau persamaan Bayes, misal dari Gambar 2.4, kita ingin mengetahui cara mendapatkan CPT dari RAIN, di mana RAIN terdiri dari nilai T dan F, lalu diketahui bahwa total data adalah 10 yang terdiri dari 2 buah nilai T dan 8 buah nilai F, maka untuk mendapatkan nilai CPT dari $RAIN = T$ adalah total jumlah T dibagi total data yang ada yaitu 2 dibagi 10 dan sebaliknya nilai CPT dari $RAIN = F$ adalah total jumlah F dibagi total data yang ada yaitu 8 dibagi 10. Adapun persamaan Bayesian network ini dituliskan sebagai berikut :

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \prod X_i) \quad (2.2)$$

Keterangan :

- $P()$ adalah probabilitas
- X_i adalah variabel ke – i
- i adalah indeks dari variabel
- n adalah total data yang digunakan

Dari persamaan 2. 2 dapat dijelaskan bahwa untuk menghitung probabilitas dari sebuah variabel X_i adalah dengan mengalikan seluruh kemungkinan dari variabel X_i terhadap variabel yang lainnya. Untuk lebih detilnya, bisa dilihat pada contoh kasus.

```

Pseudocode Bayesian Network
Input      : DAG, varA, varB, resA, resB, dataInput
Output     : probA
Proses      :
    CPT = createCPT( DAG, dataInput, varA, varB, resA, resB)

    accomProbA.B = 1
    for(sum(CPT)){
        accomProbA.B = accomProbA.B * CPT[varA,varB]
    }

    accomProbB = 1
    for(sum(CPT)){
        accomProbB = accomProbB * CPT[varB,]
    }

    probA = accomProbA.B / accomProbB

    return(probA)

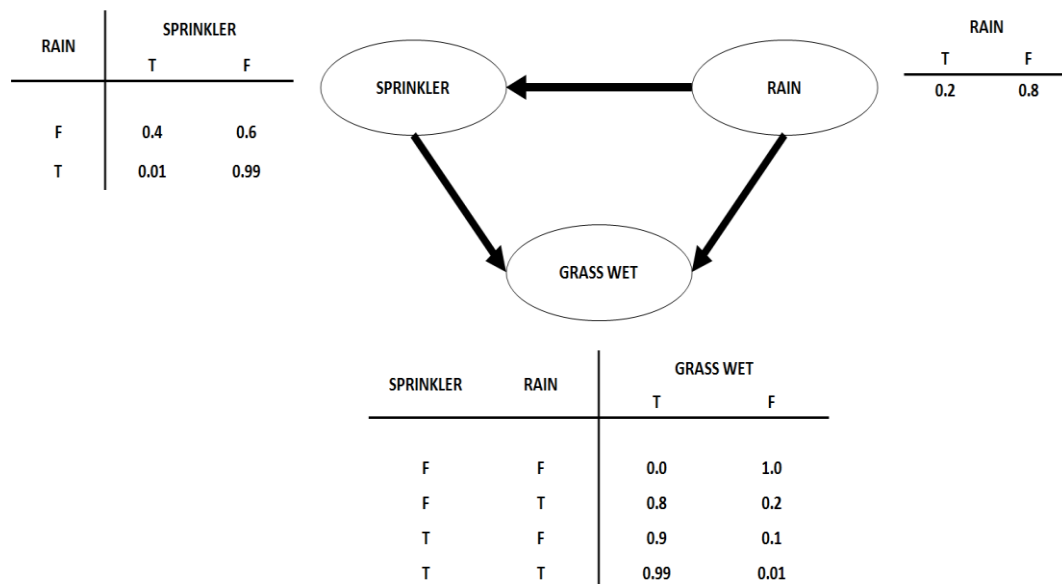
```

Gambar 2.3 Pseudocode Bayesian Network

Contoh pseudocode metode Bayesian network ditampilkan pada Gambar 2.3. Metode ini memiliki masukan DAG, varA, varB, resA, resB, dan dataInput, di mana DAG berisikan diagram acyclic graph yang dibentuk kedalam matriks, varA adalah variabel yang akan dicari probabilitasnya, varB adalah variabel yang menjadi keadaan, resA adalah nilai dari varA, resB adalah nilai dari varB, dan dataInput berisikan data yang digunakan untuk menghitung probabilitas berbentuk

matriks. Keluaran dari metode ini adalah probabilitas dari variabel yang dicari nilai kemungkinannya, ditampilkan dalam probA. Lalu metode ini terdiri dari beberapa tahapan, yang pertama, membangun CPT dengan parameter DAG, dataInput, varA, varB, resA, dan resB; kedua, hitung nilai probabilitas dari varA dengan kondisi varB melalui CPT yang terbangun dan simpan nilainya ke accomProbA.B; ketiga, hitung probabilitas dari varB melalui CPT dan simpan hasilnya ke accomProbB; keempat, hitung hasil probabilitasnya dan simpan dalam probA.

Contoh kasus dalam Bayesian Network, misalnya didapatkan nilai CPT dari data penyebab rumput basah yang berisikan variabel RAIN, SPRINKLER, dan GRASS WET dengan seluruh variabelnya bernilai T untuk TRUE dan F untuk FALSE seperti yang ditampilkan pada Gambar 2.4.



Gambar 2.4 Contoh *Conditional Probability Table* (CPT), Sumber:

<https://tex.stackexchange.com/questions/108034/how-do-i-make-bayesian-networks>

Berdasarkan data tersebut, kita dimintai untuk menghitung probabilitas munculnya hujan (RAIN = T) ketika alat penyiram otomatis sedang tidak aktif (SPRINKLER = F) dan keadaan rumputnya basah (GRASS WET = T). Berikut ini adalah cara penyelesaiannya :

a) Tentukan *Joint Probability Function*

Joint Probability Function (JPF) merupakan distribusi peluang gabungan dari dua atau lebih variabel acak, maka JPF dari variabel RAIN berdasarkan CPT yang dihasilkan seperti terlihat pada Gambar 2. 2 di mana R untuk RAIN, S untuk SPRINKLER, dan G untuk GRASS WET adalah,

$$P(G, S, R) = P(G|S, R) P(S|R) P(R) \quad (2. 3)$$

Di mana,

- $P(G, S, R)$ adalah distribusi peluang untuk variabel GRASS WET, SPRINKLER, dan RAIN
- $P(G|S, R)$ adalah peluang untuk variabel GRASS WET mengikuti kondisi SPRINKLER dan RAIN
- $P(S|R)$ adalah peluang dari variabel SPRINKLER dengan kondisi RAIN
- $P(R)$ adalah peluang variabel RAIN dari keseluruhan data yang ada

b) Memodelkan persamaan Bayesian

Berdasarkan CPT nya maka untuk menghitung probabilitas terjadinya hujan ($R = T$) jika alat penyiram otomatis tidak aktif ($S = F$) dan rumput sedang basah ($G = T$), maka persamaan Bayesian nya adalah :

$$P(R = T|S = F, G = T) = \frac{P(G = T, S = F, R = T)}{P(G = T, S = F, R)} \quad (2. 4)$$

Di mana,

- $P(R = T|S = F, G = T)$ adalah probabilitas terjadi hujan ($R = T$) ketika alat penyiram otomatis nyala ($S = F$) dan keadaan rumput basah ($G = T$)
- $P(G = T, S = F, R = T)$ adalah JPF untuk $G = T$, $S = F$, dan $R = T$

- $P(G = T, S = F, R)$ adalah JPF untuk $G = T, S = F$, terhadap setiap keadaan R

c) Menghitung probabilitas yang dibutuhkan

Dengan ekspansi JPF yaitu $P(G, S, R)$ dan probabilitas dari CPT, setiap data dapat dikalkulasi kedalam penjumlahan pembilang dan penyebut. Berdasarkan model Bayesian yang dibangun, maka dibutuhkan untuk menghitung $P(G = T, S = F, R = T)$ dan $P(G = T, S = F, R = F)$,

$$\begin{aligned} P(G = T, S = F, R = T) &= P(G = T|S = F, R = T)P(S = F|R = T)P(R = T) \\ &= 0.8 * 0.99 * 0.2 \\ &= 0.1584 \end{aligned}$$

$$\begin{aligned} P(G = T, S = F, R = F) &= P(G = T|S = F, R = F)P(S = F|R = F)P(R = F) \\ &= 0.0 * 0.6 * 0.8 \\ &= 0 \end{aligned}$$

d) Menghitung probabilitas yang ingin dicari

Setiap nilai probabilitas yang dibutuhkan sudah didapatkan, terakhir adalah menghitung probabilitas yang ingin dicari yakni $P(R = T|S = F, G = T)$, berikut adalah prosesnya,

$$\begin{aligned} P(R = T|S = F, G = T) &= \frac{P(G = T, S = F, R = T)}{P(G = T, S = F, R = T) + P(G = T, S = F, R = F)} \\ &= \frac{0.1584}{0.1584 + 0} \\ &= 1 \end{aligned}$$

Maka dapat disimpulkan bahwa nilai probabilitas terjadinya hujan apabila alat penyiram otomatis mati dan rumput basah adalah $P(R = T|S = F, G = T) * 100$ yaitu 100 %.

2.4. Teori Fuzzy Set

Teori *fuzzy set* (himpunan *fuzzy*) dikemukakan oleh Zadeh, sebagai lanjutan dari teori himpunan klasik untuk model himpunan (kumpulan objek yang memiliki

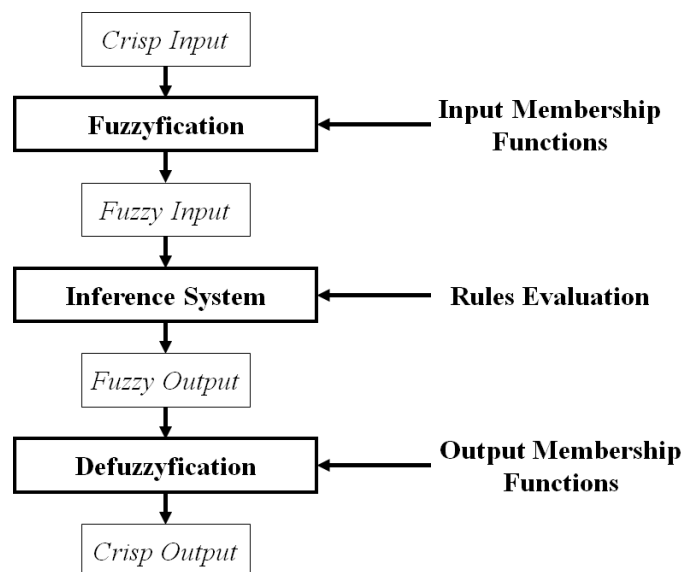
sifat yang dapat didefinisikan dengan jelas) berisi elemen – elemen yang memiliki derajat keanggotaan (Zadeh, The Concept of a Linguistic Variable and its Application to Approximate Reasoning - 1, 1975; Riza L. S., Bergmeir, Herrera, & Benitez, 2015). Daripada hanya memiliki dua nilai yakni anggota atau non-anggota, himpunan *fuzzy* memungkinkan untuk menetapkan tingkat derajat dari himpunan keanggotaan, ditentukan oleh nilai antara 0 dan 1. Nilai derajat 1 memiliki arti bahwa sebuah objek merupakan anggota dari himpunan, 0 memiliki arti bahwa sebuah objek merupakan non-anggota dari himpunan, sedangkan nilai yang berada diantara 0 dan 1 menunjukkan tingkat keanggotaan parsial. Tingkat keanggotaan dari sebuah elemen didefinisikan oleh yang disebut dengan fungsi keanggotaan. Kebanyakan konsep *fuzzy* berasal dari konsep bahasa manusia, yang pada dasarnya tidak jelas atau kabur seperti halnya umur, ada yang mengatakan umur 27 tahun itu muda, tapi juga ada yang mengatakan lumayan muda (Zadeh, The Concept of a Linguistic Variable and its Application to Approximate Reasoning - 1, 1975). Teori *fuzzy set* menyediakan fitur untuk merepresentasikan konsep, variabel, dan aturan linguistik secara efektif, menjadi model alami untuk merepresentasikan pengetahuan pakar manusia. Kunci dari konsep *fuzzy set* adalah bahwa variabel linguistik didefinisikan sebagai variabel yang memiliki nilai dari istilah linguistik, dan secara semantik digambarkan oleh himpunan *fuzzy* (Zadeh, The Concept of a Linguistic Variable and its Application to Approximate Reasoning - 1, 1975; Riza L. S., Bergmeir, Herrera, & Benitez, 2015). Nilai linguistik mengacu pada label untuk merepresentasikan pengetahuan yang memiliki makna dan makna tersebut ditentukan dengan tingkat keanggotaannya, misalnya $a_1 = \text{"muda"}$ dengan derajat keanggotaan $\mu = 0.7$ memiliki arti bahwa variabel a_1 memiliki nilai linguistik yang direpresentasikan dengan label "muda", dan artinya ditentukan oleh derajat dari 0.7 (Zadeh, The Concept of a Linguistic Variable and its Application to Approximate Reasoning - 1, 1975).

2.5. Definisi dan Algoritma Fuzzy Logic

Fuzzy Logic adalah salah satu pengetahuan yang membuat komputer dapat meniru kecerdasan manusia dan teori *fuzzy* berfokus pada pengalaman manusia juga karakteristik dari kemampuan dalam nilai derajat (Zadeh, Fuzzy Sets, 1965; Fang, Huang, & Liu, 2012; Yao, Li, Li, & Wang, 2015). Secara umum, metode ini

diterapkan pada permasalahan yang mengandung unsur seperti ketidakpastian, ketidaktepatan, ataupun *noisy* (Zadeh, Fuzzy Sets, 1965). Pencetus gagasan logika *fuzzy* adalah Prof. L. A. Zadeh tahun 1965 dari California *University* (Fang, Huang, & Liu, 2012). Pada prinsipnya himpunan *fuzzy* adalah perluasan himpunan *crisp*, yakni himpunan yang membagi sekelompok individu ke dalam dua kategori, yaitu sebagai anggota dan bukan anggota (Zadeh, Fuzzy Sets, 1965).

Di dalam algoritma *fuzzy* terdapat *fuzzy statement* yang berisikan nama dari *fuzzy set*, di mana *fuzzy set* adalah sebuah kelas dari objek dengan rangkaian peringkat dari keanggotaan atau biasa disebut derajat keanggotaan (Zadeh, Communication Fuzzy Algorithms, 1968; Fuzzy Sets, 1965; Fang, Huang, & Liu, 2012; Yao, Li, Li, & Wang, 2015). Dari segi algoritmanya, metode *fuzzy* memiliki arsitektur seperti ditampilkan pada Gambar 2.5 dan memiliki tiga proses utama (Zadeh, Communication Fuzzy Algorithms, 1968; Battaini, Casciati, & Faravelli, 1998; Aisjah & Arifin, 2011), diantaranya adalah :



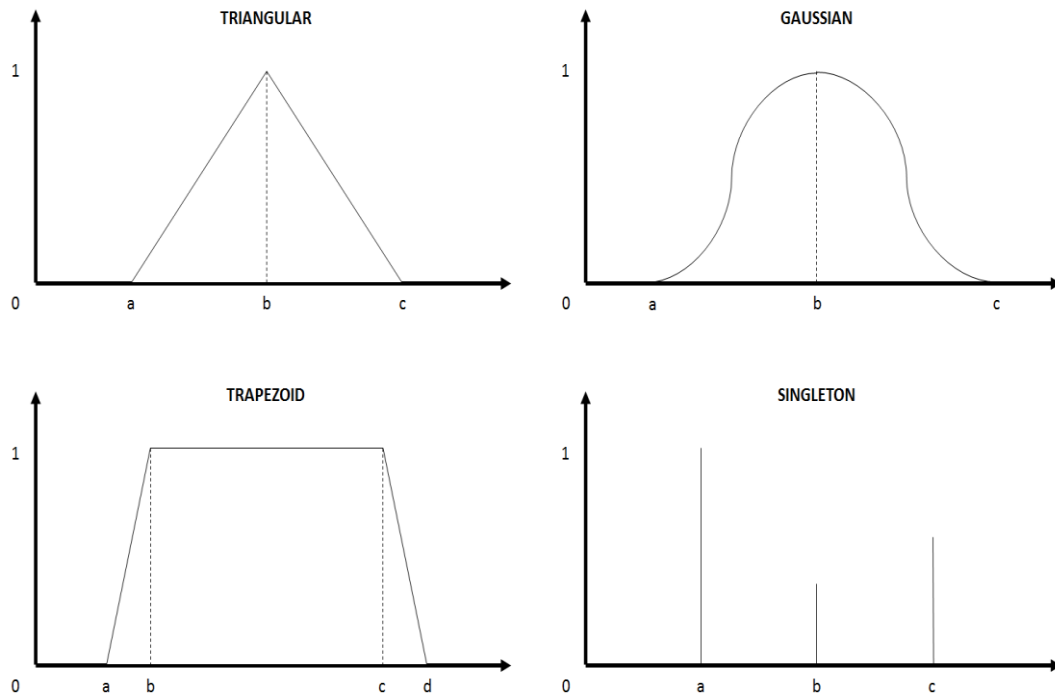
Gambar 2.5 Arsitektur *Fuzzy Logic*, Sumber :

<https://yusronrijal.wordpress.com/2012/03/27/logika-fuzzy/>

2.5.1. *Fuzzification*

Dalam proses ini, data masukan akan diubah dari bentuk tegas (*crisp input*) menjadi *fuzzy* dan biasanya ditampilkan dalam bentuk beberapa himpunan *fuzzy* dengan sebuah fungsi keanggotaannya masing-masing (Zadeh, Fuzzy Sets, 1965;

Battaini, Casciati, & Faravelli, 1998; Chai & Wang, 2011). Fungsi keanggotaan itu sendiri bermacam-macam, beberapa diantaranya ada yang menggunakan kurva segitiga (*triangular*), trapesium (*trapezoid*), *singleton*, dan *Gaussian* seperti ditampilkan pada Gambar 2.6.



Gambar 2.6 Contoh Diagram Fungsi Keanggotaan

Dan tentunya setiap kurva fungsi keanggotaan memiliki bentuk persamaan matematisnya masing-masing, misalnya mengikuti Gambar 2.6, untuk kurva segitiga memiliki bentuk persamaan seperti berikut (Chai & Wang, 2011) :

$$\mu_i[x] = \begin{cases} 1; \rightarrow x = b \\ \frac{x-a}{b-a}; \rightarrow a < x \leq b \\ \frac{c-x}{c-b}; \rightarrow b < x \leq c \\ 0; \rightarrow x > c \mid x < a \end{cases} \quad (2.5)$$

Di mana,

- $\mu[]$ adalah derajat keanggotaan

- i adalah *fuzzy label* (nama label dari variabel x , seperti pada tinggi badan, maka *fuzzy label* nya ada pendek, sedang, sangat tinggi)
- x adalah nilai yang dihitung (*crisp input*) atau dicari derajat keanggotaannya
- a, b , dan c adalah *corner point* (titik sudut) dari fungsi keanggotaannya

Dari persamaan tersebut dapat dijelaskan bahwa, jika " x " sama dengan " b " maka derajat keanggotaannya 1, jika lebih besar dari " b " atau lebih kecil dari " a " maka derajat keanggotaannya 0, jika berada diantara " a " dan " b " maka derajat keanggotaannya adalah " x " dikurangi " a " lalu dibagi hasil pengurangan dari " b " dan " a ", dan terakhir apabila berada diantara " b " dan " c " maka derajat keanggotaannya adalah " c " dikurangi " x " lalu dibagi hasil pengurangan dari " c " dan " b ".

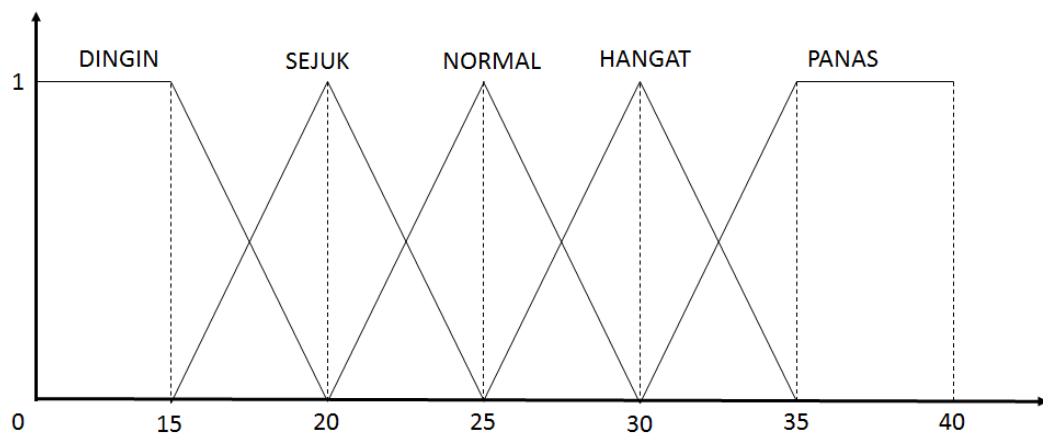
Pseudocode Fuzzification (Triangular)	
Input	: x, a, b, c
Output	: $mDegree$
Proses	:
	<pre> if($x = b$) { $mDegree = 1$ } else if($a < x$ and $x \leq b$) { $mDegree = (x - a) / (b - a)$ } else if($b < x$ and $x \leq c$) { $mDegree = (c - x) / (c - b)$ } else if($a > x$ or $x > c$) { $mDegree = 0$ } return($mDegree$) </pre>

Gambar 2.7 Pseudocode Fuzzifikasi Segitiga

Contoh pseudocode fuzzifikasi segitiga ditampilkan pada Gambar 2.7. Dapat dejasikan bahwa untuk melakukan fuzzifikasi segitiga dibutuhkan masukkan

x , a , b , dan c di mana x merupakan angka input, lalu a , b , lalu c adalah corner point untuk fungsi keanggotaan triangular (a = titik awal, b = titik tengah, dan c = titik akhir). Keluaran dari fungsi ini adalah nilai derajat keanggotaan yang disimpan dalam $mDegree$. Fuzzifikasi segitiga ini terdiri dari proses periksa angka input kepada setiap *corner point* nya (jika angka input lebih besar dari titik awal, jika angka input berada di titik awal dan tengah, jika angka input berada di titik tengah dan akhir, serta jika angka input lebih besar dari titik akhir) dan lakukan hitung derajat keanggotaan menggunakan persamaan fuzzifikasi segitiga.

Sebagai contoh untuk fuzzifikasi, misal temperatur suhu memiliki fungsi keanggotaan seperti ditampilkan pada Gambar 2.8, dengan DINGIN, SEJUK, NORMAL, HANGAT, dan PANAS sebagai *fuzzy label* nya.



Gambar 2.8 Derajat Keanggotaan TEMPERATUR SUHU (Setyaningrum & Swarinata, 2014)

Maka dengan fungsi keanggotaan juga *fuzzy label* DINGIN memiliki nilai antara 0 – 20, SEJUK antara 15 – 25, NORMAL antara 20 – 30, HANGAT antara 25 – 35, dan PANAS antara 30 – 40, didapatkan model sebagai berikut :

- untuk *fuzzy label* DINGIN

$$\mu_{\text{DINGIN}}[x] = \begin{cases} 1; \rightarrow x < 15 \\ \frac{x - 15}{20 - 15}; \rightarrow 15 < x \leq 20 \\ 0; \rightarrow x > 20 \end{cases} \quad (2.6)$$

- untuk *fuzzy label* SEJUK

$$\mu_{\text{SEJUK}}[x] = \begin{cases} 1; \rightarrow x = 20 \\ \frac{x - 15}{20 - 15}; \rightarrow 15 < x \leq 20 \\ \frac{25 - x}{25 - 20}; \rightarrow 20 < x \leq 25 \\ 0; \rightarrow x > 20 \mid x < 15 \end{cases} \quad (2.7)$$

- untuk *fuzzy label* NORMAL

$$\mu_{\text{NORMAL}}[x] = \begin{cases} 1; \rightarrow x = 25 \\ \frac{x - 20}{25 - 20}; \rightarrow 20 < x \leq 25 \\ \frac{30 - x}{30 - 25}; \rightarrow 25 < x \leq 30 \\ 0; \rightarrow x > 25 \mid x < 20 \end{cases} \quad (2.8)$$

- untuk *fuzzy label* HANGAT

$$\mu_{\text{HANGAT}}[x] = \begin{cases} 1; \rightarrow x = 30 \\ \frac{x - 25}{30 - 25}; \rightarrow 25 < x \leq 30 \\ \frac{35 - x}{35 - 30}; \rightarrow 30 < x \leq 35 \\ 0; \rightarrow x > 30 \mid x < 25 \end{cases} \quad (2.9)$$

- untuk *fuzzy label* PANAS

$$\mu_{\text{PANAS}}[x] = \begin{cases} 1; \rightarrow x > 35 \\ \frac{x - 30}{35 - 30}; \rightarrow 30 < x \leq 35 \\ 0; \rightarrow x < 30 \end{cases} \quad (2.10)$$

Diketahui bahwa suhu terdeteksi sebesar 33 °C, lalu berdasarkan fungsi keanggotaan Gambar 2.8, suhu tersebut berada di *fuzzy label* HANGAT dan PANAS, maka hasil fuzzifikasi nya adalah sebagai berikut :

- dalam *fuzzy label* HANGAT

$$\mu_{\text{HANGAT}}[x = 33] = \frac{35 - 33}{35 - 30} = \frac{2}{5} = 0.4$$

- dalam *fuzzy label* PANAS

$$\mu_{\text{PANAS}}[x = 33] = \frac{33 - 30}{35 - 30} = \frac{3}{5} = 0.6$$

Dapat disimpulkan bahwa suhu senilai 33 °C memiliki nilai derajat keanggotaan 0.4 dengan label HANGAT dan 0.6 dengan label PANAS.

2.5.2. *Fuzzification* Wang Mendel

Kebanyakan dalam masalah proses kendali dan sinyal di dunia nyata, memiliki beberapa informasi yang terkait, diantaranya evaluasi, realisasi, desain, dan lain-lain. Informasi-informasi tersebut dapat diklasifikasikan kepada dua jenis , diantaranya:

- a. Informasi numerik didapatkan dari alat ukur
- b. Informasi linguistik didapatkan dari seorang ahli

Kebanyakan pendekatan yang digunakan untuk proses kecerdasan kendali dan sinyal adalah heuristik secara ilmiah, mereka menggabungkan sebuah metode proses kendali atau sinyal biasa dengan sistem pakar secara *ad hoc* untuk permasalahan yang spesifik. Setelah itu simulasi pun dilakukan untuk menunjukkan bahwa pendekatan yang baru bekerja secara lancar untuk permasalahan spesifik. Namun pendekatan seperti ini memiliki dua titik kelemahan :

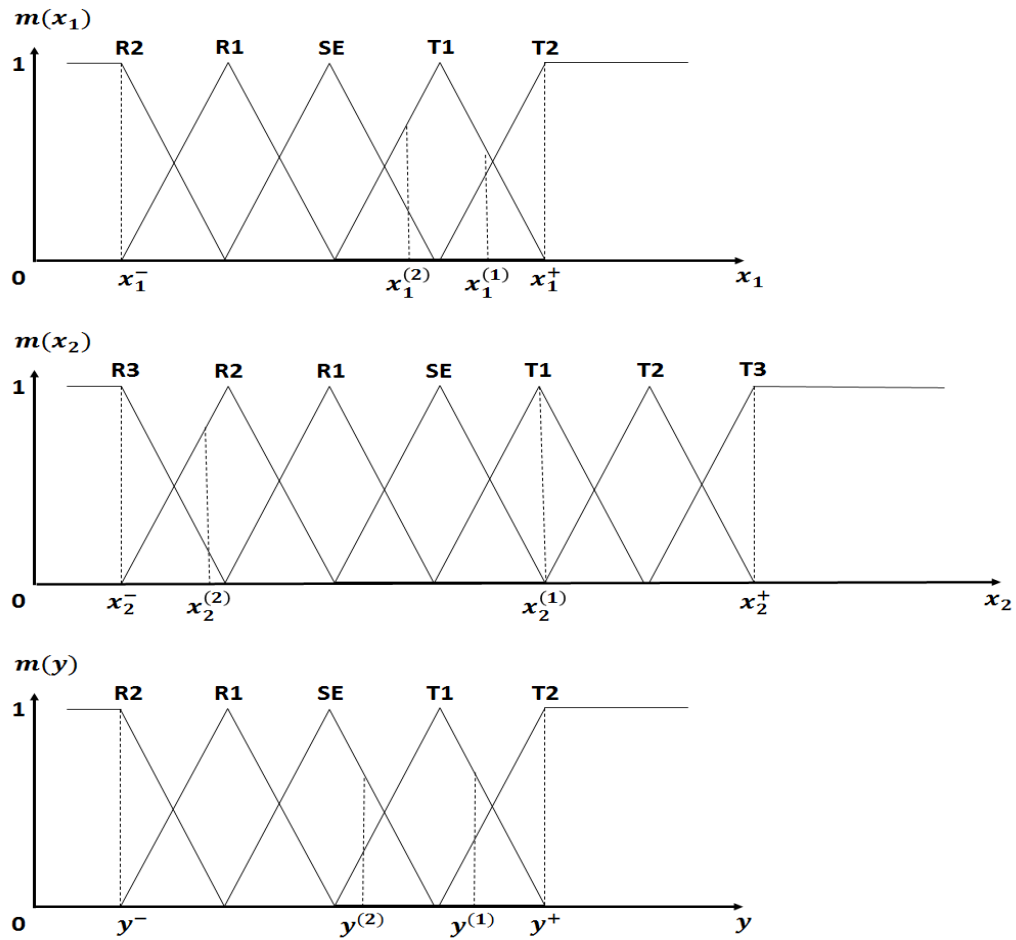
- a. Metode yang dipakai mungkin bekerja untuk satu masalah, tapi belum tentu sesuai untuk masalah yang lain
- b. Belum adanya *framework* untuk memodelkan dan merepresentasikan aspek berbeda dari strategi proses kendali atau sinyal

Dengan metode Wang Mendel ini, informasi numerik dan linguistik dapat digabungkan kedalam *framework* Fuzzy Associative Memory (FAM) Bank (Wang & Mendel, 1991).

Metode Wang Mendel digunakan untuk membangun *fuzzy rule* dari data numerik. Metode ini meliputi lima langkah :

- a. Membagi ruang *input* dan *output* dari data numerik kedalam lingkup *fuzzy*
- b. Membangun *fuzzy rule* dari data yang diberikan
- c. Berikan sebuah derajat pada setiap *rule* yang dihasilkan, ini ditujukan untuk mengatasi konflik diantara *rule* yang dibangun
- d. Membangun kombinasi Fuzzy Associative Memory Bank berdasarkan *rule* yang dihasilkan dan *rule* linguistik dari seorang ahli
- e. Menentukan pemetaan dari ruang *input* ke ruang *output* berdasarkan gabungan FAM Bank menggunakan prosedur defuzzifikasi

Adapun cara membagi ruang *input* dan *output* dari data numerik kedalam lingkup *fuzzy* yaitu, asumsikan interval domain dari x_1 , x_2 dan y adalah $[x_1^-, x_1^+]$, $[x_2^-, x_2^+]$ dan $[y^-, y^+]$, di mana interval domain dari suatu variabel kemungkinan besar berada pada interval yang ditetapkan, dan nilai dari variabel diperbolehkan berada diluar interval domain. Bagi setiap interval domain sebanyak $2N + 1$ daerah, nilai N bisa jadi berbeda untuk variabel yang berbeda, dan rentang dari daerahnya dapat setara atau tidak setara. Denotasikan setiap daerah dengan R_N (Rendah N), ..., R_1 (Rendah 1), SE (Sedang), T_1 (Tinggi 1), ..., T_N (Tinggi N), dan masukkan fungsi keanggotaan di setiap daerah nya.



Gambar 2.9 Hasil Bagi *Input* dan *Output* Menjadi Daerah Fuzzy (Wang & Mendel, 1991)

Variabel x_1 dibagi sebanyak 2 bagian ($N = 2$), x_2 dibagi sebanyak 3 bagian ($N = 3$), dan y dibagi sebanyak 2 bagian ($N = 2$), sehingga x_1 dan y terbagi menjadi 5 daerah derajat keanggotaan, lalu x_2 terbagi menjadi 7 daerah derajat keanggotaan seperti terlihat pada Gambar 2.9.

Pseudocode Generate Membership Function Wang Mendel

Input : dataInput, N

Output : mFunction

Proses :

```

a = max(dataInput)
b = min(dataInput)
c = (2*N)+1

pointCenter = ((a - b)/2) + b

rangePoint = hitungRentang(a, b)
for( i = 1; sebanyak (N*2)){
    accomPoint[i] = accomPoint[i - 1] + rangePoint
}

for( i = 0; sebanyak c){
    if( between(b and pointCenter)){
        mFunction[i] = ["kecil",mShape, accomPoint[]]
    }
    else if(sama dengan pointCenter){
        mFunction[i] = ["sedang",mShape, accomPoint[]]
    }
    else{
        mFunction[i] = ["besar",mShape, accomPoint[]]
    }
}

return(mFunction)

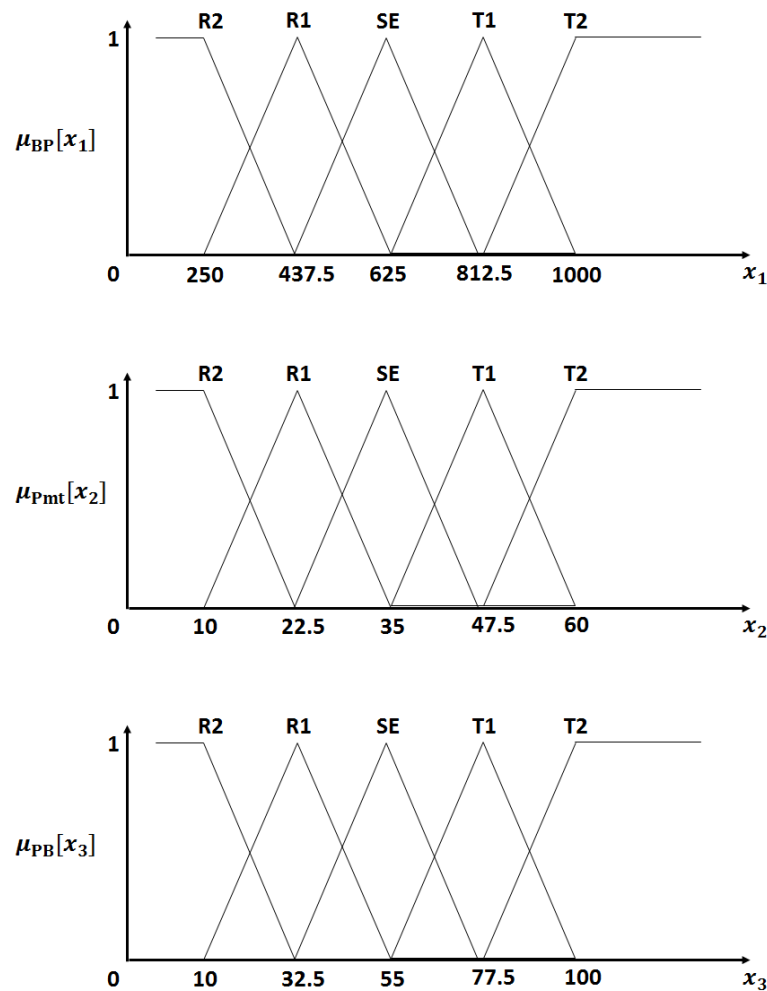
```

Gambar 2.10 Pseudocode Generate Membership Function Wang Mendel

Berdasarkan Gambar 2.10, untuk menghasilkan fungsi keanggotaan menggunakan cara Wang dan Mendel terdiri dari dua masukan yaitu dataInput dan N, di mana dataInput adalah data yang akan digunakan dalam proses *fuzzy* dan N adalah jumlah fungsi keanggotaan yang diinginkan. Keluarannya adalah fungsi keanggotaan yang ditampung kedalam mFunction. Fungsi ini terdiri dari beberapa proses, yang pertama, hitung nilai minimum dan maksimum dari dataInput; kedua, hitung jumlah fungsi keanggotaan yang akan dihasilkan yakni sebanyak $(2*N)+1$; ketiga, hitung titik tengah berdasarkan nilai maksimum dan minimum yang didapatkan; keempat, hasilkan *corner point* mengikuti nilai maksimum, minimum,

dan jumlah fungsi keanggotaan yang ingin dibuat; kelima, beri *fuzzy label* pada setiap fungsi keanggotaan yang dihasilkan dan simpan kedalam mFunction.

Contohnya, misal pada kasus perusahaan *soft drink*, terdapat data dengan variabel biaya produksi, permintaan, dan produksi barang. Masing – masing variabel memiliki nilai minimum dan maksimum nya yakni, nilai minimum untuk biaya produksi adalah 250, permintaan 10, dan produksi barang 10, lalu nilai maksimum untuk biaya produksi adalah 1000, permintaan 60, dan produksi barang 100. Setelah itu bangun fungsi keanggotaan sejumlah 5 keanggotaan untuk masing – masing variabel. Berdasarkan nilai minimum dan maksimum dari masing – masing variabel, fungsi keanggotaan yang terbangun adalah seperti ditampilkan pada Gambar 2.11.



Gambar 2.11 Hasil Proses *Generate Membership Function*

Dari Gambar 2.11 dapat dijelaskan bahwa R2 dan R1 adalah *fuzzy label* untuk rendah ke 2 dan rendah ke 1, lalu SE untuk sedang, terakhir T1 dan T2 untuk tinggi ke 1 dan tinggi ke 2. Lalu x_1 , x_2 , dan x_3 adalah nilai untuk setiap masing – masing variabel x_1 untuk biaya produksi, x_2 permintaan, dan x_3 untuk variabel produksi barang. Dan μ_{BP} merupakan derajat keanggotaan untuk variabel biaya produksi, μ_{Pmt} derajat keanggotaan variabel permintaan, terakhir μ_{PB} derajat keanggotaan variabel produksi barang.

2.5.3. Inference System

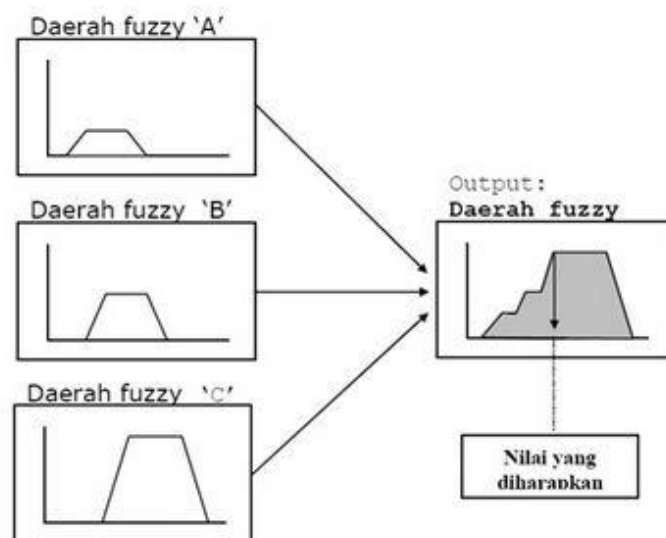
Inference system adalah acuan untuk menjelaskan hubungan antara variabel *input* dan *output* di mana variabel-variabel tersebut berbentuk *fuzzy* (Ruzic, Skenderovic, & Lesic, 2016). Tahap ini merepresentasikan sebuah model matematika formal untuk menajamkan pengetahuan manusia dan kesimpulan ketika data numerik diubah menjadi nilai tak pasti. Ada beberapa model yang dapat digunakan untuk sistem inferensi *fuzzy*, diantaranya ada model Mamdani, Takagi-Sugeno-Kang, dan Tsukamoto (Sanchez-Torrubia & Torres-Blanc, 2010; Saepullah & Wahono, 2015). Proses ini terdiri dari 4 langkah (Zadeh, Communication Fuzzy Algorithms, 1968; Sanchez-Torrubia & Torres-Blanc, 2010), yaitu :

1. Mengevaluasi anteseden (hal yang terjadi dahulu) untuk setiap aturan,
2. Dapatkan satu kesimpulan untuk masing-masing aturan,
3. Agregat seluruh kesimpulan,
4. Terakhir lakukan defuzzifikasi

Salah satu aturan yang sudah banyak digunakan adalah IF..THEN *rules*. Aturan ini merupakan salah satu metode yang sederhana terutama terhadap operasi *min – max* (Zadeh, Communication Fuzzy Algorithms, 1968; Mamdani & Assilian, 1975; Sanchez-Torrubia & Torres-Blanc, 2010). Namun dalam studi kasus ini, sistem inferensi menggunakan metode Bayesian network, sehingga proses agregat kesimpulan dari fuzzifikasi akan dilanjutkan dengan cara menghitung probabilitas menggunakan metode Bayesian network. Dan setelah mendapatkan nilai probabilitas, langkah selanjutnya adalah defuzzifikasi.

2.5.4. Defuzzification

Defuzzifikasi merupakan langkah mengubah nilai *fuzzy* menjadi nilai tegas (*crisp output*) (Zadeh, Communication Fuzzy Algorithms, 1968). Masukkan untuk proses defuzzifikasi adalah sebuah himpunan *fuzzy* yang diperoleh dari aturan – aturan *fuzzy*, dan keluarannya merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Himpunan *fuzzy* untuk proses defuzzifikasi sendiri dapat dilihat seperti pada Gambar 2.12, di mana daerah *fuzzy* ‘A’ adalah himpunan *fuzzy* untuk ‘A’, daerah *fuzzy* ‘B’ merupakan himpunan *fuzzy* untuk ‘B’, dan daerah *fuzzy* ‘C’ adalah himpunan *fuzzy* untuk ‘C’. Metode yang dapat digunakan untuk defuzzifikasi pun banyak, beberapa diantaranya ada metode *mean of maximum* (MOM), *largest of maximum* (LOM), *smallest of maximum* (SOM), *adaptive integration* (AI), *basic defuzzification distributions* (BADD), *constraint decision defuzzification* (CDD), *fuzzy mean* (FM), *bisektor*, dan *random choice of maximum* (RCOM).



Gambar 2.12 Proses Defuzzifikasi, Sumber :

<https://januarwicaksono.wordpress.com/2015/06/>

Center of gravity (COG) merupakan salah satu metode yang biasa digunakan untuk melakukan defuzzifikasi. Metode ini mengambil solusi *crisp* dengan menghitung titik pusat daerah *fuzzy*. Rumus dari metode ini adalah :

$$z = \frac{\sum_{j=1}^n z_j \mu(z_j)}{\sum_{j=1}^n \mu(z_j)} \quad (2.11)$$

Di mana,

- z adalah variabel yang akan di defuzzifikasi
- z_j adalah daerah *fuzzy* ke j dari variabel yang akan di defuzzifikasi
- $\mu(z_j)$ adalah nilai derajat keanggotaan dari tiap daerah *fuzzy*
- \sum adalah summasi
- n adalah jumlah daerah *fuzzy*

Dari persamaan 2. 11 dapat dijelaskan bahwa untuk mendapatkan nilai *crisp* dari sebuah variabel adalah dengan menghitung total luas dibagi dengan total nilai derajat keanggotaannya. Total luasnya sendiri didapatkan dengan melakukan summasi dari perkalian *corner pointnya* (z_j) dan derajat keanggotaannya (μ_j) sedangkan total derajat keanggotaan didapatkan dengan summasi nilai μ mengikuti jumlah *corner point* nya. Untuk lebih jelasnya, dapat dilihat pada contoh kasus.

Pseudocode Defuzzification (COG)

Input : z , cornerPoint, μ , areaF

Output : crispZ

Proses :

```

for(j = 1 sampai areaF){
    a = sum(cornerPoint[z[j]])*mu[z[j]]
}

b = sum(mu[z])

crispZ = a / b

return( crispZ )

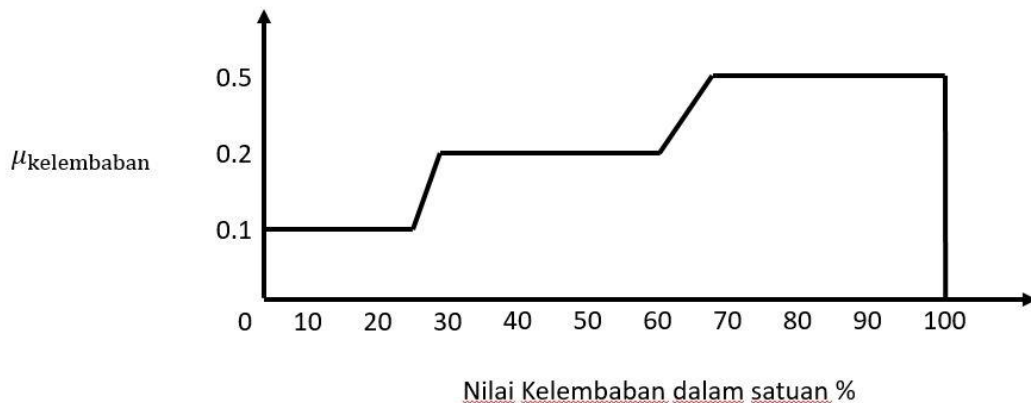
```

Gambar 2.13 Pseudocode Defuzzification COG

Contoh pseudocode untuk defuzzifikasi COG ditampilkan pada Gambar 2.13. Dapat dijelaskan bahwa fungsi defuzzifikasi COG terdiri dari beberapa

masukkan yaitu z , cornerPoint , μ , dan areaF , di mana z adalah variabel yang akan di defuzzifikasi, cornerPoint adalah titik – titik daerah kesimpulan fuzzy, μ adalah nilai derajat keanggotaan, dan areaF merupakan jumlah daerah kesimpulan fuzzy. Keluaran dari fungsi ini adalah nilai *crisp* dari z yang ditampung kedalam variabel crispZ . Prosesnya terdiri dari menghitung total luas daerah kesimpulan fuzzy, menghitung total derajat keanggotaan dari setiap daerah kesimpulan fuzzy, membagi total luas dengan total derajat keanggotaan dan simpan hasilnya ke crispZ .

Sebagai contoh kasusnya, misal BMKG memiliki data cuaca yang terdiri dari variabel curah hujan, lama penyinaran, dan kelembaban. Dari ketiga variabel tersebut, BMKG dimintai untuk menghitung kemungkinan tingkat kelembaban yang akan muncul di esok hari. Metode fuzzy menjadi salah satu cara yang mereka gunakan untuk menghitung kemungkinannya. Melalui proses fuzzifikasi dan sistem inferensi, akhirnya dapatkanlah daerah fuzzy seperti pada Gambar 2.14.



Gambar 2.14 Daerah Fuzzy Kelembaban

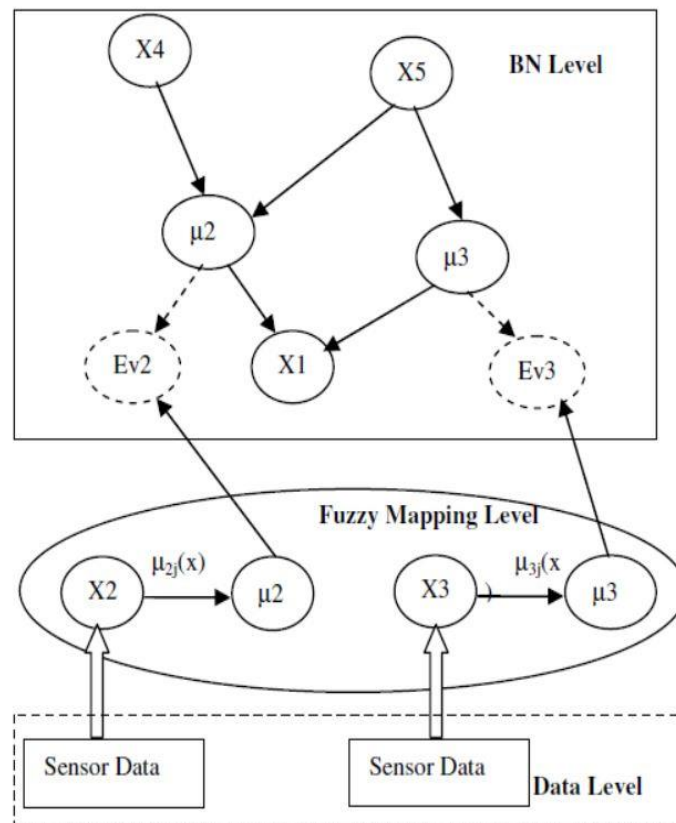
Yakni pada nilai 0 sampai 20 memiliki nilai derajat keanggotaan 0.1, 30 sampai 60 memiliki nilai derajat keanggotaan 0.2, dan dari 70 sampai 100 memiliki nilai 0.5 sebagai derajat keanggotaannya. Selanjutnya metode yang digunakan untuk melakukan defuzzifikasi adalah COG. Maka nilai *crisp* yang didapatkan adalah sebagai berikut :

$$\begin{aligned} \text{COG} &= \frac{(0 + 10 + 20) * 0.1 + (30 + 40 + 50 + 60) * 0.2 + (70 + 80 + 90 + 100) * 0.5}{0.1 + 0.1 + 0.1 + 0.2 + 0.2 + 0.2 + 0.2 + 0.5 + 0.5 + 0.5 + 0.5} \\ &= 67.4 \end{aligned}$$

Berdasarkan hasil perhitungan menggunakan COG, maka kemungkinan tingkat kelembaban di esok hari adalah 67.4 %.

2.5. Fuzzy Bayesian Network

Arsitektur atau hirarki dari gabungan metode *fuzzy* dan *Bayesian network* meliputi 2 lapis, yaitu lapisan atas (Bayesian Network Level) dan lapisan bawah (Fuzzy Mapping Level) seperti yang ditampilkan pada Gambar 2.15 (Chai & Wang, 2011). Lapisan atas digunakan untuk melakukan pencampuran sedangkan bagian bawah digunakan untuk mendiskritkan data kontinu. Bayesian network hanya bekerja pada variabel diskrit (variabel dengan nilai bilangan selalu bulat) sedangkan variabel yang ada dari setiap kasus prediksi tidak seluruhnya dalam bentuk diskrit, sehingga diperlukan proses diskritisasi (Hsu, Huang, & Chang, 2008; Chai & Wang, 2011). Oleh karena itu, salah satu solusinya adalah menggunakan teori *fuzzy* untuk mempartisi nilai dari variabel kontinu (variabel dengan nilai bilangan bulat atau pecahan) ke dalam *fuzzy state*.



Gambar 2.15 Model Proses *Fuzzy Bayesian Network* (Chai & Wang, 2011)

Dalam *fuzzy* Wang Mendel Bayesian Network, proses membangun fungsi keanggotaan dilakukan secara otomatis, sehingga penggunaan metode ini akan lebih efektif dibandingkan dengan metode *fuzzy* yang biasa. Sebab pada *fuzzy* yang biasa, fungsi keanggotaan harus dibangun secara manual dan perlu adanya validasi terhadap pakar. Bentuk persamaan untuk *fuzzy* Wang Mendel Bayesian Network tidak berbeda dengan *fuzzy* Bayesian Network pada umumnya, yakni :

- Apabila variabel yang dihitung probabilitasnya memiliki nilai *fuzzy* sedangkan yang lainnya tidak,

$$P(\beta|A_j) = \sum_{j \in J} \frac{\mu_{\beta}(B_i) P(A_j|B_i) P(B_i)}{P(A_j)} \quad (2.12)$$

Di mana,

- β adalah *fuzzy label* dari variabel B_i
 - B_i adalah variabel yang dihitung probabilitasnya
 - A_j adalah variabel yang menjadi keadaan dari B_i
 - \sum adalah proses summasi
 - $\mu_{\beta}(B_i)$ adalah nilai derajat keanggotaan dari variabel B_i
 - $P()$ adalah nilai probabilitas
- Apabila variabel yang dihitung tidak memiliki nilai *fuzzy* sedangkan yang lainnya punya

$$P(B_i|\alpha) = \sum_{j \in J} \frac{\mu_{\alpha}(A_j) P(A_j|B_i) P(B_i)}{\mu_{\alpha}(A_j) P(A_j)} \quad (2.13)$$

Di mana,

- α adalah *fuzzy label* dari variabel A_j
- B_i adalah variabel yang dihitung probabilitasnya
- A_j adalah variabel yang menjadi keadaan dari B_i
- \sum adalah proses summasi
- $\mu_{\alpha}(A_j)$ adalah nilai derajat keanggotaan dari variabel A_j

- $P()$ adalah nilai probabilitas
- Apabila seluruh variabel memiliki nilai *fuzzy*

$$P(\beta|\alpha) = \sum_{i \in I} \sum_{j \in J} \frac{\mu_{\beta}(B_i) \mu_{\alpha}(A_j) P(A_j|B_i) P(B_i)}{\mu_{\alpha}(A_j) P(A_j)} \quad (2.14)$$

Di mana,

- α adalah *fuzzy label* dari variabel A_j
- B_i adalah variabel yang dihitung probabilitasnya
- A_j adalah variabel yang menjadi keadaan dari B_i
- \sum adalah proses summasi
- $\mu_{\beta}(B_i)$ adalah nilai derajat keanggotaan dari variabel B_i
- $\mu_{\alpha}(A_j)$ adalah nilai derajat keanggotaan dari variabel A_j
- $P()$ adalah nilai probabilitas

Salah satu contoh kasusnya, misal kantor pajak memiliki sebuah data perpajakan yang terdiri dari variabel sensus populasi dan biro pajak. Melalui data tersebut, kantor pajak dimintai untuk menghitung kemungkinan pendapatan tahunan tinggi untuk setiap tahap umur. Gabungan metode *fuzzy* dan Bayesian network mereka gunakan untuk menghitung kemungkinannya. Melalui data perpajakan didapatkan Tabel probabilitas sensus populasi dan biro pajak seperti ditampilkan pada Tabel 2.3 dan 2.4.

Tabel 2.3 Hasil Probabilitas Sensus Populasi (Tang & Liu, 2007)

B_1	B_2	B_3	B_4
0.2	0.25	0.3	0.25

Tabel 2.4 Hasil Probabilitas Biro Pajak (Tang & Liu, 2007)

$P(A_j B_i)$	B_1	B_2	B_3	B_4
A_1	0.1	0.2	0	0
A_2	0.5	0.6	0.8	0
A_3	0.4	0.2	0.2	1

Keterangan Tabel 2.3 dan Tabel 2.4 :

- B_1 adalah Umur Tua
- B_2 adalah Umur Tidak Terlalu Tua
- B_3 adalah Umur Dewasa
- B_4 adalah Umur Remaja
- A_1 adalah Pendapatan Tinggi
- A_2 adalah Pendapatan Sedang
- A_3 adalah Pendapatan Rendah

Lalu dari data perpajakan tersebut didapatkan derajat keanggotaan untuk masing – masing variabelnya seperti ditampilkan pada Tabel 2.5 dan 2.6.

Tabel 2.5 Derajat Keanggotaan Sensus Populasi (Tang & Liu, 2007)

B_1	B_2	B_3	B_4
0.1	0.4	0.6	0.1

Tabel 2.6 Derajat Keanggotaan Biro Pajak (Tang & Liu, 2007)

A_1	A_2	A_3
1	0.5	0

Dengan seluruh data yang ada (dari Tabel 2.5 sampai 2.6), kemungkinan pendapatan tinggi untuk setiap tahap umur bisa dihitung. Namun pertama – tama hitung dahulu total kemungkinan dari pendapatan tinggi, berikut adalah proses perhitungannya,

$$\begin{aligned}
 P(A_1) &= \mu_\alpha(A_1)\mu_\beta(B_1)P(A_1|B_1)P(B_1) + \mu_\alpha(A_1)\mu_\beta(B_2)P(A_1|B_2)P(B_2) \\
 &\quad + \mu_\alpha(A_1)\mu_\beta(B_3)P(A_1|B_3)P(B_3) + \mu_\alpha(A_1)\mu_\beta(B_3)P(A_1|B_3)P(B_3) \\
 &= (1 * 0.1 * 0.1 * 0.2) + (1 * 0.4 * 0.2 * 0.25) + 0 + 0 \\
 &= 0.002 + 0.02 \\
 &= 0.022
 \end{aligned}$$

Setelah mendapatkan nilai total, selanjutnya menghitung kemungkinan pendapatan tinggi pada setiap tahap umur,

- Pendapatan Tinggi dengan Umur Tua

$$\begin{aligned}
 P(B_1|A_1) &= \frac{\mu_\alpha(A_1)\mu_\beta(B_1)P(B_1|A_1)P(B_1)}{P(A_1)} \\
 &= \frac{1 * 0.1 * 0.1 * 0.2}{0.022} \\
 &= \frac{0.002}{0.022} \\
 &= 0.09
 \end{aligned}$$

- Pendapatan Tinggi dengan Umur Tidak Terlalu Tua

$$\begin{aligned}
 P(B_2|A_1) &= \frac{\mu_\alpha(A_1)\mu_\beta(B_2)P(B_2|A_1)P(B_2)}{P(A_1)} \\
 &= \frac{1 * 0.4 * 0.2 * 0.25}{0.022} \\
 &= \frac{0.02}{0.022} \\
 &= 0.9
 \end{aligned}$$

- Pendapatan Tinggi dengan Umur Dewasa

$$\begin{aligned}
 P(B_3|A_1) &= \frac{\mu_\alpha(A_1)\mu_\beta(B_3)P(B_3|A_1)P(B_3)}{P(A_1)} \\
 &= \frac{1 * 0.6 * 0 * 0.3}{0.022} \\
 &= \frac{0}{0.022} \\
 &= 0
 \end{aligned}$$

- Pendapatan Tinggi dengan Umur Tua

$$\begin{aligned}
 P(B_4|A_1) &= \frac{\mu_\alpha(A_1)\mu_\beta(B_4)P(B_4|A_1)P(B_4)}{P(A_1)} \\
 &= \frac{1 * 1 * 0 * 0.25}{0.022} \\
 &= \frac{0}{0.022} \\
 &= 0
 \end{aligned}$$

Berdasarkan hasil perhitungan, maka dapat disimpulkan bahwa kemungkinan pendapatan tinggi ketika umur nya tua adalah 0.09 dan tidak terlalu tua 0.9.

2.6. Bahasa R

R adalah bahasa pemrograman yang digunakan untuk analisis statistika dan grafik (Ihaka & Gentleman, 1997). R dibuat oleh Ross Ihaka dan Robert Gentleman di Universitas Auckland, Selandia Baru. Saat ini bahasa R dikembangkan oleh R Development Core Team (Riza L. S., Bergmeir, Herrera, & Benitez, 2014). Bahasa R menjadi standar *de facto* diantara stastikawan untuk pengembangan perangkat lunak statistika dan dipergunakan secara luas untuk pengembangan perangkat lunak statistika. Bahasa R merupakan bersi (*template*) *open-source* dari bahasa pemrograman S. Bahasa R memiliki kemampuan yang tidak kalah dengan paket program pengolahan data komersial. R memiliki 5 dasar kelas sebuah objek, diantaranya adalah *character*, *numeric (real number)*, *integer*, *complex*, dan *logical*. Jenis paling mendasar dari objek R adalah sebuah vektor. Ada beberapa fungsi yang digunakan untuk membaca data kedalam R :

- *read.table*, *read.csv*, digunakan untuk membaca data table
- *readLines*, untuk membaca garis dari sebuah file teks
- *source*, untuk membaca dalam file kode R
- *dget*, untuk membaca dalam file kode R
- *load*, untuk membaca dalam tempat kerja yang tersimpan
- *unserialize*, untuk membaca satu objek R dalam bentuk biner

Kita juga dapat menuliskan objek R kedalam bentuk file. Ada beberapa fungsi yang digunakan untuk menuliskan data ke dalam file :

- *write.table*, digunakan untuk menuliskan data tabular atau table ke dalam file teks
- *writeLines*, digunakan untuk menuliskan data karakter baris per baris ke dalam sebuah file atau koneksi
- *dump*, digunakan untuk membuang sebuah representasi teks dari objek R yang berjumlah banyak
- *dput*, digunakan untuk mengeluarkan sebuah representasi teks dari sebuah objek R
- *save*, digunakan untuk menyimpan semua nomer dari objek R dalam bentuk biner ke dalam file
- *serialize*, untuk mengubah sebuah objek R ke dalam bentuk biner untuk diubah menjadi sebuah koneksi atau file

Di dalam R terdapat 3 operator yang bisa digunakan untuk mengekstrak *subsets* dari objek R, diantaranya adalah :

- [

Operator ini selalu mengembalikan sebuah objek dari kelas yang sama dengan yang aslinya. Operator ini juga dapat digunakan untuk memilih elemen yang banyak dari sebuah objek.

➤ `[]`

Operator ini digunakan untuk mengekstrak elemen dari sebuah list atau *data frame*. Operator ini hanya dapat mengekstrak satu buah elemen dan kelas dari pengembalian objek tidak akan seharusnya menjadi list atau *data frame*.

➤ `$`

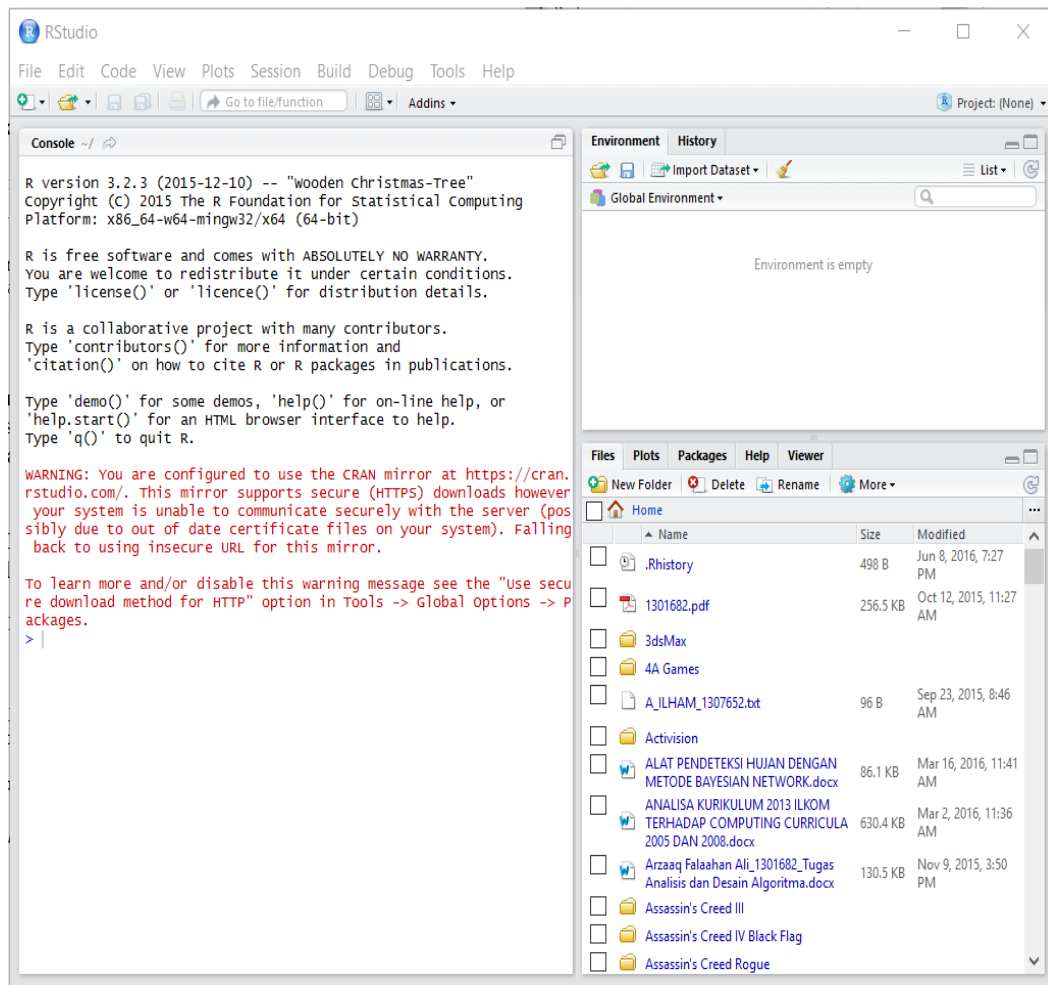
Operator ini dipakai untuk mengekstrak elemen dari sebuah list atau *data frame* dengan nama yang tepat. Operator ini memiliki kesamaan dengan operator `[]`

R *Project* pertama kali dikembangkan oleh Robert Gentleman dan Ross Ihaka yang bekerja di departemen statistic Universitas Auckland tahun 1995. Sejak saat itu program R mendapat sambutan yang luar biasa dari kalangan satistikawan, *industrial engineering*, peneliti, *programmar*, dan lain sebagainya. Saat ini, *source code* kernel R dikembangkan oleh R Core Team yang beranggotaan 17 orang statistisi dari berbagai penjuru dunia.

Program R dapat diunduh secara gratis di <http://cran.r-project.org/> dengan situs resminya di <http://www.r-project.org/> . R pertama kali diluncurkan pada tahun 1997. R memiliki banyak kelebihan, diantaranya:

- R tersedia untuk berbagai sistem operasi
- Memiliki kemampuan membuat graph yang canggih
- Mempunyai sintaks yang mudah dipelajari dengan menciptakan fungsi-fungsi buatan pengguna sendiri
- Memiliki banyak *package* yang dapat diunduh gratis

Namun, kelebihan utama bekerja dengan R adalah pemrograman ini menyediakan ekosistem ilmiah dari *package* yang disumbangkan oleh para peneliti dan praktisi (Riza L. S., Bergmeir, Herrera, & Benitez, 2014). Dan ada lebih dari 5000 *package* tersedia yang terklasifikasi ke lebih dari 20 tampilan tugas. Misalnya, tampilan tugas *Machine Learning* berisikan lebih dari 30 *package* untuk jaringan syaraf tiruan, *random forest*, ataupun *support vector machines*.



Gambar 2.16 RStudio

R Studio merupakan salah satu GUI untuk pemrograman bahasa R. Gambar 2.16, merupakan tampilan dari R Studio. R Studio dapat digunakan apabila program R telah di install. Tab console pada R Studio digunakan untuk menjalankan fungsi-fungsi atau sintaks-sintaks bahasa R.

Meskipun R mengutamakan penggunaan bahasa pemrograman, untuk pengguna awam dengan metode statistik dan bahasa pemrograman, dapat memanfaatkan paket R-commander yang telah disediakan di *library*. Dengan R-commander, pengguna dapat melakukan pengolahan data secara statistik dengan mudah. Hal ini sangat dimungkinkan karena melalui R-commander, pengguna bias langsung melakukan pengolahan data dengan memilih menu-menu yang disediakan pada jendela R-commander.

Rexer's Annual Data Miner Survey 2010 menyebutkan bahwa R telah menjadi alat data mining yang digunakan oleh mayoritas pengguna. Salah satu penyebabnya adalah adanya Rattle, yaitu suatu *library* yang digunakan secara khusus untuk Data Mining melalui GUI (*Graphic User Interface*). Rattle dapat menyajikan ringkasan data secara statistik dan secara visual dari berbagai sumber data seperti Excel, SQL, ataupun XML, yang selanjutnya dapat mentransformasi data ke dalam bentuk yang siap untuk dimodelkan. Untuk pemodelannya dapat digunakan berbagai metode baik *supervised* maupun *unsupervised* dan sekaligus mampu membuat laporan secara grafis untuk menampilkan kerja model yang dibangun.

Pada R prompt kita dapat memasukkan berbagai macam ekspresi. Setiap simbol atau ekspresi memiliki banyak fungsi contohnya simbol “<-”, simbol ini berfungsi sebagai operator penugasan atau *assignment operator* seperti berikut :

```
> x <- 10
> y <- 20
> z <- 30
```

Gambar 2.17 Operator Penugasan - 1

Dari ekspresi yang dimasukkan pada Gambar 2.17, maka nilai x adalah 10, y adalah 20, dan z adalah 30. *Assignment operator* tidak hanya untuk nilai angka saja, nilai *string* pun bisa, contohnya :

```
> p <- "ini"
> q <- "itu"
> r <- "dan"
```

Gambar 2.18 Operator Penugasan - 2

Dari ekspresi yang dimasukkan pada Gambar 2.18, maka p berisikan kata “ini”, q berisi kata “itu”, dan r berisi kata “dan”. Dalam pemrograman R dapat melakukan operasi tambah, kurang, kali, dan bagi. Berikut ini beberapa contoh dari operasi tambah, kurang, kali, dan bagi :

- Pertambahan x dan y yang memiliki nilai x=5 dan y=5

```
> x <- 5
> y <- 5
> x + y
[1] 10
```

Gambar 2.19 Operator Pertambahan

- Mengurangi x dan y yang memiliki nilai x=11 dan y=10

```
> x <- 11
> y <- 10
> x - y
[1] 1
```

Gambar 2.20 Operator Pengurangan

- Membagi x dengan y dengan masing-masing nilainya x=4 dan y=2

```
> x <- 4
> y <- 2
> x / y
[1] 2
```

Gambar 2.21 Operator Pembagian

- Perkalian a dengan b dengan masing-masing nilai a=10 dan b=12

```
> a <- 10
> b <- 12
> a * b
[1] 120
```

Gambar 2.22 Operator Perkalian

Di dalam bahasa R pun kita dapat membuat fungsi. Berikut ini merupakan salah satu contoh proses gerbang logika AND dengan menggunakan perhitungan net yang dibuat dalam bahasa R :

```
# FUNGSI AND
fungsi_and <- function (x1, x2){
  #Inisialisasi Bobot
  w1 = 1
  w2 = 1

  #Menghitung NET
  net = (x1*w1) + (x2*w2)

  #Syarat
  y = NULL
  if(net >= 2){
    y = 1
  }else{
    y = 0
  }

  return(y)
}
```

Gambar 2.23 Fungsi Gerbang Logika AND

Dari fungsi tersebut dapat dijelaskan bahwa :

- fungsi_and memiliki dua parameter masukan yaitu x1 dan x2
- x1 dan x2 memiliki nilai biner 1 atau 0
- Inisialisasi bobot $w1 = 1$ dan $w2 = 1$
- Melakukan perhitungan net dengan rumus “ $net = (x1*w1) + (x2*w2)$ ”
- Membuat variabel kosong y sebagai nilai *output*
- Melakukan syarat sesuai dengan gerbang logika AND, jika nilai $net \geq 2$ maka nilai *output* $y = 1$, jika nilai net tidak ≥ 2 maka nilai *output* $y = 0$
- Keluarkan nilai hasil dengan return (“variabel *output*”)

Pemrograman R juga dapat digunakan untuk perhitungan matriks. Berikut ini adalah contoh membuat matriks di dalam bahasa R :

```
> m <- matrix(1:6, nrow = 2, ncol = 3)
> m
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

Gambar 2.24 Bangun Matriks

Dari ekspresi yang dimasukkan pada Gambar 2.24, maka m berisikan matriks dengan jumlah baris (*nrow*) = 2, kolom (*ncol*) = 3, dan memiliki nilai dari 1 sampai 6 untuk setiap indeks matriks nya. Adapun contoh lainnya :

```

> x <- 1:3
> y <- 10:12
> cbind(x, y)
      x y
[1,] 1 10
[2,] 2 11
[3,] 3 12
> rbind(x, y)
      [,1] [,2] [,3]
x         1     2     3
y        10    11    12

```

Gambar 2.25 Gabung Matriks

Dari ekspresi yang ditampilkan pada Gambar 2.25, dapat dijelaskan bahwa *x* memiliki nilai dari 1 sampai 3 dan *y* memiliki nilai dari 10 sampai 12, lalu hasil keluaran yang pertama adalah matriks yang dibangun dengan menggabungkan kolom berdasarkan nilai *x* dan *y* (*cbind(x,y)*), sedangkan yang terakhir dari penggabungan baris berdasarkan nilai *x* dan *y* (*rbind(x,y)*).

Ada pula fungsi dalam bahasa R yang digunakan untuk membaca file yaitu *read.table("nama_file.format_file")*. Dengan fungsi ini kita dapat melakukan proses data menggunakan fungsi atau program R yang telah dibuat. Contoh :

```

> data <- read.table("tabel.txt")

```

Gambar 2.26 Read File in R

Berdasarkan perintah yang ditampilkan pada Gambar 2.26, maka variabel “data” akan berisi nilai-nilai yang ada dalam file “table.txt”.

Fungsi-fungsi yang kita buat dapat disimpan , format penyimpanan file untuk pemrograman R adalah “nama_file”.R. Setelah program yang kita bangun tersimpan, maka cara untuk menjalankan setiap fungsi yang ada dalam file tersebut adalah dengan perintah *source("direktori_file/nama_file.R")*, contoh :

```

> source('D:/Kampus/Jaringan Syaraf Tiruan/Tugas 2/net.R')

```

Gambar 2.27 Jalankan File R

Dengan menggunakan perintah seperti Gambar 2.27, maka fungsi yang ada dalam file “net.R” dapat digunakan.

BAB III

METODOLOGI PENELITIAN

3.1. Desain Penelitian

Penelitian ini dilakukan melalui beberapa tahapan, diantaranya ada studi literatur, analisis kebutuhan model, perancangan aplikasi, pengujian aplikasi, dan dokumentasi seperti ditampilkan pada Gambar 3.1. Pada tahap studi literatur, penulis mempelajari serta memahami tentang prediksi cuaca, metode *fuzzy logic*, Bayesian network, cara menggunakan kombinasi metode *fuzzy* dan Bayesian network, juga penggunaan bahasa R. Dengan mengumpulkan dan membaca jurnal, buku, serta *website* lah penulis mempelajarinya. Dan tidak cukup dengan membaca, penulis juga mencoba melatih diri dengan mengaplikasikan tahap – tahap penggunaan metode *fuzzy*, Bayesian network, serta pemrograman bahasa R dari jurnal acuan, buku, dan *website*. Latihan tersebut dilakukan supaya lebih mengerti serta paham dengan cara kerja kombinasi metode *fuzzy* dan Bayesian network, terutama untuk mengatasi permasalahan prediksi cuaca.

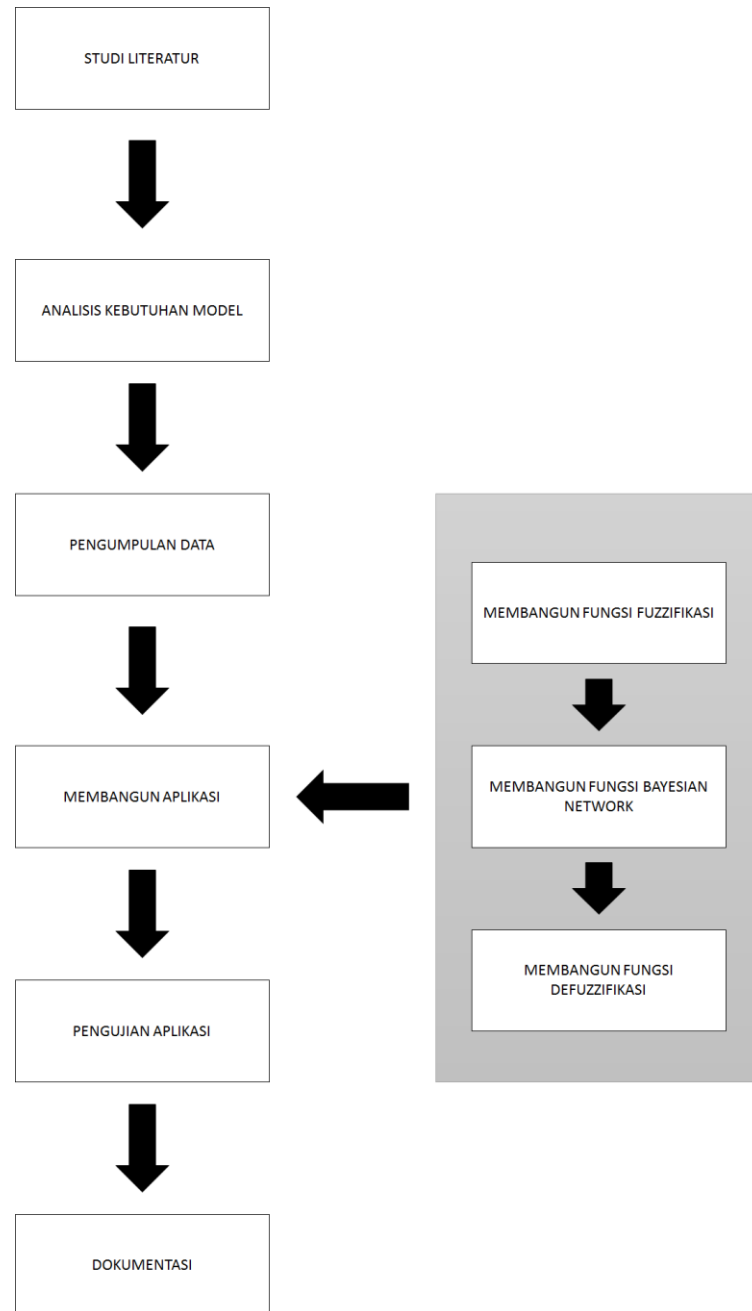
Melalui pembelajaran dan latihan yang telah dilakukan sebelumnya pada tahap studi literatur, selanjutnya adalah menganalisa data – data yang dibutuhkan untuk memprediksi cuaca menggunakan kombinasi metode *fuzzy* dan Bayesian network. Dan berdasarkan hasil analisis metode serta prediksi cuaca yang ditempuh oleh penulis, dalam penelitian ini dibutuhkan data cuaca iklim, *diagram acyclic graph*, dan fungsi keanggotaan.

Setelah mengetahui data – data yang dibutuhkan, langkah selanjutnya adalah mengumpulkan data. Data yang dikumpulkan terdiri dari data cuaca iklim, *diagram acyclic graph*, dan fungsi keanggotaan. Data cuaca iklim didapatkan dari salah satu *website* resmi yang dibangun oleh Badan Meteorologi, Klimatologi dan Geofisika (BMKG) yaitu www.dataonline.bmkg.go.id. Untuk mengambil data di *website* tersebut, kita harus melakukan registrasi terlebih dahulu. Lalu untuk mengunduh data nya sendiri kita dimintai untuk memberikan komentar, kritik, dan saran terlebih dahulu. Data yang diambil adalah data cuaca iklim di Kota Bandung

dari awal bulan Januari 2010 hingga akhir bulan Desember 2016. Untuk detail pengunduhan data cuaca iklim dapat dilihat di lampiran. Setelah data cuaca iklim didapatkan, selanjutnya membangun fungsi keanggotaan untuk metode *fuzzy* dan *diagram acyclic graph* untuk metode Bayesian network. Fungsi keanggotaan dan *diagram acyclic graph* dibangun sendiri, namun harus ada validasi dari ahli atau pakar. Dalam mengatasi masalah prediksi cuaca, penulis melakukan validasi kepada salah satu peneliti cuaca dan iklim di BMKG Bandung yang bernama bapak Muhammad Iid. Fungsi keanggotaan dan *diagram acyclic graph* akan dibahas lebih lanjut di BAB IV.

Setelah pengumpulan data, langkah selanjutnya adalah membangun aplikasi. Dalam membangun aplikasi, dibutuhkan 3 fungsi utama yaitu, fungsi fuzzifikasi, Bayesian network, dan defuzzifikasi. Ketiga fungsi tersebut harus dibangun karena metode *fuzzy* dan Bayesian network memiliki 3 proses utama yakni, diskritisasi variabel kontinu menggunakan fuzzifikasi, menghitung probabilitas menggunakan Bayesian network, dan melakukan penegasan nilai menggunakan defuzzifikasi. Dan dalam pembangunan setiap fungsi terdapat tahap uji coba yang digunakan untuk mengetahui kondisi baik buruknya dari fungsi yang kita bangun hingga akhirnya fungsi tersebut jadi. Setelah seluruh fungsi jadi, tahap selanjutnya adalah melakukan uji aplikasi.

Dalam pengujian aplikasi, penulis menggunakan validasi silang *k-folds*. Pengujian ini ditujukan untuk mengetahui tingkat akurasi, *error* dan kecepatan dari aplikasi yang telah dibangun. Karena lebih kurangnya aplikasi yang dibangun dapat dilihat melalui tingkat akurasi, *error*, dan kecepatan. Dan setelah seluruh hasil setiap tahapan mulai dari studi literatur hingga pengujian aplikasi didapatkan, maka langkah selanjutnya adalah mendokumentasikannya.



Gambar 3.1 Desain Penelitian

Pada Gambar 3.1 dapat dijelaskan bahwa dalam penyusunan penelitian ini terdiri dari beberapa tahapan, dimulai dari studi literatur hingga dokumentasi. Dalam tahap membangun aplikasi dibutuhkan untuk membangun tiga fungsi utama. Dan tiga fungsi utama dalam membangun aplikasi adalah fungsi fuzzifikasi, Bayesian network, serta defuzzifikasi.

3.2. Alat dan Bahan Penelitian

Berikut ini merupakan alat yang digunakan dalam penelitian :

- *Hardware* :
 - *Processor* Intel Core i7-4710MQ CPU @ 2.50GHz
 - RAM 8 GB DDR3
 - Nvidia Geforce GTX 950M VRAM 2 GB
 - *Harddisk* 500 GB
- *Software* :
 - Windows 10 versi 1607
 - RStudio 0.99.902
 - R-3.2.3
 - Microsoft Excel 2016
 - Power Designer 6

Adapun bahan yang penulis gunakan, diantaranya adalah jurnal, buku, dokumentasi di web, dan hasil dari observasi.

3.3. Metode Penelitian

3.3.1. Proses Pengumpulan Data

Dalam peroses ini penulis mencari data dan informasi yang sesuai dan akurat dengan penelitian yang dilakukan, sehingga dapat dijadikan sebagai acuan atau referensi dalam proses penelitian. Adapun teknik yang dilakukan dalam pengumpulan data ini, diantaranya :

a. Eksplorasi dan Studi Literatur

Proses ini dilakukan dengan mempelajari dan memahami metode *fuzzy logic*, Bayesian network, dan bahasa pemrograman R melalui buku, jurnal-jurnal, dan berbagai dokumentasi yang ada di internet.

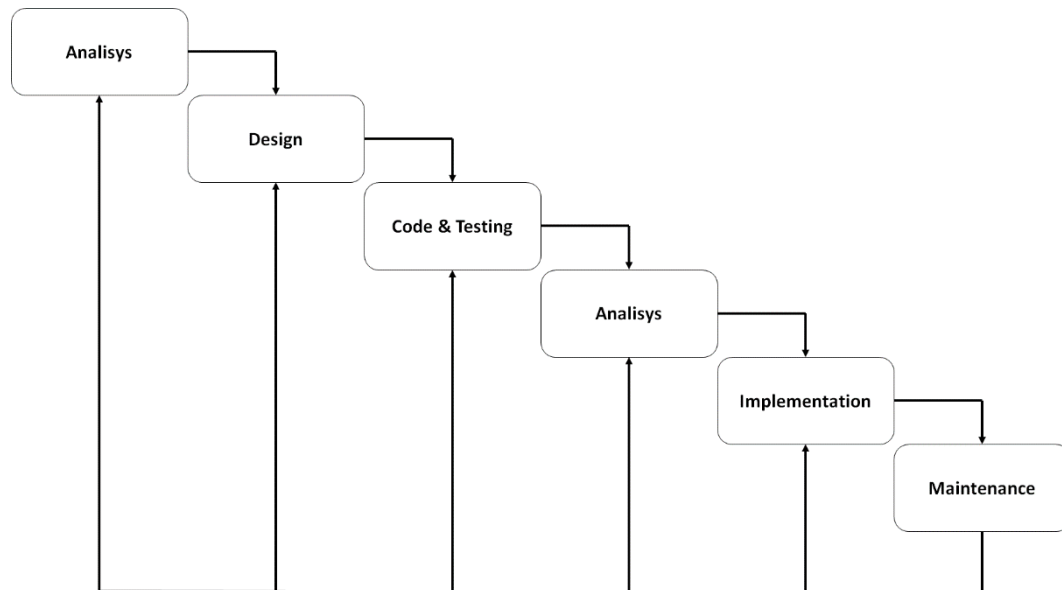
b. Observasi

Melakukan pengumpulan data dari web resmi BMKG dan PTDI Bandung, lalu data dibandingkan sehingga data yang dipakai dalam penelitian lebih

akurat. Data yang diambil beralokasikan di Kota Bandung mulai dari awal bulan Januari 2010 sampai akhir bulan Desember 2016.

3.3.2. Proses Pengembangan Perangkat Lunak

Dalam proses ini penulis menggunakan metode *modern waterfall* untuk pengembangan perangkat lunak. Metode ini memiliki tahap analisis, desain, tes, implementasi, dan *maintenance* seperti yang ditampilkan pada Gambar 3. 2.



Gambar 3.2 Model *Modern Waterfall*

Dari Gambar 3.1. dapat dijelaskan bahwa metode *modern waterfall* memiliki beberapa tahap :

a. Analisis (Analisis)

Mengumpulkan kebutuhan sistem dan melakukan analisis dari awal sampai akhir. Kebutuhan dari sistem ini dimulai dari data cuaca iklim, fungsi keanggotaan, dan DAG.

b. Desain (Design)

Hasil kebutuhan analisis dikonversikan ke dalam rancangan dalam membangun sistem. Hal ini dilakukan dengan merancang tahapan – tahapan dari model *fuzzy* dan Bayesian network kedalam bentuk diagram.

c. Pengkodingan dan Pengujian (Code & Testing)

Menyusun kode program dari hasil analisis dan rancangan sistem, lalu lakukan pengujian dari kode program yang telah dibangun. Kode program disusun menggunakan bahasa R, lalu setiap kali fungsi yang dibangun selesai, selalu dilakukan pengujian untuk menentukan baik tidaknya fungsi yang dibangun.

d. Implementasi (Implementation)

Lakukan implementasi kode program yang telah dibangun terhadap data yang telah dikumpulkan. Hal ini dilakukan untuk menguji kesesuaian kinerja fungsi yang telah dibangun dengan memperhatikan keluaran dari setiap fungsinya.

e. Perawatan (Maintenance)

Kode program yang telah dibangun dapat dikembangkan kembali walaupun pengembangan yang dilakukan sudah mencapai tahap akhir. Dan dengan adanya pengembangan kode program, maka aplikasi yang telah dibangun akan bertambah lebih baik dalam kinerjanya, baik itu dalam hal akurasi ataupun kecepatan.

BAB IV

PENELITIAN DAN PEMBAHASAN

4.1. Pengumpulan Data

Penelitian ini menggunakan data iklim yang diunduh dari dataonline.bmkg.go.id dengan format csv. Data berisikan informasi detail iklim dari 1 Januari 2010 sampai dengan 31 Desember 2016, yakni sebanyak 2500 baris. Data yang digunakan merupakan data iklim harian di Kota Bandung. Data terdiri dari beberapa informasi, diantaranya :

- Tanggal data iklim
- Lokasi data iklim
- Nama stasiun yang melakukan pengumpulan data iklim
- ID stasiun yang melakukan pengumpulan data iklim
- Suhu minimum dengan satuan °C
- Suhu maksimum dengan satuan °C
- Suhu rata-rata dengan satuan °C
- Kelembaban rata-rata dengan satuan %
- Curah hujan dengan satuan millimeter (mm)
- Lama penyinaran dengan satuan jam
- Kecepatan angin rata-rata dengan satuan knot
- Arah angin terbanyak dengan satuan *degree* atau derajat
- Kecepatan angin terbesar dengan satuan knot
- Arah angin saat kecepatan maksimum dengan satuan *degree*

Dari 2500 data yang ditampilkan, terdapat nilai 8888 dan 9999. Nilai 8888 menandakan bahwa data tidak terukur dan 9999 menandakan bahwa data tidak ada, oleh karena itu penulis membuang data-data yang bernilai 8888 dan 9999 menggunakan Microsoft Excel. Akhirnya jumlah data valid yang digunakan adalah 1660 baris.

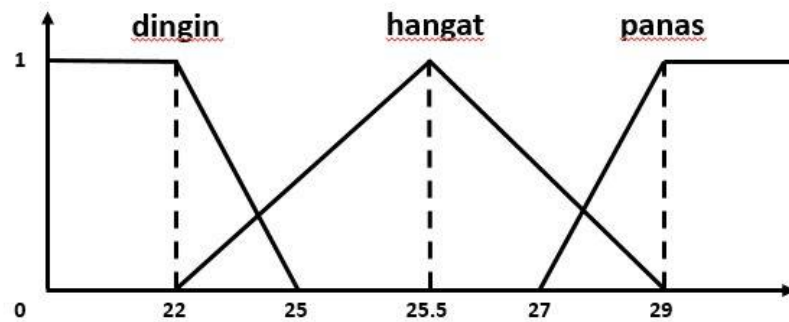
Berdasarkan validasi ke BMKG Bandung yakni dengan pak Muhammad Iid sebagai peneliti cuaca iklim, variabel yang digunakan untuk penelitian ini adalah suhu (suhu rata-rata), kelembaban (kelembaban rata-rata), curah hujan, dan kecepatan angin (kecepatan angin rata-rata). Berdasarkan hasil analisis yang dilakukan menggunakan Microsoft Excel, data ini memiliki nilai-nilai berikut.

Tabel 4.1 Detil Variabel

Nama Variabel	Satuan	Nilai Minimum	Nilai Maksimum	Standar Deviasi	Nilai Rata-rata
Curah Hujan	mm	0	122.9	15.03788	8.428253
Suhu	°C	18.4	26.7	0.8932065	23.48114
Kelembaban	%	52	96	7.448893	78.42349
Kecepatan Angin	knot	1	5	0.5838546	1.777108

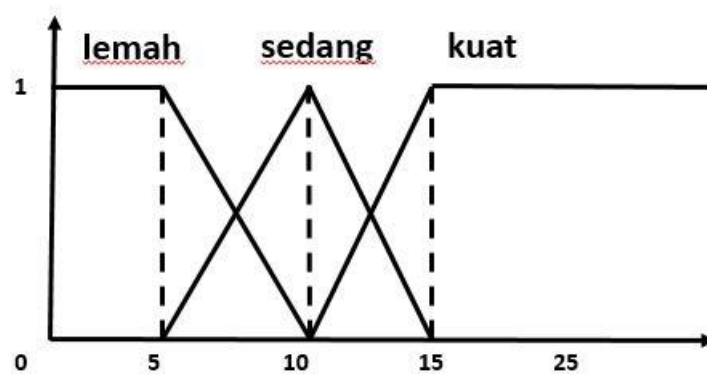
Tabel 4.1 menampilkan satuan, nilai minimum, nilai maksimum, standar deviasi, dan nilai rata – rata dari variabel curah hujan, suhu, kelembaban, dan kecepatan angin. Masing – masing variabel memiliki nilai minimum dan maksimumnya diantaranya, curah hujan memiliki nilai minimum 0 dan maksimum 122.9 dengan satuan mm, suhu memiliki nilai minimum 18.4 dan maksimum 26.7 dengan satuan °C, kelembaban memiliki nilai minimum 52 dan maksimum 96 dengan satuan %, serta kecepatan angin memiliki nilai minimum 1 dan maksimum 5 dengan satuan knot.

Variabel-variabel yang digunakan bersifat kontinu, oleh karena itu dibutuhkan proses fuzzifikasi. Dalam fuzzifikasi dibutuhkan fungsi keanggotaan dari setiap variabelnya dan fungsi yang digunakan untuk menghitung prediksi cuaca ini ditampilkan pada Gambar 4.1, 4.2, 4.3, dan 4.4.



Gambar 4.1 Derajat Keanggotaan **SUHU**

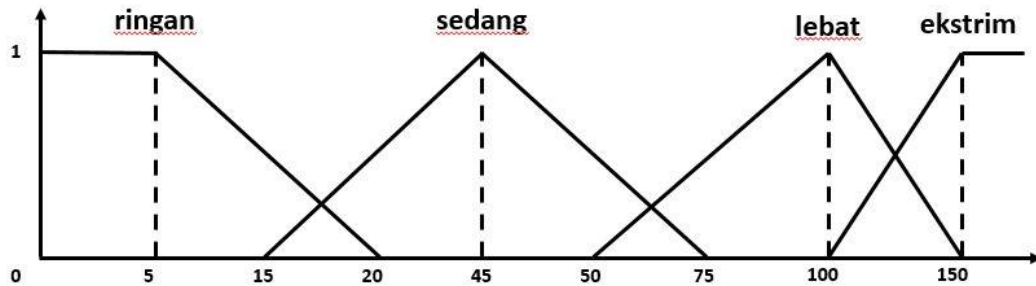
Gambar 4.1 menampilkan diagram derajat keanggotaan untuk variabel SUHU. Diagram ini dibangun berdasarkan pakar cuaca iklim. Dan dapat dijelaskan bahwa variabel ini memiliki tiga *fuzzy label* yaitu “dingin”, “hangat”, dan “panas”. Lalu setiap *fuzzy label* memiliki fungsi keanggotaannya masing – masing, untuk “dingin” menggunakan trapezoid kiri, “hangat” menggunakan segitiga, dan “panas” menggunakan trapezoid kanan. Serta, garis panah dari kiri ke kanan menunjukkan *corner point* untuk SUHU, terdiri dari 0, 22, 25, 25.5, 27, dan 29 dalam satuan $^{\circ}\text{C}$ sedangkan garis panah dari bawah ke atas menampilkan nilai derajat keanggotaan mulai dari 0 hingga 1.



Gambar 4.2 Derajat Keanggotaan **KECEPATAN ANGIN**

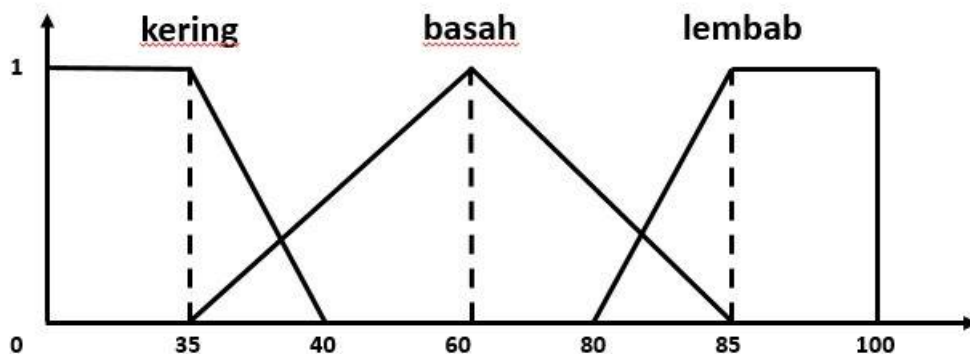
Gambar 4.2 menampilkan diagram derajat keanggotaan untuk variabel KECEPATAN ANGIN. Dapat dijelaskan bahwa variabel ini memiliki tiga *fuzzy label* yaitu “lemah”, “sedang”, dan “kuat”. Lalu setiap *fuzzy label* memiliki fungsi

keanggotaannya masing – masing, untuk “lemah” menggunakan trapezoid kiri, “sedang” menggunakan segitiga, dan “kuat” menggunakan trapezoid kanan. Serta, garis panah dari kiri ke kanan menunjukkan *corner point* untuk KECEPATAN ANGIN, terdiri dari 0, 5, 10, 15, dan 25 dalam satuan knot.



Gambar 4.3 Derajat Keanggotaan **CURAH HUJAN**

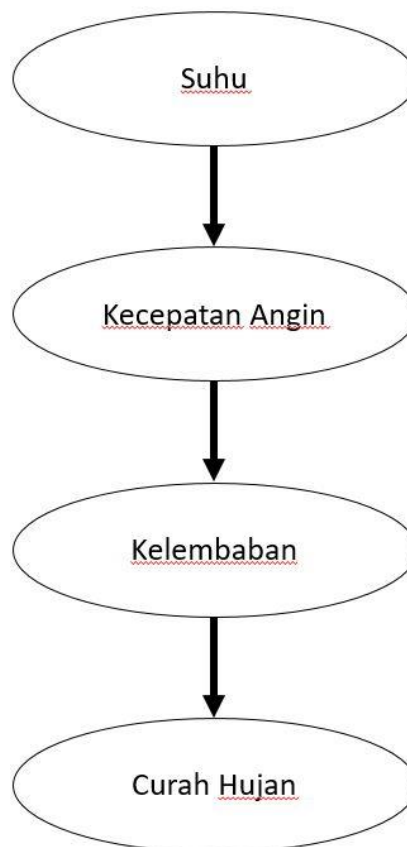
Gambar 4.3 menampilkan diagram derajat keanggotaan untuk variabel CURAH HUJAN. Dapat dijelaskan bahwa variabel ini memiliki empat *fuzzy label* yaitu “ringan”, “sedang”, “lebat”, dan “ekstrim”. Lalu setiap *fuzzy label* memiliki fungsi keanggotaannya masing – masing, untuk “ringan” menggunakan trapezoid kiri, “sedang” dan “lebat” menggunakan segitiga, serta “ekstrim” menggunakan trapezoid kanan. Garis panah dari kiri ke kanan menunjukkan *corner point* untuk KECEPATAN ANGIN, terdiri dari 0, 5, 15, 20, 45, 50, 75, 100, dan 150 dalam satuan mm.



Gambar 4.4 Derajat Keanggotaan **KELEMBABAN**

Gambar 4.4 menampilkan diagram derajat keanggotaan untuk variabel KELEMBABAN. Dapat dijelaskan bahwa variabel ini memiliki tiga *fuzzy label* yaitu “kering”, “basah”, dan “lembab”. Lalu setiap *fuzzy label* memiliki fungsi keanggotaannya masing – masing, untuk “kering” menggunakan trapezoid kiri, “basah” menggunakan segitiga, dan “lembab” menggunakan trapezoid kanan. Garis panah dari kiri ke kanan menunjukkan *corner point* untuk KELEMBABAN, terdiri dari 0, 35, 40, 60, 80, 85, dan 100 dalam satuan %.

Selain derajat keanggotaan, dibutuhkan pula *diagram acyclic graph* (DAG) yang akan dipakai dalam menghitung probabilitas menggunakan Bayesian network. DAG yang digunakan dalam penelitian ini ditampilkan pada Gambar 4.5.



Gambar 4.5 *Diagram Acyclic Graph*

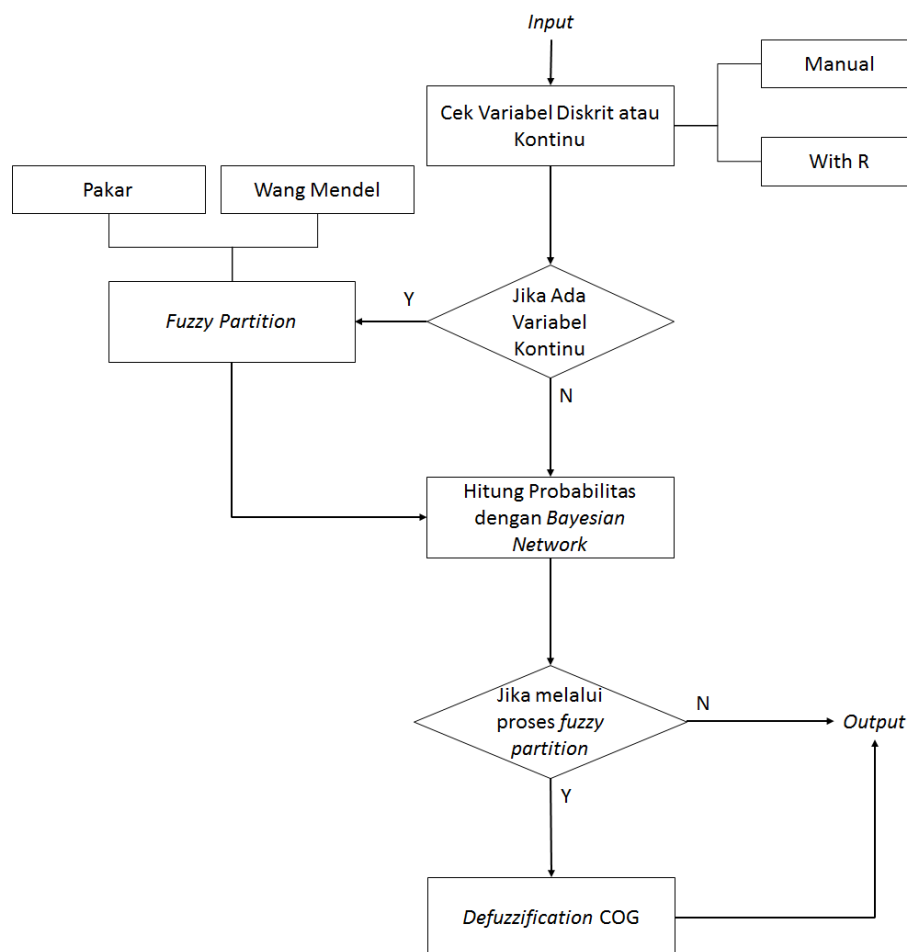
Gambar 4.5 merupakan *diagram acyclic graph* untuk memprediksi cuaca. Diagram ini dibangun berdasarkan pakar cuaca dan iklim di BMKG Bandung. Lalu diagram ini memiliki arti dari setiap panahnya bahwa suhu akan mempengaruhi kecepatan

angin, kecepatan angin mempengaruhi kelembaban, dan kelembaban akan mempengaruhi curah hujan.

Setelah seluruh data yang dibutuhkan terkumpul, langkah selanjutnya adalah merancang model dari kombinasi metode *fuzzy* dan Bayesian network. Perancangan tersebut dibutuhkan untuk mengetahui kebutuhan dari aplikasi yang akan dibangun.

4.2. Perancangan Model Fuzzy dan Bayesian Network

Kombinasi metode *fuzzy* dan Bayesian network ini memiliki beberapa tahapan dalam prosesnya diantaranya, memeriksa variabel diskrit dan kontinu pada data, melakukan fuzzifikasi untuk data kontinu, menghitung probabilitas dari variabel yang ditentukan menggunakan Bayesian network, dan melakukan defuzzifikasi.



Gambar 4.6 Diagram Kombinasi Fuzzy dan Bayesian Network

Seperti yang terlihat pada Gambar 4.6, proses cek diskrit atau kontinu pada data dapat dilakukan dengan manual atau dengan program R, lalu fuzzifikasi dapat dilakukan dengan dua cara yaitu berdasarkan pakar atau Wang Mendel. Apabila berdasarkan pakar, maka fungsi keanggotaan yang dibangun harus ada validasi oleh pakar atau dari pakarnya itu sendiri. Apabila berdasarkan Wang Mendel, maka fungsi keanggotaan akan dihasilkan secara langsung. Lalu seperti yang diketahui bahwa pada metode *fuzzy* terdapat tahap sistem inferensi. Dalam kombinasi metode ini sistem inferensi menggunakan metode Bayesian network yang di dalam nya membutuhkan proses membangun *diagram acyclic graph* (DAG), lalu membangun *coditional probability table* (CPT), dan menghitung probabilitasnya mengikuti CPT. Selanjutnya adalah implementasi kombinasi metode *fuzzy* dan Bayesian network untuk prediksi cuaca.

4.2.1. Implementasi Fuzzy Expert

Dalam metode ini, fuzzifikasi dilakukan berdasarkan pakar atau ahli, lalu karena studi kasus yang diangkat adalah prediksi cuaca, maka untuk mencari tahu keadaan akan hujan atau tidak yaitu dengan menghitung probabilitas curah hujan. Karena fuzzifikasi yang digunakan adalah berdasarkan pakar, maka fungsi keanggotaan pada Gambar 4.1, 4.2, 4.3, dan 4.4 lah yang akan digunakan. Penulis mengambil 5 baris dari data cuaca iklim sebagai contoh, seperti terlihat pada Tabel 4. 2.

Tabel 4.2 Data Iklim Cuaca

Suhu	Kelembaban	Curah Hujan	Kecepatan Angin
23	87	2.4	1
22.1	87	4.2	1
22.9	77	7.2	2
23.3	83	7.5	1
23.3	84	0.5	1

Berdasarkan fungsi keanggotaan dari Gambar 4.1, 4.2, 4.3, dan 4.4, maka nilai derajat keanggotaan untuk masing – masing variabelnya seperti ditampilkan pada Tabel 4.3, 4.4, 4.5, dan 4.6.

Tabel 4.3 Nilai Derajat Keanggotaan Suhu

Suhu		
Dingin	Hangat	Panas
0.67	0.286	0
0.967	0.029	0
0.7	0.257	0
0.567	0.371	0
0.567	0.371	0

Dilihat pada Tabel 4.3, dapat dijelaskan bahwa dengan menggunakan fungsi keanggotaan Suhu yang ditampilkan pada Gambar 4.1, maka nilai derajat keanggotaan untuk data Suhu dari Tabel 4.2 secara berturut – turut adalah 0.67, 0.967, 0.7, 0.567, dan 0.567 pada *fuzzy label* “dingin”, lalu 0.286, 0.029, 0.257, 0.371, dan 0.371 pada *fuzzy label* “hangat”, sedangkan pada *fuzzy label* “panas” nilainya 0 semua. Dan apabila dikaitkan dengan teori probabilitas, nilai derajat keanggotaan menggambarkan tingkat kemungkinan suatu peristiwa akan terjadi, misalnya pada data pertama Suhu dengan *fuzzy label* “Dingin” memiliki nilai derajat yang paling tinggi diantara yang lainnya, maka kemungkinan Suhu adalah “Dingin” sangat besar, lalu dapat diperhatikan pula bahwa nilai derajat pada *fuzzy label* “Panas” bernilai 0 seluruhnya, hal ini menunjukkan bahwa kemungkinan terjadinya Suhu yang “Panas” adalah tidak ada, bisa juga diartikan tidak akan ada Suhu yang “Panas”.

Tabel 4.4 Nilai Derajat Keanggotaan Kelembaban

Kelembaban		
Kering	Basah	Lembab
0	0	1
0	0	1
0	0.32	0
0	0.08	0.6
0	0.04	0.8

Seperti terlihat pada Tabel 4.4, dapat dijelaskan bahwa dengan menggunakan fungsi keanggotaan Kelembaban yang ditampilkan pada Gambar 4.4, maka nilai derajat keanggotaan untuk data Kelembaban dari Tabel 4.2 secara berturut – turut adalah 0, 0, 0.32, 0.08, dan 0.04 pada *fuzzy label* “basah”, lalu 1, 1, 0, 0.6, dan 0.8 pada *fuzzy label* “lembab”, sedangkan pada *fuzzy label* “kering” nilainya 0 semua, dan dapat diperhatikan pada baris pertama serta kedua bahwa Kelembaban dengan *fuzzy label* “Lembab” bernilai 1, ini dapat diartikan bahwa peristiwa munculnya Kelembaban yang “Lembab” akan benar-benar terjadi.

Tabel 4.5 Nilai Derajat Keanggotaan CURAH HUJAN

CURAH HUJAN			
Ringan	Sedang	Lebat	Ekstrim
1	0	0	0
1	0	0	0
0.853	0	0	0
0.833	0	0	0
1	0	0	0

Dilihat pada Tabel 4.5, dapat dijelaskan bahwa dengan menggunakan fungsi keanggotaan CURAH HUJAN yang ditampilkan pada Gambar 4.3, maka nilai derajat keanggotaan untuk data CURAH HUJAN dari Tabel 4.2 secara berturut – turut adalah 1, 1, 0.853, 0.833, dan 1 untuk *fuzzy label* “ringan”, sedangkan pada *fuzzy label* yang lain bernilai 0, nilai 0 ini sendiri memiliki arti bahwa peristiwa munculnya Curah Hujan yang “Sedang”, “Lebat”, dan “Ekstrim” tidak akan terjadi.

Tabel 4.6 Nilai Derajat Keanggotaan Kecepatan Angin

Kecepatan Angin		
Lemah	Sedang	Kuat
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0

Dan pada Tabel 4.6, dapat dijelaskan bahwa dengan menggunakan fungsi keanggotaan Kecepatan Angin yang ditampilkan pada Gambar 4.2, maka nilai derajat keanggotaan untuk data Kecepatan Angin dari Tabel 4.2 adalah satu seluruhnya pada *fuzzy label* “lemah”, sedangkan pada *fuzzy label* yang lain nilainya 0 semua, dan nilai satu disini dapat diartikan bahwa munculnya peristiwa Kecepatan Angin yang “Lemah” benar-benar akan terjadi.

Setelah hitung derajat keanggotaan, bangun data *dummy* untuk membangun *conditional probability table*. Data *dummy* ini akan berisi *fuzzy label* dari masing-masing variabel yang bersifat kontinu dan dibangun berdasarkan nilai derajat keanggotaan tertingginya, misalnya di Tabel 4.3, pada baris pertama, nilai derajat keanggotaan “dingin” lebih besar dibandingkan dengan nilai derajat keanggotaan “hangat” dan “panas”, maka *fuzzy label* yang muncul pada data *dummy* untuk variabel suhu adalah dingin. Dari nilai derajat keanggotaan tertinggi yang

didapatkan (Tabel 4. 3, 4. 4, 4. 5, dan 4. 6), maka data *dummy* yang terbangun untuk 5 baris data cuaca iklim adalah seperti ditampilkan pada Tabel 4.7.

Tabel 4.7 Data *Dummy* Cuaca Iklim

Suhu	Kelembaban	Curah Hujan	Kecepatan Angin
Dingin	Lembab	Ringan	Lemah
Dingin	Lembab	Ringan	Lemah
Dingin	Basah	Ringan	Lemah
Dingin	Lembab	Ringan	Lemah
Dingin	Lembab	Ringan	Lemah

Dapat dijelaskan dari Tabel 4.7 bahwa, berdasarkan nilai derajat keanggotaan tertinggi dari masing – masing variabel yakni Suhu, Kelembaban, Curah Hujan, dan Kecepatan Angin, maka *fuzzy label* yang diambil untuk membangun data *dummy* adalah “dingin” seluruhnya untuk variabel Suhu, “ringan” seluruhnya untuk Curah Hujan, “lemah” untuk variabel Kecepatan Angin, lalu “lembab”, “lembab”, “basah”, “lembab”, dan “lembab” untuk variabel Kelembaban.

```
fuzzyPartition <- function(dataInput, varContinue, rangeData, rangevar, var.mf=NULL, automatic=FALSE,N=1){
  #PEMERIKSAAN VARIABEL KONTINU
  cek_variabel = p_k_var(varContinue, dataInput)

  #JIKA DATA TERSEDIA
  if(cek_variabel == 0){
    #TAMPUNG FUZZY STATE
    fuzzyLabel = list()
    point = list()

    #MENAMPUNG KE DALAM LIST
    #jika automatic
    if(automatic == TRUE){
      var.mf <- automatic.mf(dataInput, varContinue, N)
    }

    out.mf <- list()
    for(i in 1:length(varContinue)){
      out.mf[[varContinue[i]]] = var.mf[[i]]
    }

    l.fs = t_fs(varContinue, var.mf, fuzzyLabel, point)
    point = l.fs[[1]]
    fuzzyLabel = l.fs[[2]]

    #MENGHITUNG DERAJAT KEANGGOTAAN (d_kgt)
    h.fp = h_d_kgt(varContinue, dataInput, fuzzyLabel, point, rangeData)

    #jika ada proses automatic
    h.fp[[3]] = out.mf

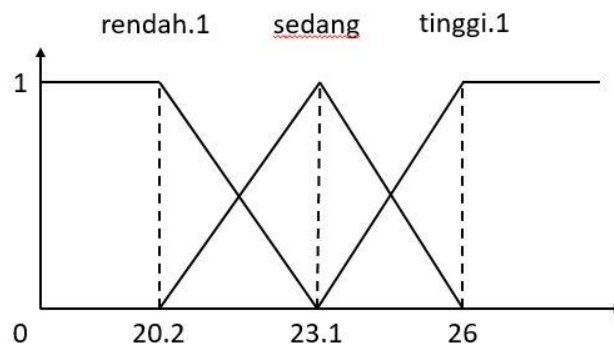
    return(h.fp)
  }
}
```

Gambar 4.7 Source Code Fuzzy Partition

Dapat dilihat pada Gambar 4.7, bahwa *source code* dalam R yang dibangun untuk melakukan fuzzifikasi, baik secara manual (pakar) maupun otomatis (Wang Mendel) diberikan nama *fuzzy partition*. Dan fungsi ini terdiri dari beberapa tahap proses, mulai dari memeriksa ada tidaknya variabel kontinu yang dimasukkan pengguna hingga menampung hasil proses fuzzifikasi.

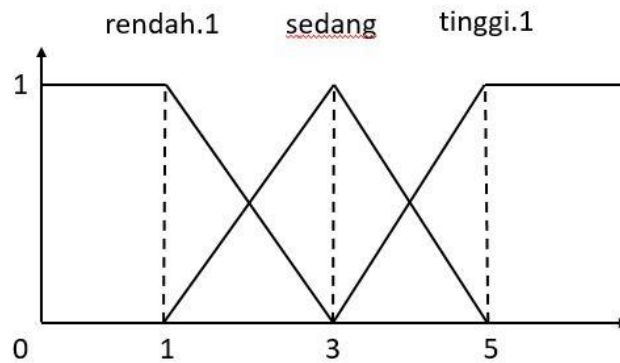
4.2.2. Implementasi Fuzzy Wang Mendel

Dalam metode ini, proses fuzzifikasi mengikuti model Wang dan Mendel, sehingga fungsi keanggotaan dihasilkan secara otomatis. Sebagai contoh kasus, penulis gunakan data pada Tabel 4.2. Lalu untuk proses fuzzifikasi, penulis menggunakan 3 buah fungsi keanggotaan pada setiap variabel kontinu. Dan fungsi keanggotaan yang dihasilkan adalah seperti ditampilkan pada Gambar 4.8, 4.9, 4.10, dan 4.11.



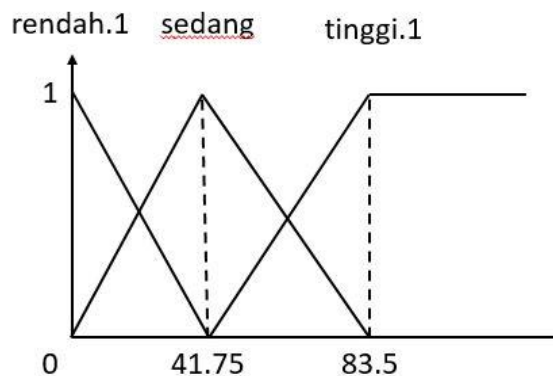
Gambar 4.8 Derajat Keanggotaan Wang Mendel SUHU

Dapat diperhatikan pada Gambar 4.8, variabel SUHU terbagi menjadi tiga buah fungsi keanggotaan dengan menggunakan *fuzzy* Wang Mendel. Yang pertama memiliki *fuzzy label* “rendah.1”, kedua “sedang”, dan ketiga “tinggi.1”. Lalu berkat nilai minimum dan maksimum SUHU dari data yang ada pada Tabel 4.2 adalah 20.2 dan 26, maka dihasilkanlah 3 *corner point* yaitu 20.2, 23.1, dan 26.



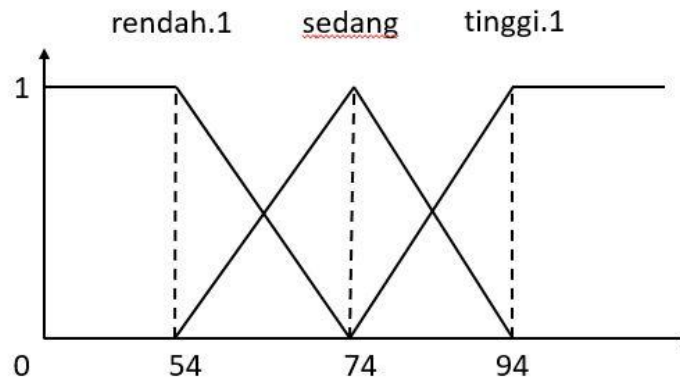
Gambar 4.9 Derajat Keanggotaan Wang Mendel **KECEPATAN ANGIN**

Lalu pada Gambar 4.9, variabel KECEPATAN ANGIN terbagi menjadi tiga buah fungsi keanggotaan juga dengan menggunakan *fuzzy* Wang Mendel. Dengan *fuzzy label* yang sama seperti diagram pada Gambar 4.8. Dan berkat nilai minimum serta maksimum KECEPATAN ANGIN dari data yang ada pada Tabel 4.2 adalah 1 dan 5, maka dihasilkanlah 3 *corner point* yaitu 1, 3, dan 5.



Gambar 4.10 Derajat Keanggotaan Wang Mendel **CURAH HUJAN**

Selanjutnya pada Gambar 4.10, variabel CURAH HUJAN terbagi menjadi tiga buah fungsi keanggotaan juga dengan menggunakan *fuzzy* Wang Mendel. Dengan *fuzzy label* yang sama seperti diagram pada Gambar 4.8. Dan berkat nilai minimum serta maksimum CURAH HUJAN dari data yang ada pada Tabel 4.2 adalah 0 dan 83.5, maka dihasilkanlah 3 *corner point* yaitu 0, 41.75, dan 83.5.



Gambar 4.11 Derajat Keanggotaan Wang Mendel **KELEMBABAN**

Dan pada Gambar 4.11, dapat dijelaskan bahwa variabel KELEMBABAN juga terbagi menjadi tiga buah fungsi keanggotaan karena penggunaan *fuzzy* Wang Mendel. Dengan *fuzzy label* yang sama pula seperti pada Gambar 4.8. Lalu karena nilai minimum serta maksimum KELEMBABAN dari data pada Tabel 4.2 adalah 54 dan 94, maka dihasilkan 3 *corner point* yaitu 54, 74, dan 94.

Setelah fungsi keanggotaan terbangun mulai dari Gambar 4.8 sampai 4.11, langkah selanjutnya adalah melakukan fuzzifikasi pada seluruh variabel. Dan hasil fuzzifikasi dari Tabel 4.2 ditampilkan pada Tabel 4.12, 4.13, 4.14, dan 4.15.

Tabel 4.8 Nilai Derajat Keanggotaan Wang Mendel SUHU

SUHU		
Rendah.1	Sedang	Tinggi.1
0.03	0.97	0
0.34	0.66	0
0.07	0.93	0
0	0.93	0.07
0	0.93	0.07

Tabel 4.8 menunjukkan nilai derajat keanggotaan dari variabel SUHU menggunakan Wang Mendel. Derajat keanggotaan variabel SUHU pada *fuzzy label* “rendah.1” secara berurut adalah 0.03, 0.34, 0.07, dan dua kali muncul 0, lalu dalam

“sedang” adalah 0.97, 0.66, dan tiga kali muncul 0.93, terakhir pada label “tinggi.1” adalah tiga kali muncul 0 dan dua kali muncul 0.07.

Tabel 4.9 Nilai Derajat Keanggotaan Wang Mendel KECEPATAN ANGIN

KECEPATAN ANGIN		
Rendah.1	Sedang	Tinggi.1
1	0	0
1	0	0
0.5	0.5	0
1	0	0
1	0	0

Lalu, Tabel 4.9 menampilkan nilai derajat keanggotaan dari variabel KECEPATAN ANGIN menggunakan Wang Mendel. Derajat keanggotaan variabel KECEPATAN ANGIN pada *fuzzy label* “rendah.1” secara berurut adalah dua kali muncul 1, 0.5, dan dua kali lagi muncul 1, lalu dalam “sedang” adalah dua kali muncul 0, 0.5, dan dua kali lagi muncul 0, terakhir pada label “tinggi.1” adalah 0 seluruhnya.

Tabel 4.10 Nilai Derajat Keanggotaan Wang Mendel CURAH HUJAN

CURAH HUJAN		
Rendah.1	Sedang	Tinggi.1
0.94	0.06	0
0.9	0.1	0
0.83	0.17	0
0.82	0.18	0
0.99	0.01	0

Selanjutnya, Tabel 4.10 menunjukkan nilai derajat keanggotaan dari variabel CURAH HUJAN menggunakan Wang Mendel. Derajat keanggotaan variabel

CURAH HUJAN pada *fuzzy label* “rendah.1” secara berurut adalah 0.94, 0.9, 0.83, 0.82 dan 0.99, lalu dalam “sedang” adalah 0.06, 0.1, 0.17, 0.18 dan 0.01, terakhir pada label “tinggi.1” adalah 0 seluruhnya.

Tabel 4.11 Nilai Derajat Keanggotaan Wang Mendel KELEMBABAN

KELEMBABAN		
Rendah.1	Sedang	Tinggi.1
0	0.35	0.65
0	0.35	0.65
0	0.85	0.15
0	0.55	0.45
0	0.5	0.5

Dan terakhir, Tabel 4.11 menampilkan nilai derajat keanggotaan dari variabel KELEMBABAN menggunakan Wang Mendel. Derajat keanggotaan variabel KELEMBABAN pada *fuzzy label* “rendah.1” adalah 0 semua, lalu dalam “sedang” adalah dua kali muncul 0.35, lalu 0.85, 0.55, dan 0.5, terakhir pada label “tinggi.1” adalah dua kali muncul 0.65, 0.15, 0.45 dan 0.5.

Setelah menghitung nilai derajat keanggotaan, langkah selanjutnya membangun data *dummy* untuk membangun CPT. Berdasarkan nilai derajat keanggotaannya, maka data *dummy* yang terbangun adalah seperti pada Tabel 4.16.

Tabel 4.12 Data *Dummy* Wang Mendel Cuaca Iklim

Suhu	Kelembaban	Curah Hujan	Kecepatan Angin
Sedang	Tinggi.1	Rendah.1	Rendah.1
Sedang	Tinggi.1	Rendah.1	Rendah.1
Sedang	Sedang	Rendah.1	Rendah.1
Sedang	Sedang	Rendah.1	Rendah.1

Sedang	Sedang	Rendah.1	Rendah.1
--------	--------	----------	----------

Dari Tabel 4.12 dapat dijelaskan bahwa berdasarkan nilai derajat keanggotaan tertinggi dari masing – masing variabel mulai dari Suhu, Kelembaban, Curah Hujan, dan Kecepatan Angin, maka didapatkan *fuzzy label* untuk variabel Suhu adalah “sedang” semua, variabel Kelembaban adalah “tinggi.1” sebanyak dua kali dan “sedang” sebanyak tiga kali, serta variabel Curah Hujan dan Kecepatan Angin berisi “rendah.1” seluruhnya.

```
automatic.mf <- function(dataInput, varContinue, N=1){
  #mengambil nilai maksimum dan minimum
  maksimum <- as.matrix(unlist(lapply(dataInput[,varContinue], max)))
  minimum <- as.matrix(unlist(lapply(dataInput[,varContinue], min)))

  #mengambil titik tengah
  rentang = (maksimum - minimum)/2 #rentang
  t.tengah <- minimum + rentang

  #selang antar nilai titik
  if(N > 1){
    for(i in 1:(N-1)){
      rentang = rentang / 2
    }
  }

  #bangun nilai titik
  nilai = NULL
  for (i in 1:nrow(minimum)) {
    nilai = cbind(nilai, seq(minimum[i], maksimum[i], by = rentang[i]))
  }

  #jika hanya satu data kontinu
  if(length(varContinue) == 1){
    row.names(nilai) <- paste(as.character(varContinue))
  }else{
    nilai <- t(nilai) #transpose t()
    row.names(nilai) <- paste(varContinue)
  }

  #bangun membership function (SEGITIGA)
  var.mf = list()

  for(i in 1:length(varContinue)){
    return(var.mf)
  }
}
```

Gambar 4.12 Generate Fuzzy Function

Seperti yang ditampilkan pada Gambar 4.12, fungsi untuk *fuzzy partition* memiliki sub fungsi yang digunakan untuk melakukan *generate* fungsi keanggotaan secara otomatis atau menggunakan cara Wang Mendel. Sub fungsi ini dinamakan *automatic.mf()*, dengan spesifikasi fungsi keanggotaan yang digunakan adalah segitiga. Sub fungsi ini memiliki beberapa tahap proses, diantaranya menghitung

nilai minimum dan maksimum, menentukan nilai tengah, menentukan rentang antar nilai, membangun nilai titik, serta menampung fungsi keanggotaan yang terbentuk dari data.

4.2.3. Implementasi Bayesian Network

Perhitungan probabilitas menggunakan Bayesian network dapat dilakukan apabila seluruh data yang digunakan diskrit atau setelah data *dummy* terbangun. Dalam tahap Bayesian network, setelah seluruh data menjadi diskrit, langkah selanjutnya adalah membuat *conditional probability table* (CPT). Dalam contoh kasus ini, CPT didapatkan dari data training ketika melakukan uji coba di program R, seperti ditampilkan pada Tabel 4.13, 4.14, 4.15 dan 4.16.

Tabel 4.13 CPT CURAH HUJAN

SUHU	KECEPATAN ANGIN	KELEMBABAN	CURAH HUJAN			
			Ringan	Sedang	Lebat	Ekstrim
Dingin	Lemah	Kering	0	0	0	0
Hangat	Lemah	Kering	0	0	0	0
Panas	Lemah	Kering	0	0	0	0
Dingin	Sedang	Kering	0	0	0	0
Hangat	Sedang	Kering	0	0	0	0
Panas	Sedang	Kering	0	0	0	0
Dingin	Kuat	Kering	0	0	0	0
Hangat	Kuat	Kering	0	0	0	0
Panas	Kuat	Kering	0	0	0	0
Dingin	Lemah	Basah	0.93	0.074	0	0
Hangat	Lemah	Basah	0.94	0.052	0.005	0
Panas	Lemah	Basah	0	0	0	0
Dingin	Sedang	Basah	0	0	0	0

Hangat	Sedang	Basah	0	0	0	0
Panas	Sedang	Basah	0	0	0	0
Dingin	Kuat	Basah	0	0	0	0
Hangat	Kuat	Basah	0	0	0	0
Panas	Kuat	Basah	0	0	0	0
Dingin	Lemah	Lembab	0.72	0.25	0.03	0
Hangat	Lemah	Lembab	0.75	0.23	0.02	0
Panas	Lemah	Lembab	0	0	0	0
Dingin	Sedang	Lembab	0	0	0	0
Hangat	Sedang	Lembab	0	0	0	0
Panas	Sedang	Lembab	0	0	0	0
Dingin	Kuat	Lembab	0	0	0	0
Hangat	Kuat	Lembab	0	0	0	0
Panas	Kuat	Lembab	0	0	0	0

Dari Tabel 4.13 dapat dijelaskan bahwa nilai probabilitas dari variabel CURAH HUJAN didapatkan dengan mengikuti keadaan dari SUHU, KECEPATAN ANGIN, dan KELEMBABAN. Salah satunya adalah kemungkinan CURAH HUJAN = Ringan adalah 0.75 ketika SUHU = Hangat, KECEPATAN ANGIN = Lemah, dan KELEMBABAN = Lembab.

Tabel 4.14 CPT KELEMBABAN

SUHU	KECEPATAN ANGIN	KELEMBABAN		
		Kering	Basah	Lembab
Dingin	Lemah	0	0.44	0.56
Hangat	Lemah	0	0.76	0.24
Panas	Lemah	0	0	0

Dingin	Sedang	0	0	0
Hangat	Sedang	0	0	0
Panas	Sedang	0	0	0
Dingin	Kuat	0	0	0
Hangat	Kuat	0	0	0
Panas	Kuat	0	0	0

Seperti ditampilkan pada Tabel 4.14, bahwa CPT untuk variabel KELEMBABAN didapatkan dengan mengikuti keadaan dari SUHU dan KECEPATAN ANGIN. Salah satunya adalah ketika SUHU = Hangat dan KECEPATAN ANGIN = Lemah, kemungkinan KELEMBABAN = Basah sebesar 0.76.

Tabel 4.15. CPT KECEPATAN ANGIN

SUHU	KECEPATAN ANGIN		
	Lemah	Sedang	Kuat
Dingin	1	0	0
Hangat	1	0	0
Panas	0	0	0

Dari Tabel 4.15 dapat disimpulkan bahwa nilai probabilitas dari KECEPATAN ANGIN didapatkan dengan mengikuti keadaan dari SUHU. Dan dapat dijelaskan bahwa kemungkinan KECEPATAN ANGIN = Lemah adalah 1 apabila SUHU = Dingin atau SUHU = Hangat.

Tabel 4.16 CPT SUHU

SUHU		
Dingin	Hangat	Panas
0.58	0.42	0

Lalu pada Tabel 4.16 dapat dijelaskan bahwa nilai kemungkinan SUHU didapatkan langsung melalui jumlah munculnya nilai dari SUHU itu sendiri dibagi dengan total data yang digunakan. Misalnya SUHU = Dingin berjumlah 58, dan karena total data yang digunakan adalah 100, maka kemungkinan SUHU = Dingin adalah 58 dibagi 100 yaitu 0.58.

4.2.4. Implementasi Defuzzifikasi COG

Defuzzifikasi merupakan tahap terakhir dari model *fuzzy* dan Bayesian network. Tahap ini dilakukan apabila data yang digunakan pernah melewati proses fuzzifikasi dan telah melalui proses Bayesian network. Sebagai contoh kasusnya, misal kita akan menghitung kemungkinan curah hujan pada baris pertama di Tabel 4.7 dengan nilai derajat keanggotaan pada Tabel 4.3 sampai 4.6, maka nilai kemungkinan yang didapatkan adalah sebagai berikut :

1. Pada baris pertama data *dummy* diketahui bahwa
 - Suhu = Dingin
 - Kelembaban = Lembab
 - Kecepatan Angin = Lemah
2. Lalu diketahui pula nilai derajat keanggotaan (Tabel 4.3, 4.4, 4.5, dan 4.6) untuk baris pertama adalah
 - Suhu Dingin = 0.67
 - Suhu Hangat = 0.286
 - Suhu Panas = 0
 - Kelembaban Kering = 0
 - Kelembaban Basah = 0
 - Kelembaban Lembab = 1
 - Kecepatan Angin Lemah = 1
 - Kecepatan Angin Sedang = 0
 - Kecepatan Angin Kuat = 0
 - Curah Hujan Ringan = 1
 - Curah Hujan Sedang = 0
 - Curah Hujan Lebat = 0
 - Curah Hujan Ekstrim = 0
3. Asumsikan curah hujan dengan CH, suhu dengan SH, kelembaban dengan KL, dan kecepatan angin dengan KA, maka kemungkinan curah hujan adalah

a. Curah Hujan = Ringan

$$\begin{aligned}
 P(\text{CH}|\text{SH}, \text{KL}, \text{KA}) &= \frac{\mu_{\text{CH}}(\text{CH})\mu_{\text{SH}}(\text{SH})\mu_{\text{KL}}(\text{KL})\mu_{\text{KA}}(\text{KA})P(\text{CH}|\text{SH}, \text{KL}, \text{KA})}{P(\text{SH} = \text{Dingin}, \text{KL} = \text{Lembab}, \text{KA} = \text{Lemah}, \text{CH})} \\
 &= \frac{1 * 0.67 * 1 * 1 * 0.72}{(1 * 0.67 * 1 * 1 * 0.72) + 0 + 0 + 0} \\
 &= \frac{0.4824}{0.4824} \\
 &= 1
 \end{aligned}$$

b. Curah Hujan = Sedang

$$\begin{aligned}
 P(\text{CH}|\text{SH}, \text{KL}, \text{KA}) &= \frac{\mu_{\text{CH}}(\text{CH})\mu_{\text{SH}}(\text{SH})\mu_{\text{KL}}(\text{KL})\mu_{\text{KA}}(\text{KA})P(\text{CH}|\text{SH}, \text{KL}, \text{KA})}{P(\text{SH} = \text{Dingin}, \text{KL} = \text{Lembab}, \text{KA} = \text{Lemah}, \text{CH})} \\
 &= \frac{0 * 0.67 * 1 * 1 * 0.25}{(1 * 0.67 * 1 * 1 * 0.72) + 0 + 0 + 0} \\
 &= \frac{0}{0.4824} \\
 &= 0
 \end{aligned}$$

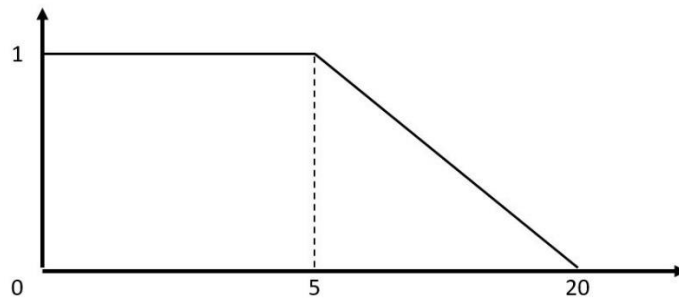
c. Curah Hujan = Lebat

$$\begin{aligned}
 P(\text{CH}|\text{SH}, \text{KL}, \text{KA}) &= \frac{\mu_{\text{CH}}(\text{CH})\mu_{\text{SH}}(\text{SH})\mu_{\text{KL}}(\text{KL})\mu_{\text{KA}}(\text{KA})P(\text{CH}|\text{SH}, \text{KL}, \text{KA})}{P(\text{SH} = \text{Dingin}, \text{KL} = \text{Lembab}, \text{KA} = \text{Lemah}, \text{CH})} \\
 &= \frac{0 * 0.67 * 1 * 1 * 0.03}{(1 * 0.67 * 1 * 1 * 0.72) + 0 + 0 + 0} \\
 &= \frac{0}{0.4824} \\
 &= 0
 \end{aligned}$$

d. Curah Hujan = Ekstrim

$$\begin{aligned}
 P(\text{CH}|\text{SH}, \text{KL}, \text{KA}) &= \frac{\mu_{\text{CH}}(\text{CH})\mu_{\text{SH}}(\text{SH})\mu_{\text{KL}}(\text{KL})\mu_{\text{KA}}(\text{KA})P(\text{CH}|\text{SH}, \text{KL}, \text{KA})}{P(\text{SH} = \text{Dingin}, \text{KL} = \text{Lembab}, \text{KA} = \text{Lemah}, \text{CH})} \\
 &= \frac{0 * 0.67 * 1 * 1 * 0}{(1 * 0.67 * 1 * 1 * 0.72) + 0 + 0 + 0} \\
 &= \frac{0}{0.4824} \\
 &= 0
 \end{aligned}$$

Berdasarkan hasilnya, kemungkinan curah hujan ringan dengan kondisi suhu dingin, lembab, dan kecepatan angin lemah adalah 1. Namun nilai kemungkinan ini masih dalam bentuk *fuzzy*, oleh karena itu dibutuhkan proses defuzzifikasi supaya nilai ini menjadi tegas. Dalam penelitian ini, penulis menggunakan metode COG untuk melakukan defuzzifikasi. Karena curah hujan bernilai 1 pada *fuzzy label* ringan, maka daerah kesimpulannya adalah seperti ditampilkan pada Gambar 4.7.



Gambar 4.13 Daerah Kesimpulan Curah Hujan

Dari Gambar 4.13 dapat dijelaskan bahwa daerah kesimpulan fuzzy untuk Curah Hujan adalah hanya mengambil daerah *fuzzy label* “ringan” yang ditampilkan pada Gambar 4.3. Hal tersebut dikarenakan nilai kemungkinan Curah Hujan = Ringan adalah 1 pada *fuzzy label* “ringan”. Lalu dengan mengikuti daerah kesimpulan *fuzzy* tersebut, maka nilai *crisp* untuk curah hujan nya adalah,

$$\begin{aligned} \text{COG} &= \frac{(0 + 0 + 5 + 20) * 1}{1 + 1 + 1 + 0} \\ &= \frac{25}{3} \\ &= 8.33 \end{aligned}$$

Berdasarkan hasil perhitungan tersebut dapat disimpulkan bahwa nilai curah hujan untuk data cuaca iklim yang pertama adalah 8.33 mm.

```
defuzz <- function(probCountVar, probTable, rangeData, countVar.MF=h.fp[[3]]){
  #ambil seluruh nilai probabilitas
  getProb = probTable[[probCountVar]]

  #ambil membership function
  get.MF = countVar.MF[[probCountVar]]
  varLevel = get.MF[1,]

  #proses defuzzifikasi
  resultDeffuz = matrix(nrow = rangeData)
  colnames(resultDeffuz) = paste(probCountVar)
  for(i in 1:rangeData){
    r.MF = list() #tampung bangun ulang fungsi keanggotaan
    i.MF = 1 #indeks untuk menampung bangun ulang fungsi keanggotaan

    for (j in 1:length(varLevel)) {
      n.Level = as.character(varLevel[j]) #ambil level ke - j
      typeof.MF = get.MF[2,j] #ambil jenis membership function
      muvar = as.numeric(getProb[i,n.Level]) #ambil nilai derajat keanggotaan ke-i
      cornerPoint = get.MF[3:6,j] #ambil corner point

      if (muvar > 0 && muvar < 1) {
        r.MF[[i.MF]] = c(muvar,rCreateMF(typeof.MF, cornerPoint, muvar))
        i.MF = i.MF + 1
      }else if(muvar == 1){
        resultDeffuz[i] = cog(NULL, c(n.Level,typeof.MF,muvar,cornerPoint))
      }
    }

    #jika derajat keanggotaan bukan 1
    if(length(r.MF)!=0){
      resultDeffuz[i] = cog(r.MF, NULL)
    }
  }

  resultCrisp = as.data.frame(resultDeffuz)
  resultCrisp[is.na(resultDeffuz)] = 0
  return(resultCrisp)
}
```

Gambar 4.14 Source Code Defuzzification COG

Pada Gambar 4.14 dapat dijelaskan bahwa fungsi defuzzifikasi yang bernama `defuzz()` ini terdiri dari beberapa tahapan proses. Dimulai dari mengambil nilai probabilitas, mengambil fungsi keanggotaan dari variabel yang akan didefuzzifikasi, hingga akhirnya menghitung dan menampung nilai *crisp* dari variabel yang didefuzzifikasi.

4.3. Model Fuzzy dan Bayesian Network Di R

Aplikasi ini dirancang dengan tiga fungsi utama yaitu, fuzzifikasi, Bayesian network, dan defuzzifikasi. Dan cara kerjanya adalah pertama – tama periksa variabel apa saja pada data yang bersifat kontinu atau diskrit. Apabila pada data tersebut tidak terdapat variabel bersifat kontinu, maka data dihitung langsung probabilitasnya menggunakan fungsi Bayesian network. Namun apabila data yang dimasukkan memiliki variabel kontinu, maka data tersebut harus didiskritkan menggunakan fungsi fuzzifikasi (*fuzzy partition*) dan barulah hitung probabilitasnya dengan fungsi Bayesian network. Lalu apabila data yang dihitung probabilitasnya telah melewati proses fuzzifikasi, maka harus dilakukan penegasan nilai melalui proses defuzzifikasi.

Setiap fungsi yang ada, penulis bangun menggunakan pemrograman R. Adapun detil dari setiap fungsi yang dibangun, mulai dari nama sampai dengan deskripsi dari fungsinya, yaitu :

4.3.1. Fungsi Cek Diskrit Kontinu

Nama Fungsi

`checkDiscreteContinue()`

Input

<code>dataInput</code>	berisikan data masukkan berbentuk matriks
<code>rangeVar</code>	berisikan jumlah variabel yang ada dari data masukkan bentuknya <i>integer</i>

Output

`saveVarContinue` digunakan untuk menyimpan variabel – variabel yang bersifat kontinu dari data masukkan bentuknya *character*

Proses

1. Mengambil seluruh nama variabel dari data
2. Periksa variabel yang bersifat kontinu menggunakan fungsi “`lapply()`” dan “`is.factor()`”
3. Tampung setiap variabel yang bersifat kontinu
4. Tampilkan setiap variabel yang bersifat kontinu

Deskripsi Fungsi

Fungsi ini digunakan untuk mengambil variabel bernilai kontinu dan bersifat opsional, dalam artian boleh digunakan atau tidak digunakan juga tidak apa – apa. Karena ada kemungkinan dengan fungsi ini variabel yang seharusnya bersifat kontinu terbaca sebagai variabel diskrit, misalnya dalam sebuah data terdapat variabel JARAK yang nilainya hanya terdiri dari 55, 50, dan 20, karena hanya ada 3 angka pada variabel tersebut, fungsi ini membacanya sebagai variabel diskrit padahal sebenarnya termasuk variabel kontinu. Maka dari itu, sebagai solusinya, pengguna dapat menentukan langsung variabel kontinu nya. Berikut ini cara menggunakannya :

- Apabila menggunakan fungsi
`varContinue <- checkDiscreteContinue(dataInput, rangeVar)`
- Apabila tidak menggunakan fungsi dan variabel kontinu nya banyak
`varContinue <- c(“nama variabel ke 1”, “nama variabel ke n”)`
- Apabila tidak menggunakan fungsi dan variabel kontinu nya hanya satu
`varContinue <- “nama variabel”`

Lalu nama variabel yang dimasukkan apabila tidak menggunakan fungsi, harus sama persis dengan yang ditampilkan oleh R, karena jika tidak sama, proses diskritisasi tidak bisa dilakukan.

4.3.2. Fungsi Fuzzifikasi

Nama Fungsi

fuzzyPartition()

Input

dataInput	berisikan data masukan berbentuk matriks
varContinue	berisikan variabel – variabel bersifat kontinu berbentuk <i>character</i>
rangeData	jumlah baris dari data masukan bentuknya <i>integer</i>
rangeVar	berisikan jumlah variabel yang ada dari data masukan bentuknya <i>integer</i>
var.mf	berisikan fungsi keanggotaan dari seluruh variabel kontinu dan berbentuk <i>list of matrix</i> , matriks nya berukuran $(n * 6)$, di mana “n” adalah jumlah dari fungsi keanggotaannya, lalu pada baris pertama matriks ini berisikan <i>fuzzy label</i> , kedua bentuk diagram fungsi keanggotaan, dan ketiga sampai ke enam merupakan <i>corner point</i> untuk fungsi keanggotaannya, serta nilai <i>default</i> untuk variabel ini adalah NULL
automatic	variabel ini digunakan supaya fuzzifikasi dapat dilakukan dengan otomatis, bentuknya adalah <i>Boolean</i> antara TRUE atau FALSE

N variabel ini digunakan untuk menentukan jumlah fungsi keanggotaan apabila fuzzifikasi dilakukan otomatis, bentuknya *integer* dan nilai *default* nya 1

Output

h.fp berisikan hasil fuzzifikasi, terdiri dari tiga matriks yang digabung dalam *list*, di *list* yang pertama berisikan matriks derajat keanggotaan, kedua *fuzzy label* dengan derajat keanggotaan tertinggi, dan ketiga berisikan fungsi keanggotaan dari masing – masing variabel

Proses

1. Periksa ketersediaan variabel kontinu yang dimasukkan
2. Bangun fungsi keanggotaan jika menjalankan *automatic* = TRUE
3. Menyesuaikan variabel kontinu dengan fungsi keanggotaannya
4. Menghitung derajat keanggotaan dan bangun matriks *fuzzy label* yang nilai derajat keanggotaannya paling tinggi
5. Tampung derajat keanggotaan, matriks *fuzzy label*, dan fungsi keanggotaan

Deskripsi Fungsi

Fungsi ini digunakan untuk mendiskritkan variabel kontinu dan dapat dijalankan secara otomatis serta manual. Jika otomatis, fungsi keanggotaan tidak perlu diketikkan. Lalu untuk menentukan diagram fungsi keanggotaannya adalah dengan memasukkan angka, jika 1 maka SEGITIGA, 2 untuk TRAPEZOID TENGAH, 3 untuk TRAPEZOID KANAN, dan 4 untuk TRAPEZOID KIRI. Berikut ini adalah contoh menggunakan fungsi *fuzzy partition* :

- *Fuzzy Partition Manual*

```

A.mf <- matrix(c("dingin", "hangat", "panas", 4, 1, 3, 0, 22, 27, 0,
25.5, 29, 22, 35, 25, "-", 35), nrow = 6, byrow = TRUE)
B.mf <- matrix(c("kering", "basah", "lembab", 4, 1, 3, 0, 35, 80, 0,
60, 85, 35, 85, 100, 40, "-", 100), nrow = 6, byrow = TRUE)
var.mf <- list(A.mf, B.mf)
h.fp <- fuzzyPartition(dataInput, varContinue, rangeData, rangeVar,
var.mf)

```

- *Fuzzy Partition* Otomatis

```

h.fp <- fuzzyPartition(dataInput, varContinue, rangeData, rangeVar,
var.mf = NULL, automatic = TRUE, N = 2)

```

Dalam menggunakan *fuzzy partition* manual, menempatkan fungsi keanggotaan kedalam var.mf harus sesuai dengan urutan variabel kontinu yang dibangun atau muncul dari fungsi checkDiscreteContinue, karena jika tidak berurutan, fungsi keanggotaan jadi tidak sesuai dengan variabelnya masing – masing.

4.3.3. Fungsi Bayesian Network

Nama Fungsi

```
bayesianNetwork()
```

Input

dag	merupakan <i>diagram acyclic graph</i> yang akan digunakan untuk proses hitung dalam fungsi bayesianNetwork, bentuknya adalah matriks (2 * n), di mana “n” menunjukkan jumlah relasi antar node di DAG, lalu kolom pertama matriks menunjukkan node asal dan kolom kedua node tujuan
dataInput	berisikan data masukan berbentuk matriks
dataTraining	berisikan data latih berbentuk matriks

probCountVar	berisikan nama variabel yang akan dihitung probabilitasnya, bentuknya <i>character</i>
rangeData	jumlah baris dari data masukkan bentuknya <i>integer</i>
rangeVar	berisikan jumlah variabel yang ada dari data masukkan bentuknya <i>integer</i>
rangeDataTraining	berisi jumlah data latih yang akan digunakan untuk menghitung probabilitas
h.fp	keluaran fungsi <i>fuzzy partition</i> untuk data uji, nilai <i>default</i> untuk variabel ini adalah NULL
h.fpTrain	keluaran fungsi <i>fuzzy partition</i> untuk data latih, nilai <i>default</i> untuk variabel ini adalah NULL

Output

save.result	berisikan nilai-nilai probabilitas dari probCountVar, bentuknya adalah <i>list</i> dari satu matriks, dan isi matriks tersebut merupakan nilai probabilitasnya
-------------	--

Proses

1. Tampung nama variabel
2. Periksa variabel yang ada di dag
3. Periksa ada tidaknya variabel kontinu, jika ada maka tampung nilai derajat keanggotaan dan matriks *fuzzy label*
4. Mengambil level dari setiap variabel
5. Bangun matriks *parent* dan *child* untuk membuat *conditional probability table*
6. Menentukan node awal dari dag
7. Membangun *list parent* dan *child*

8. Menelusuri node dari seluruh variabel untuk mengetahui keturunannya
9. Tampung setiap node keturunan untuk masing – masing variabel
10. Bangun *conditional probability table* mengikuti dataTraining
11. Hitung dan tampung hasil hitung probabilitas ke dalam matriks
12. Tampilkan *conditional probability table* dan simpan hasil probabilitas kedalam *list*

Deskripsi Fungsi

Fungsi ini digunakan untuk menghitung nilai probabilitas dan dapat dioperasikan langsung tanpa harus melalui proses *fuzzy partition* apabila data yang dimasukkan tidak memiliki variabel kontinu. Berikut ini contoh menggunakan fungsi bayesianNetwork :

- Apabila telah melalui proses *fuzzy partition* (ada variabel yang bersifat kontinu)

```
dag <- matrix(c("Suhu", "Kecepatan.Angin", "Kelembaban",
               "Kecepatan.Angin", "Kelembaban", "Curah.Hujan"), ncol = 2)
probabilitas <- bayesianNetwork(dag, dataInput, dataTraining,
                                probCountVar = "Curah.Hujan", rangeData, rangeVar,
                                rangeDataTraining, h.fp, h.fptrain)
```

- Apabila tidak melalui proses *fuzzy partition* (seluruh variabel adalah diskrit)

```
dag <- matrix(c("Suhu", "Kecepatan.Angin", "Kelembaban",
               "Kecepatan.Angin", "Kelembaban", "Curah.Hujan"), ncol = 2)
probabilitas <- bayesianNetwork(dag, dataInput, dataTraining,
                                probCountVar = "Curah.Hujan", rangeData, rangeVar,
                                rangeDataTraining)
```

Sama halnya dengan fungsi checkDiscreteContinue, nama variabel yang akan dimasukkan untuk membangun *diagram acyclic graph* harus sama

dengan yang ditampilkan oleh R, karena apabila tidak sama, fungsi `bayesianNetwork` tidak akan bekerja.

4.3.4. Fungsi Defuzzifikasi

Nama Fungsi

`defuzz()`

Input

<code>probCountVar</code>	variabel yang akan didefuzzifikasikan, berbentuk <i>character</i>
<code>probTable</code>	Tabel hasil probabilitas dari variabel yang akan didefuzzifikasi, variabel ini berbentuk <i>list</i> dari matriks
<code>rangeData</code>	jumlah baris dari data masukkan bentuknya <i>integer</i>
<code>countVar.MF</code>	berisikan fungsi keanggotaan dari variabel yang didefuzzifikasi, bentuknya adalah matriks berukuran $(n * 6)$, di mana “n” merupakan jumlah dari diagram fungsi keanggotaannya, lalu variabel ini sudah di set <i>default</i> = <code>h.fp[[3]]</code>

Output

<code>resultCrisp</code>	berisikan nilai tegas dari variabel yang didefuzzifikasi dan bentuknya adalah matriks
--------------------------	---

Proses

1. Tampung seluruh nilai probabilitas
2. Tampung fungsi keanggotaan
3. Apabila nilai probabilitasnya 1 maka langsung lakukan defuzzifikasi

4. Apabila nilai probabilitasnya diantara 1 dan 0 maka bangun daerah kesimpulan *fuzzy*, setelah itu lakukan defuzzifikasi
5. Simpan hasil defuzzifikasi kedalam matriks

Deskripsi Fungsi

Fungsi ini dibangun untuk melakukan defuzzifikasi terhadap variabel kontinu yang dihitung nilainya. Defuzzifikasi dilakukan dengan metode *center of gravity*. Contoh penggunaan fungsi ini adalah sebagai berikut :

- Defuzzifikasi

```
hasil <- defuzz(probCountVar = "Curah.Hujan", probTable =
probabilitas, rangeData)
```

4.4. Mengaplikasikan Model Fuzzy dan Bayesian Network

Model *fuzzy* dan Bayesian network ini dapat dijalankan dengan 3 cara, yang pertama, hanya menggunakan fungsi Bayesian network saja; kedua, menggunakan fungsi *fuzzy partition* manual; ketiga, menggunakan fungsi *fuzzy partition* otomatis. Namun, sebelum memanggil setiap fungsi – fungsi yang ada dalam model *fuzzy* dan Bayesian network, pertama – tama, data yang akan diolah perlu dibaca terlebih dahulu oleh R. Dan berikut ini tahap – tahap menjalankan fungsi yang ada dalam model *fuzzy* dan Bayesian network :

- Apabila seluruh data diskrit

```
#Mengambil Data
dataInput = read.csv("DataBuah.csv")
dataTraining = read.csv("DataBuah.csv")

#Hitung Jumlah Variabel dan Data
rangeData = nrow(dataInput)
rangeDataTraining = nrow(dataTraining)
rangeVar = ncol(dataInput)

dag <- matrix(c("Kelas", "Kulit.Buah", "warna", "Kelas", "Kulit.Buah", "warna", "Ukuran", "Bau"), ncol = 2)
probabilitas = bayesianNetwork(dag, dataInput, dataTraining, probCountVar = "Ukuran", rangeData, rangeVar, rangeDataTraining)
```

Gambar 4.15. Demo Fungsi Bayesian Network

Dari Gambar 4.15 dapat dijelaskan bahwa untuk kasus dengan data yang seluruhnya diskrit memiliki beberapa tahap penyelesaiannya, yang pertama, ambil data latih dan uji menggunakan fungsi `read.csv()`; kedua, hitung jumlah data latih,

jumlah data uji, dan jumlah variabel dari data uji menggunakan fungsi `nrow()` serta `ncol()`; ketiga, bangun DAG dengan fungsi `matrix()`; dan terakhir, jalankan fungsi `bayesianNetwork()` untuk menghitung probabilitas dari variabel yang dituju.

- Apabila terdapat data kontinu dan menggunakan *fuzzy partition* manual

```
dataInput = read.csv("Data/Data Evaluasi/datauji.5.csv")
dataTraining = read.csv("Data/Data Evaluasi/dataTraining.5.csv")

rangeData = nrow(dataInput)
rangeDataTraining = nrow(dataTraining)
rangeVar = ncol(dataInput)

varContinue <- c("Suhu", "Kelembaban", "Curah.Hujan", "Kecepatan.Angin")

suhu.mf <- matrix(c("dingin", "hangat", "panas", 4,1,3,0,22,27,0,25.5,29,22,29,35,25,"-",35), nrow = 6, byrow = TRUE)
kelembaban.mf <- matrix(c("kering", "basah", "lembab", 4,1,3,0,35,80,0,60,85,35,85,100,40,"-",100), nrow = 6, byrow = TRUE)
kecepatan.angin.mf <- matrix(c("lemah", "sedang", "kuat", 4,1,3,0,5,10,0,10,15,5,15,25,10,"-",25), nrow = 6, byrow = TRUE)
curah.hujan.mf <- matrix(c("ringan", "sedang", "lebat", "ekstrem", 4,1,1,3,0,15,50,100,0,45,100,150,5,75,150,175,20,"-",175), nrow = 6, byrow = TRUE)
var.mf <- list(suhu.mf, kelembaban.mf, curah.hujan.mf, kecepatan.angin.mf)
h.fp <- fuzzyPartition(dataInput, varContinue, rangeData, rangeVar, var.mf)

h.fpTrain <- fuzzyPartition(dataTraining, varContinue, rangeDataTraining, rangeVar, var.mf, automatic = FALSE)

dag <- matrix(c("Suhu", "Kecepatan.Angin", "Kelembaban", "Kecepatan.Angin", "Kelembaban", "Curah.Hujan"), ncol = 2)
probabilitas = bayesianNetwork(dag, dataInput, dataTraining, probCountVar = "Curah.Hujan", rangeData, rangeVar, rangeDataTraining, varContinue, h.fp, h.fpTrain)

hasil <- defuzz(probCountVar = "Curah.Hujan", probTable = probabilitas, rangeData)
```

Gambar 4.16 Demo Fuzzy Partition Manual

Dapat dijelaskan dari Gambar 4.16, untuk menyelesaikan kasus menggunakan *fuzzy partition* secara manual terdiri dari beberapa langkah, yang pertama, ambil data latih dan uji menggunakan fungsi `read.csv()`; kedua, hitung jumlah data uji, jumlah data latih, dan jumlah variabel menggunakan fungsi `nrow()` serta `ncol()`; ketiga, tentukan variabel – variabel kontinu dari data lalu simpan dalam variabel `varContinue`; keempat, bangun fungsi keanggotaan dari setiap variabel kontinu menggunakan `matrix()` dan masukkan setiap fungsi keanggotaan kedalam `list()`; kelima, jalankan fungsi *fuzzy partition* secara manual dengan cara mengubah status *automatic* menjadi `FALSE`; keenam, bangun DAG menggunakan fungsi `matrix()` dan jalankan fungsi `bayesianNetwork()`; terakhir, lakukan defuzzifikasi dengan menggunakan fungsi `defuzz()`.

- Apabila terdapat data kontinu dan menggunakan *fuzzy partition* otomatis

```
dataInput = read.csv("Data/Data Evaluasi/dataUji.5.csv")
dataTraining = read.csv("Data/Data Evaluasi/dataTraining.5.csv")

rangeData = nrow(dataInput)
rangeDataTraining = nrow(dataTraining)
rangeVar = ncol(dataInput)

varContinue <- c("Suhu", "Kelembaban", "Curah.Hujan", "Kecepatan.Angin")

#Melakukan Fuzzy Partition Secara OTOMATIS
h.fp <- fuzzyPartition(dataInput, varContinue, rangeData, rangeVar, var.mf=NULL, automatic = TRUE, N=4)
h.fpTrain <- fuzzyPartition(dataTraining, varContinue, rangeDataTraining, rangeVar, var.mf=NULL, automatic = TRUE, N=4)

dag <- matrix(c("Suhu", "Kecepatan.Angin", "Kelembaban", "Kecepatan.Angin", "Kelembaban", "Curah.Hujan"), ncol = 2)
probabilitas = bayesianNetwork(dag, dataInput, dataTraining, probCountVar = "Curah.Hujan", rangeData, rangeVar, rangeDataTraining, varContinue, h.fp, h.fpTrain)

hasil <- defuzz(probCountVar = "Curah.Hujan", probTable = probabilitas, rangeData)
```

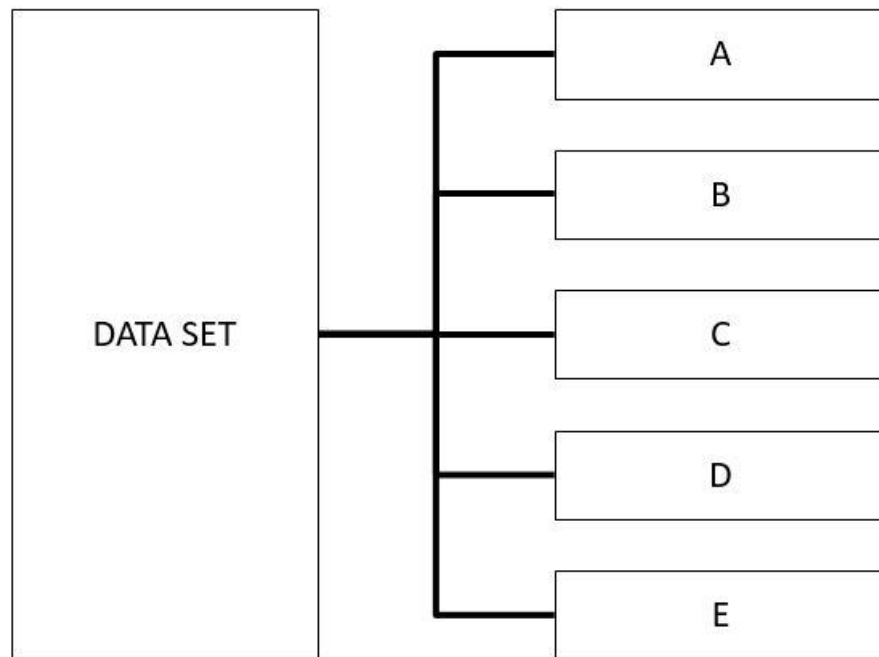
Gambar 4.17 Demo Fuzzy Partition Otomatis

Berdasarkan Gambar 4.17, maka dapat dijelaskan bahwa untuk mengatasi kasus menggunakan *fuzzy partition* secara otomatis terdiri dari beberapa langkah, yang pertama, ambil data latih dan data uji menggunakan fungsi `read.csv()`; kedua, hitung banyak data latih, data uji, dan banyak variabel pada data menggunakan fungsi `nrow()` serta `ncol()`; ketiga, tentukan variabel kontinu dari data dan masukkan kedalam variabel `varContinue`; keempat, jalankan *fuzzy partition* dengan mengubah status `automatic` menjadi `TRUE` agar *fuzzy partition* dijalankan secara otomatis; kelima, bangun DAG menggunakan fungsi `matrix()` dan hitung probabilitas menggunakan fungsi `bayesianNetwork()`; keenam, jalankan fungsi `defuzz()` untuk melakukan defuzzifikasi.

4.5. Rancangan Eksperimen

Validasi silang adalah metode evaluasi model yang lebih baik dari residual. Ide dasar dari metode validasi silang adalah tidak menggunakan seluruh data set ketika proses pelatihan model. Lalu beberapa data dibuang sebelum memulai pelatihan dan ketika proses latihan selesai, data yang tadi dibuang dapat digunakan untuk menguji performa dari model terlatih pada data yang baru. Banyak metode yang dapat dilakukan untuk validasi silang, diantaranya ada metode *holdout*, *k-fold*, dan *leave-one-out*. Dari setiap metode validasi silang yang ada, penulis menggunakan metode *k-folds*.

Validasi silang *k-folds* bekerja dengan membagi data set sebanyak jumlah k subset. Satu subset k digunakan sebagai data uji dan sisanya digunakan untuk data latih. Dan untuk mengoperasikan validasi silang *k-folds* ini, penulis menggunakan *package* di R yang bernama *cvTools*. Lalu dengan *package* ini, penulis bagi data kedalam 5 *folds* dari seluruh data yang berjumlah 1660 baris.



Gambar 4.18 Pembagian Data Dengan *K-Folds*

Dari Gambar 4.18 dapat dijelaskan bahwa total data set yang berjumlah 1660 baris dibagi menjadi 5 subset, ada subset A, B, C, D, dan E. Lalu setiap subset terdiri dari 332 baris data yang dihasilkan dari pengambilan secara acak dari data set.

Tabel 4.17 Data Latih dan Data Uji

DATA LATIH	DATA UJI
A, B, C, dan D	E
A, B, C, dan E	D
A, B, D, dan E	C
A, C, D, dan E	B
B, C, D, dan E	A

Dari Tabel 4.17 dapat dijelaskan bahwa pengujian model dilakukan sebanyak jumlah *folds* nya yaitu 5 kali. Pada pengujian pertama, data latih merupakan gabungan subset data A, B, C, dan D sedangkan E menjadi data uji; pengujian kedua, data latih merupakan gabungan subset data A, B, C, dan E sedangkan D menjadi data uji; pengujian ketiga, data latih merupakan gabungan subset data A, B, D, dan E sedangkan C menjadi data uji; pengujian keempat, data latih merupakan gabungan subset data A, C, D, dan E sedangkan B menjadi data uji; serta pengujian terakhir, data latih merupakan gabungan subset data B, C, D, dan E sedangkan A menjadi data uji.

Lalu untuk pengujian modelnya, penulis melakukan 7 skenario. Skenario pertama untuk *fuzzy* manual berdasarkan pakar, kedua dan ketiga untuk *fuzzy* manual buatan sendiri dengan 3 fungsi keanggotaan namun diagram yang berbeda, keempat sampai ketujuh untuk *fuzzy* Wang Mendel dengan diagram fungsi keanggotaan yang sama namun dengan jumlah fungsi keanggotaan yang berbeda – beda.

4.6. Analisis dan Hasil Eksperimen

Pengujian dilakukan terhadap model *fuzzy* Bayesian network di mana fuzzifikasi nya manual dan otomatis (fuzzifikasi Wang Mendel). Lalu dalam pengujian ini, penulis menggunakan *root mean squared error* (RMSE) untuk menghitung tingkat galat serta fungsi *Sys.time* di R untuk menghitung kecepatan dari model. Berikut ini merupakan hasil pengujian aplikasi :

Tabel 4.18 Hasil Pengujian Manual Fuzzy – 1

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	8.35	21.12
2	9.12	21.46
3	8.36	21.44
4	8.97	22.43
5	8.36	21.25

RATA-RATA	8.632	21.54
------------------	-------	-------

Dari Tabel 4.18 dapat dijelaskan, nilai *error* terkecil untuk pengujian *fuzzy* manual adalah pada pengujian pertama yaitu 8.35 % dengan kecepatan komputasi 21.12 detik, lalu nilai *error* tertinggi didapatkan pada pengujian kedua yaitu 9.12 % dengan kecepatan komputasi 21.46, dan walaupun pada pengujian keempat memiliki *error* yang cukup kecil yakni 8.97 %, tapi kecepatan komputasinya termasuk yang paling lama dibandingkan dengan pengujian yang lainnya yaitu 22.43 detik.

Tabel 4.19 Hasil Pengujian Manual Fuzzy – 2

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	9.07	17.09
2	10.71	16.75
3	9.88	16.17
4	9.61	16.06
5	9.07	16.84
RATA-RATA	9.668	16.58

Lalu dari Tabel 4.19 dapat dijelaskan bahwa pengujian *fuzzy* manual yang kedua memiliki rata – rata kecepatan komputasi yang lebih baik dari pengujian yang pertama yaitu 16.58 detik, namun rata – rata tingkat *error* nya lebih besar daripada pengujian yang pertama yaitu 9.668 %.

Tabel 4.20 Hasil Pengujian Manual Fuzzy – 3

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	8.28	16.1
2	8.52	16.87
3	8.79	16.05

4	8.82	16.91
5	8.28	16.03
RATA-RATA	8.54	16.39

Lalu hasil pengujian *fuzzy* manual ketiga pada Tabel 4.20, menunjukkan rata – rata tingkat *error* yang lebih kecil dan kecepatan yang lebih baik dibandingkan dengan *fuzzy* manual pertama dan kedua, yakni dengan tingkat *error* 8.54 % dan kecepatan komputasi 16.39 detik.

Tabel 4.21 Hasil Pengujian Automatic Fuzzy – 1

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	10.45	16.17
2	13.04	17.04
3	16.06	16.68
4	16.54	16.79
5	11.94	16.04
RATA-RATA	13.61	16.54

Lalu hasil pengujian *fuzzy* otomatis pada Tabel 4.21 menunjukkan, tingkat *error* terkecil didapatkan ketika pengujian pertama yaitu 10.45 % dengan kecepatan komputasi 16.17 detik, walaupun tingkat *error* nya paling kecil diantara lima pengujian, komputasi tercepat didapatkan pada pengujian kelima yaitu 16.04 detik.

Tabel 4.22 Hasil Pengujian Automatic Fuzzy – 2

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	9.81	121.2
2	12.94	121.2
3	10.45	120.6

4	12.48	127.2
5	11.42	124.8
RATA-RATA	11.42	123

Pada Tabel 4.22, dapat dijelaskan, tingkat *error* terkecil terjadi ketika pengujian pertama yaitu 9.81 %, walaupun pengujian kedua memiliki kecepatan komputasi yang sama dengan pengujian kesatu yaitu 121.2 detik, namun nilai *error* nya paling tinggi yaitu 12.94 %, lalu kecepatan komputasi paling tinggi didapatkan ketika pengujian ketiga yaitu 120.6 %.

Tabel 4.23 Hasil Pengujian Automatic Fuzzy – 3

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	13.63	496.2
2	13.92	496.2
3	16.37	493.2
4	16.47	502.2
5	13.88	494.4
RATA-RATA	14.85	496.44

Selanjutnya, melalui hasil pengujian *fuzzy* otomatis ketiga yang ditampilkan pada Tabel 4.23, dapat disimpulkan bahwa nilai *error* terkecil ditampilkan ketika pengujian ke pertama yaitu 13.63 %, namun rata – rata kecepatan dari *fuzzy* otomatis ketiga menunjukkan waktu yang paling lama daripada *fuzzy* manual, *fuzzy* otomatis pertama, juga *fuzzy* otomatis kedua yaitu 496.44 detik atau sama dengan 8.27 menit.

Tabel 4.24 Hasil Pengujian Automatic Fuzzy - 4

PENGUJIAN	RMSE	KECEPATAN
	(%)	(detik)
1	12.47	1462.2

2	8.5	1465.2
3	15.12	1459.8
4	15.87	1466.4
5	12.8	1467
RATA-RATA	12.95	1464.12

Dan terakhir adalah hasil pengujian *fuzzy* otomatis keempat yang ditampilkan pada Tabel 4.24, dapat disimpulkan bahwa *fuzzy* otomatis keempat memiliki tingkat *error* terkecil pada pengujian kedua yaitu 8.5 %, namun jika dibandingkan dengan pengujian *fuzzy* lainnya, *fuzzy* otomatis keempat ini menunjukkan kecepatan komputasi paling lama yakni 1464.12 detik atau sama dengan 24.4 menit.

Dari seluruh hasil pengujian yang telah didapatkan (Tabel 4.18 sampai 4.24), maka disimpulkan bahwa akurasi dan kecepatan pada masing – masing pengujian adalah sebagai berikut (Tabel 4.25):

Tabel 4.25 Hasil Pengujian Model

SKENARIO	FUZZY	JUMLAH FUNGSI KEANGGOTAAN	DIAGRAM FUNGSI KEANGGOTAAN	AKURASI	KECEPATAN
1	Manual-1	4	1. Trapezoid Kiri 2. Segitiga 3. Segitiga 4. Trapezoid Kanan	91.37 %	21.54 detik
2	Manual-2	3	1. Trapezoid Kiri 2. Trapezoid Tengah 3. Trapezoid Kanan	90.33 %	16.58 detik
3	Manual-3	3	1. Trapezoid Kiri 2. Segitiga 3. Trapezoid Kanan	91.46 %	16.39 detik

4	Automatic-1	3	Segitiga	86.39 %	16.54 detik
5	Automatic-2	5	Segitiga	88.15 %	2.05 menit
6	Automatic-3	7	Segitiga	85.15 %	8.27 menit
7	Automatic-4	9	Segitiga	87.05 %	24.4 menit

Dari Tabel 4.25 dapat dijelaskan bahwa skenario pertama hingga ketujuh menunjukkan *fuzzy* yang digunakan, jumlah fungsi keanggotaannya, diagram fungsi keanggotaannya, akurasi yang didapatkan berdasarkan tingkat *error* dari masing – masing skenario, dan kecepatan komputasi dari masing – masing skenario. Lalu dapat disimpulkan dari Tabel 4.25, bahwa manual *fuzzy* ke 3 dengan 3 fungsi keanggotaan memiliki tingkat akurasi dan kecepatan yang lebih baik dibandingkan dengan model *fuzzy* yang lain, yakni dengan akurasi 91.46 % dan kecepatan 16.39 detik. Lalu *automatic fuzzy* ke 4 dengan 9 fungsi keanggotaan memiliki kinerja yang lebih lama dibanding dengan model *fuzzy* lainnya yaitu 24.4 menit. Dan dapat disimpulkan pula, bahwa di seluruh pengujian *automatic fuzzy*, penggunaan model *automatic fuzzy* ke 2 dengan 5 fungsi keanggotaan memiliki akurasi yang lebih baik dibandingkan dengan *automatic fuzzy* lainnya yaitu 88.15 % walaupun kecepatannya 2.05 menit.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dan pembahasan yang telah dilakukan, maka dapat ditarik kesimpulan sebagai berikut :

1. Untuk menghitung prediksi cuaca menggunakan kombinasi metode *fuzzy* dan Bayesian network adalah dengan melakukan proses diskritisasi, hitung probabilitas menggunakan Bayesian network, dan lakukan defuzzifikasi.
2. Kombinasi metode *fuzzy* dan Bayesian network dapat dilakukan secara efektif dengan mengotomatiskan proses fuzzifikasi menggunakan cara yang diajukan oleh Wang dan Mendel.
3. Akurasi dan kecepatan dari kombinasi metode *fuzzy* dan Bayesian network ini dapat diterima, baik untuk *fuzzy* berdasarkan pakar maupun berdasarkan Wang dan Mendel.

5.2. Saran

Berikut ini adalah beberapa saran untuk penelitian lebih lanjut:

1. Mengurangi penggunaan perulangan atau menggunakan sistem *parallel computing* untuk meningkatkan kecepatan komputasi dari aplikasi kombinasi metode *fuzzy* dan Bayesian network.
2. Menambahkan fungsi keanggotaan lainnya dalam fungsi *fuzzy partition* untuk meningkatkan akurasi dari aplikasi kombinasi metode *fuzzy* dan Bayesian network.

DAFTAR PUSTAKA

- Aisjah, A. S., & Arifin, S. (2011). Maritime Weather Prediction using Fuzzy Logic in Java Sea. *Instrumentation Control and Automation*, 205-208.
- Allaby, M., & Garratt, R. (2003). *A Chronology of Weather*. New York: Facts on File.
- Battaini, M., Casciati, F., & Faravelli, L. (1998). FUZZY CONTROL OF STRUCTURAL VIBRATION. AN ACTIVE MASS SYSTEM DRIVEN BY A FUZZY CONTROLLER. *EARTHQUAKE ENGINEERING AND STRUCTURAL DYNAMICS*, 1267-1276.
- Ben-Gal, I. (2007). Bayesian Network. *Encyclopedia of Statistics in Quality and Reliability*.
- Chai, H., & Wang, B. (2011). A Hierarchical Situation Assessment Model Based on Fuzzy Bayesian Network. 444-454.
- Chaves, R. R., Ross, R. S., & Krishnamurti, T. N. (2005). Weather and Seasonal Climate Prediction for South America Using a Multi-Model Superensemble. *INTERNATIONAL JOURNAL OF CLIMATOLOGY*, 1881-1914.
- Chen, P., & Zabaras, N. (2014). Uncertainty quantification for multiscale disk forging of polycrystal materials using probabilistic graphical model techniques. *Computational Materials Science*, 278-292.
- Corne, D., Dissanayake, M., Peacock, A., Galloway, S., & Owens, E. (2014). Accurate Localized Short Term Weather Prediction for Renewables Planning. *Symposium on Computational Intelligence Applications in Smart Grid*, 1-8.
- Corso, J. J., Ye, G., & Hager, G. D. (2005). Analysis of composite gestures with a coherent probabilistic graphical model. *Computational Interaction and Robotics Lab*, 242-252.

- D'Angelo, M. F., Palhares, R. M., Cosme, L. B., Aguiar, L. A., Fonseca, F. S., & Caminhas, W. M. (2014). Fault detection in dynamic systems by a Fuzzy/Bayesian network formulation. *Applied Soft Computing*.
- Fang, J., Huang, J., & Liu, M. (2012). The research on fuzzy Bayesian network model for the network public opinion situation and threat assessment. *Third International Conference on Networking and Distributed Computing*.
- Fontaine, B. L. (2004). *All About the Weather*. Dover Publications.
- Harahap, E., Sakamoto, W., & Nishi, H. (2010). Failure Prediction Method for Network Management System by using Bayesian Network and Shared Database. *Information and Telecommunication Technologies*, 1-6.
- Hristea, F. T. (2013). *The Naive Bayes Model for Unsupervised Word Sense Disambiguation*. London: Springer.
- Hsu, C.-C., Huang, Y.-P., & Chang, K.-W. (2008). Extended Naive Bayes Classifier for Mixed Data. *Expert Systems with Application*, 1080-1083.
- Ihaka, R., & Gentleman, R. (1997). R: A Language for Data Analysis and Graphics.
- Kasahara, A. (1974). Various Vertical Coordinate Systems Used for Numerical Weather Prediction.
- Kimura, R. (2002). Numerical weather prediction. *Journal of Wind Engineering*, 1403-1414.
- Koten, C. v., & Gray, A. R. (2005). An application of Bayesian network for predicting object-oriented software maintainability. *Information and Software Technology*, 59-67.
- Leiming, M., Fuhai, G., Qiqige, W., & Guangqiang, Z. (2012). Numerical Weather Prediction in Yangtze River Delta Region with Assimilation of AWS and GPS/PWV Data. *IEEE Symposium on Robotics and Application*, 741-743.
- Leroy, G., Miller, T., Rosembat, G., & Browne, A. (2008). A Balanced Approach to Health Information Evaluation: A Vocabulary-Based Naive Bayes

- Classifier and Readability Formulas. *Journal of The American Society for Information Science and Technology*, 1409-1419.
- Liu, W.-Y., Yue, K., & Gao, M.-H. (2011). Constructing probabilistic graphical model from predicate formulas for fusing logical and probabilistic knowledge. *Information Sciences*, 3828-3845.
- Lorenz, E. N. (1960). Energy and Numerical Weather Prediction.
- Mamdani, E. H., & Assilian, S. (1975). An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *Int. J. Man-Machine Studies*, 1-13.
- Morrison, M. L., McCluggage, W. G., Price, G. J., Diamond, J., Sheeran, M. R., Mulholland, K. M., . . . Hamilton, P. W. (2002). Expert system support using a Bayesian belief network for the classification of endometrial hyperplasia. *Journal of Pathology*, 403-414.
- Nandar, A. (2009). Bayesian Network Probability Model for Weather Prediction.
- Pasquier, E., Pernot, L. D., Burdin, V., Mounayer, C., Rest, C. L., Colin, D., . . . Baccino, E. (1999). Determination of Age at Death: Assessment of an Algorithm of Age Prediction Using Numerical Three-Dimensional CT Data From Pubic Bones. *AMERICAN JOURNAL OF PHYSICAL ANTHROPOLOGY*, 261-268.
- Rego, J. L., & Li, C. (2010). Storm Surge Propagation in Galveston Bay during Hurricane Ike. *Journal of Marine Systems*, 265-279.
- Riza, L. S., Bergmeir, C., Herrera, F., & Benitez, J. M. (2014). Learning from Data Using the R Package "frbs". *IEEE International Conference on Fuzzy Systems*.
- Riza, L. S., Bergmeir, C., Herrera, F., & Benitez, J. M. (2015). frbs: Fuzzy Rule-Based Systems for Classification and Regression in R. *Journal of Statistical Software*, 1-30.

- Roos, J., Bonnevey, S., & Gavin, G. (2016). Short-Term Urban Rail Passenger Flow Forecasting: A Dynamic Bayesian Network Approach. *International Conference on Machine Learning and Application*.
- Ruzic, M. D., Skenderovic, J., & Lesic, K. T. (2016). Application of The Mamdani Fuzzy Inference System to Measuring HR Performance in Hotel Companies. *A PILOT STUDY*, 976-999.
- Saepullah, A., & Wahono, R. S. (2015). Comparative Analysis of Mamdani, Sugeno And Tsukamoto Method of Fuzzy Inference System for Air Conditioner Energy Saving. *Journal of Intelligent Systems*, 143-147.
- Sanchez-Torrubia, G., & Torres-Blanc, C. (2010). A MAMDANI-TYPE FUZZY INFERENCE SYSTEM TO AUTOMATICALLY ASSESS DIJKSTRA'S ALGORITHM SIMULATION. *International Journal Information Theories and Applications*, 35-100.
- Setyaningrum, A. H., & Swarinata, P. M. (2014). Weather Prediction Application Based on ANFIS (Adaptive Neural Fuzzy Inference System) Method In West Jakarta Region. *Cyber and IT Service Management*, 113-118.
- Stiros, S. C. (1997). Costs and benefits of earthquake prediction studies in Greece. *ASSESSMENT OF SCHEMES FOR EARTHQUAKE PREDICTION*, 478-484.
- Sugimoto, T., Murakami, H., Kozuki, Y., Nishikawa, K., & Shimada, T. (2002). A Human Damage Prediction Method for Tsunami Disasters Incorporating Evacuation Activities. 585-600.
- Sundqvist, H., Berge, E., & Kristjansson, J. E. (1989). Condensation and Cloud Parameterization Studies with a Mesoscale Numerical Weather Prediction Model. *American Meteorological Society*, 1641-1657.
- Tang, H., & Liu, S. (2007). Basic Theory of Fuzzy Bayesian Networks and Its Application in Machinery Fault Diagnosis. *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*.

- Tang, Y. M., Capon, R., Forbes, R., & Clark, P. (2009). Fog Prediction Using a Very High Resolution Numerical Weather Prediction Model Forced With a Single Profile. *METEOROLOGICAL APPLICATIONS*, 129-141.
- Vahidinasab, V., & Jadid, S. (2008). Bayesian neural network model to predict day-ahead electricity prices. *European Transactions on Electrical Power*, 231-246.
- Vandelli, C., Renzo, F., Braun, H. B., Tisminetzky, S., Albrecht, M., Palma, M. D., . . . Michel, G. (1999). Prediction of Successful Outcome in a Randomised Controlled Trial of the Long-Term Efficacy of Interferon Alpha Treatment for Chronic Hepatitis C. *Journal of Medical Virology*, 26-34.
- Wang, L.-X., & Mendel, J. M. (1991). GENERATING FUZZY RULES BY LEARNING FROM EXAMPLES. *Proceedings of The 1991 IEEE International Symposium on Intelligent Control*.
- Wang, M., & Zhou, J. (2014). A Bayesian Network-based Classifier for Machining Error Prediction. *International Conference on Advanced Intelligent Mechatronics*, 841-844.
- Weber, P., Medina-Oliva, G., Simon, C., & Iung, B. (2010). Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, 671-682.
- Yamada, K. (1999). Real-Time Prediction of Near-Future Seismic Excitation Adapting AR Model to Preceding Information. *Earthquake Engineering and Structural*, 1587-1599.
- Yao, J. Y., Li, J., Li, H., & Wang, X. (2015). Modeling System Based on Fuzzy Dynamic Bayesian Network for Fault Diagnosis and Reliability Prediction. *Annual Reliability and Maintainability Symposium*, 1-6.
- Yin, H., Lin, C., Sebastien, B., Li, B., & Min, G. (2005). Network Traffic Prediction Based on a New Time Series Model. *International Journal of Communication*, 711-729.

Zadeh, L. A. (1965). Fuzzy Sets.

Zadeh, L. A. (1968). Communication Fuzzy Algorithms.

Zadeh, L. A. (1975). The Concept of a Linguistic Variable and its Application to Approximate Reasoning - 1. *Information Sciences*, 199-249.

Zhang, H. (2004). The Optimality of Naive Bayes. *American Association for Artificial Intelligence*.

Zhang, Y., Hu, J., Dong, J., Yuan, Y., Zhou, J., & Shi, J. (2009). Location Prediction Model Based on Bayesian Network Theory.

Zhao, J., & Zhang, W. (2013). Study of fault diagnosis method based on fuzzy Bayesian network and application in CTCS-3 train control system. *Study of test cases building technology*.

Zheng, X., & Liu, M. (2010). Forecasting model for pedestrian distribution under emergency evacuation. *Reliability Engineering and System Safety*, 1186-1192.

LAMPIRAN