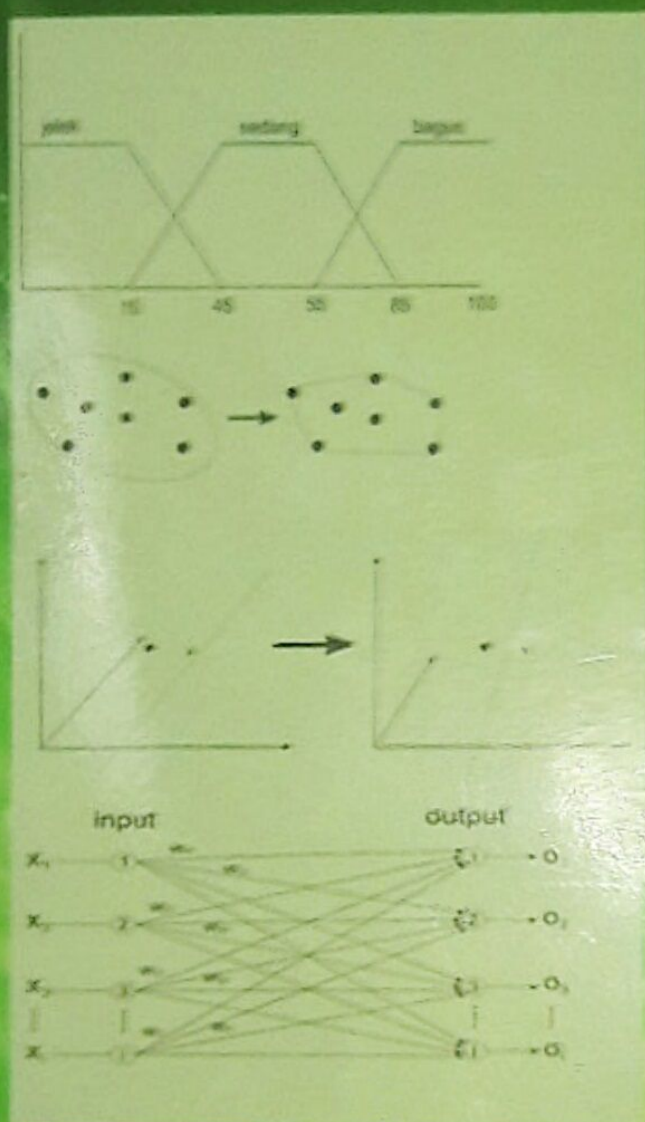


SISTEM PAKAR

Studi Kasus dan Pemecahannya



Priadhana Edi Kresnha, M.Kom
Ir. Eka Budhy Prasetya, M.M
Emi Susilowati, M.Kom

SISTEM PAKAR

Studi Kasus dan Penyelesaiannya

Tim Penyusun	: Priadhana Edi Kresna, S.Kom., M.Kom Ir. Eka Budhy Prasetya, M.M Emi Susilowati, S.Kom., M.Kom
Penerbit	: PT. Canting Mas Anyar Perum Agatama Regency Banguntapan A8 Yogyakarta. Telp. 082116448037
Percetakan	: PT. Canting Mas Anyar Perum Agatama Regency Banguntapan A8 Yogyakarta. Telp. 082116448037
Cetakan I	: September 2018 iv + 33 hal., 15,5x23,5 cm
ISBN	: 978-602-52162-6-8

Hak Cipta © 2018 pada penerbit

Hak Cipta dilindungi undang-undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh buku ini dalam bentuk apapun, secara elektronik maupun mekanis, termasuk memfotokopi, merekam, atau dengan teknik periklanan lainnya, tanpa izin tertulis dari penerbit.

DAFTAR ISI

DAFTAR ISI	iv
KATA PENGANTAR	1
BAB 1	2
<i>FUZZY MATHEMATICS</i>	2
1.1. Pendahuluan	2
1.2. Logika Fuzzy	3
1.2.1. Logika Fuzzy	3
1.2.2. <i>FUZZY MATHEMATICS</i>	4
1.2.2.1. <i>Evaluation Factors Set</i>	4
1.2.2.2. <i>Rank Level Set</i>	4
1.2.2.3. <i>Fuzzy Relationship Matrix</i>	5
1.2.2.4. Bobot Faktor Evaluasi	5
1.2.2.5. Total Rank	5
1.3. Studi Kasus Online Learning & Collaboration Learning	6
1.3.1. Pemetaan Masalah	6
1.3.1.1. Pemetaan <i>Evaluation Factor</i>	8
1.3.1.2. Perhitungan <i>Fuzzy Membership Function</i>	9
1.3.1.3. Penentuan Bobot Faktor Evaluasi	11
1.3.1.4. Alur Perhitungan <i>Fuzzy Mathematics</i>	11
1.3.2. Pemecahan Menggunakan <i>Fuzzy Mathematics</i>	13
1.3.2.1. Perhitungan <i>Evaluation Factor Set</i>	14
1.3.2.2. Bobot Faktor Evaluasi dan Hasil Akhir	17
1.4. Ringkasan <i>Fuzzy Mathematics</i>	19
1.5. Latihan <i>Fuzzy Mathematics</i>	20
BAB 2	22

<i>LINEAR VECTOR QUANTIZATION</i>	22
2.1. Pendahuluan	22
2.2. Arsitektur <i>Linear Vector Quantization</i>	25
2.3. <i>Sample Case</i>	26
2.4. Optimalisasi Inisialisasi Bobot Vektor.....	31
2.5. Ringkasan LVQ.....	32
2.6. Latihan LVQ.....	32
DAFTAR PUSTAKA.....	33

KATA PENGANTAR

Assalamua'alaikum wr. wb.
Bismillahirrohmanirrohiim

Puja dan puji syukur kami haturkan sepenuhnya kepada Allah SWT, yang atas rahmat dan nikmat yang diberikan, buku modul bahan ajar mata kuliah Sistem Pakar ini dapat diselesaikan. Berkat doa tim penulis dan pihak-pihak pendukung pula, buku ini bisa diselesaikan dengan baik, walaupun mungkin masih jauh dari kata sempurna. Bisa jadi ada beberapa poin yang akan diketahui nanti, yang harus diperbaiki untuk menjadikan buku ini lebih sempurna.

Ucapan terima kasih juga kami sampaikan kepada semua pihak, yaitu *stakeholder* yang telah, secara langsung maupun tidak langsung, berperan serta dalam pengembangan buku ini, dari tahap awal perumusan, pengumpulan data, penelitian, hingga terbentuk produk jadi yang bisa dibaca oleh semua orang, baik itu pelajar, akademisi, maupun insan lain yang tertarik mempelajari sistem pakar dan cara pemecahan masalah berupa *case-case* yang bisa dipecahkan dengan sistem pakar ini.

Buku ini disusun dengan bantuan hibah penelitian yang diselenggarakan oleh kemenristekdikti dan LPPM (Lembaga Penelitian dan Pengembangan Masyarakat) UMI antara tahun 2016 hingga 2018. Untuk itu kami mengucapkan terima kasih sebesar-besarnya kepada kemenristekdikti dan Universitas Muhammadiyah Jakarta atas hibah yang diselenggarakan. Semoga kedepannya penerima hibah semakin bertambah, dan nominal yang diberikan juga semakin besar, demi kesuksesan penelitian dan peningkatan pengetahuan para pengajar dan peneliti di Perguruan Tinggi Indonesia.

Akhir kata, silakan membaca dan mempelajari buku ini. Semoga buku ini bermanfaat bagi para pembaca, dan bisa menambah ilmu dan khazanah, serta mengembangkan kemampuan iptek di negeri kita.

Wassalamu'alaikum wr. wb.
Jakarta, September 2018,
Program Studi Informatika FT-UMI,

Penulis

BAB I

FUZZY MATHEMATICS

1.1. Pendahuluan

Fuzzy merupakan salah satu cabang ilmu komputer, lebih tepatnya bidang *expert system*. Fuzzy digunakan untuk pemecahan masalah secara cepat dan relatif tepat, sebagaimana manusia menentukan keputusan dalam masalah yang sedang dihadapi. Tidak harus tepat 100%, namun hasil cukup memuaskan, dan masih berada di wilayah "bisa diterima".

Dalam perkembangan selanjutnya, setelah fuzzy bisa diterima di dunia kecerdasan buatan, muncul berbagai teori tentang fuzzy. Dimulai dari sekedar perubahan himpunan *crisp* menjadi himpunan yang didalamnya terdapat elemen yang memiliki tingkat keanggotaan tertentu terhadap himpunan tersebut, hingga modifikasi berbagai metode yang dimasukkan fuzzy didalamnya, seperti FLVQ (Fuzzy Linear Vector Quantization), FNN (Fuzzy Neural Network), FGA (Fuzzy Genetic Algorithms), dan sebagainya.

Selain itu, ada pula teori fuzzy berdasarkan cara memecahkan masalahnya. Metode fuzzy ini diberi nama sesuai dengan *inventor*-nya. Contohnya Fuzzy Mamdani, diciptakan oleh Mahmood Mamdani, menjelaskan bagaimana melakukan defuzzifikasi menggunakan output berupa fungsi keanggotaan. Kemudian ada Fuzzy Sugeno, Fuzzy Takagi-Sugeno-Kang, dan lain sebagainya.

Pada buku ini, pembahasan akan difokuskan pada Fuzzy Mathematics, Fuzzy ini digunakan oleh (Shuqin, 2011) untuk menganalisa faktor kualitas dari kuliah online. Fuzzy ini, dengan penggabungan dengan perhitungan skala likert, juga digunakan oleh (Kresnia, Susilowati, & Ambo, 2016) untuk menganalisa efektifitas kuliah berbantu online di salah satu institusi di Jakarta.

Beberapa elemen / istilah yang ada di *fuzzy mathematics* di buku ini antara lain *Evaluation Factor Set*, *Rank*, *Level Set*, *Fuzzy Relationship Matrix*, Bobot Faktor Evaluasi, *Total Rank*, dan *Fuzzy Result*. *Fuzzy Result* menentukan efektifitas keseluruhan dari penerapan metode pembelajar berbasis online & kolaborasi.

Bab pertama ini ditujukan untuk membantu pelajar dalam mempelajari logika fuzzy lebih dalam, terutama pada perhitungan *fuzzy mathematics*. Setelah mempelajari *fuzzy mathematics*, diharapkan pelajar diharapkan dapat lebih mengerti logika fuzzy, dan variannya, yaitu *fuzzy mathematics*. Pelajar juga bisa memecahkan permasalahan yang diberikan pada soal di buku ini, dan bisa menggunakan metode ini untuk memecahkan permasalahan yang ada di dunia nyata. Pelajar juga bisa menerapkan cara pemecahan yang ada di buku ini untuk studi kasus tugas akhir logika fuzzy, maupun tugas akhir kuliah (skripsi).

1.2. Logika Fuzzy

1.2.1. Logika Fuzzy

Jauh sebelum teori logika fuzzy berkembang, terdapat pemikiran yang mirip dengan teori ini. Pemikiran ini bermula pada 500 SM, ketika agama Buddha pertama kali muncul. Filosofinya adalah dunia tersusun atas kontradiksi, dan segala sesuatu bisa mengandung hal yang berlawanan dengan sesuatu tersebut. Contohnya ada elemen A, bisa masuk ke dalam golongan AA dan bukan AA pada waktu yang sama.

Berlawanan dengan filosofi Buddha, 200 tahun kemudian muncul aristoteles yang memiliki pemikiran logika *biner*. Segala sesuatu berlawanan dengan hal yang berlawanan, dimana elemen yang sudah masuk ke 1 himpunan, tidak bisa masuk ke himpunan lain pada waktu yang sama. Contohnya pria vs wanita, panas vs dingin, basah vs kering, dan sebagainya. Teori ini diperkuat dengan munculnya aljabar *boolean*, yang diciptakan oleh George Boole pada pertengahan abad 19, yang hingga kini menjadi dasar perhitungan komputer.

Pemikiran fuzzy muncul ketika pada tahun 1964, dimana Prof. Asker Zadeh berpikir, bahwa ada logika yang lebih baik digunakan untuk mesin produksi dan mesin pekerja. Contohnya adalah pengatur suhu ruangan (AC). Kita bisa mengatur sedemikian rupa, sehingga AC bekerja lebih keras ketika suhu udara sudah mulai memanas. Sebaliknya pula, kerjanya akan berkurang ketika suhu udara perlahan sudah mendekati suhu yang diinginkan. Akan lebih efisien jika hal ini diimplementasikan dalam bentuk aturan (*rule*) untuk setiap kondisi temperatur.

Realisasi dari ide ini adalah ditulisnya paper berjudul *Fuzzy Sets* (Zadeh, 1965) yang dipublikasikan tahun 1965. Dalam paper tersebut terdapat penjelasan mengenai Himpunan Fuzzy, dan berbagai operator fuzzy yang diadopsi dari operasi himpunan *crisp* biasa, seperti *inclusion, union, intersection, relation, convexity, AND, OR, NOT*, dan sebagainya.

Setelah melalui beberapa tahun, *fuzzy* telah berkembang dengan berbagai variannya, seperti *fuzzy mamdani, fuzzy takagi-sugeno-kang, fuzzy lvq*, dan lain sebagainya. Ada pula *fuzzy mathematics* yang akan di jelaskan pada subbab berikutnya.

1.2.2. FUZZY MATHEMATICS

Fuzzy Mathematics adalah salah satu bentuk dari logika *fuzzy* dengan beberapa karakteristik matematika, antara lain faktor evaluasi, rank, dan matrix. Salah satu penggunaan *Fuzzy mathematics* adalah untuk menganalisa data yang berasal dari survey dengan pilihan jawaban terbatas, sebagaimana yang dilakukan oleh (Shuaqin, 2011) dan (Kresna, Susilowati, & Ambo, 2016). Pada makalah yang ditulis oleh (Shuaqin, 2011), komponen yang ada dalam *fuzzy mathematics* adalah *evaluation factor, rank level set, fuzzy relation matrix, dan weighted factor evaluation*. Pada penelitian tersebut, *evaluation factor* diambil dari pertanyaan survey, sementara pilihan jawabannya menjadi *linguistic variables*. Kedua elemen tersebut menyusun *fuzzy relation matrix* dan digunakan dalam perhitungan pembobotan faktor evaluasi.

1.2.2.1. Evaluation Factors Set

Evaluation Factor Set dilambangkan dengan bentuk berikut,

$$P \text{ Evaluation factors } : u = \{u, u_1, \dots, u_p\} \quad (1)$$

Dimana u adalah nilai faktor evaluasi dan p adalah banyaknya faktor evaluasi.

1.2.2.2. Rank Level Set

Evaluation Factor Set dilambangkan dengan bentuk berikut,

$$P \text{ Evaluation factors } : u = \{u, u_1, \dots, u_p\} \quad (2)$$

Dimana u adalah nilai faktor evaluasi dan p adalah banyaknya faktor evaluasi.

1.2.2.3. Fuzzy Relationship Matrix

Fuzzy Relationship Matrix berbentuk sebagai berikut,

$$R = \begin{matrix} \begin{matrix} [R] \\ [R] \\ \dots \\ [R] \end{matrix} \begin{matrix} | \\ u_1 \\ | \\ u_p \end{matrix} \end{matrix} = \begin{matrix} \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & \dots & \dots & \dots \\ r_{p1} & r_{p2} & \dots & r_{pn} \end{bmatrix} \end{matrix} \quad (3)$$

Dimana r_{ij} melambangkan derajat keanggotaan dari rank level v_i pada faktor evaluasi u_j .

1.2.2.4. Bobot Faktor Evaluasi

Bobot faktor evaluasi dilambangkan dengan bentuk sebagai berikut,

$$A = (a, a_1, \dots, a_p) \quad (4)$$

Dimana a adalah nilai dari bobot faktor evaluasi. Total nilai a adalah 1 sebagaimana persamaan berikut,

$$\sum_{i=1}^p a_i = 1 \quad (5)$$

Bobot faktor didapat dari perhitungan total rank dari faktor dan digunakan untuk menganalisa tingkat kepentingan dari suatu faktor.

1.2.2.5. Total Rank

Vektor evaluasi dihitung dengan komprehensif menggunakan persamaan berikut,

$$A \circ R = \begin{matrix} \begin{pmatrix} a_1 & a_2 & \dots & a_p \\ a_1 & a_2 & \dots & a_p \\ \dots & \dots & \dots & \dots \\ a_1 & a_2 & \dots & a_p \end{pmatrix} \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & \dots & \dots & \dots \\ r_{p1} & r_{p2} & \dots & r_{pn} \end{bmatrix} \end{matrix} = \begin{matrix} \begin{pmatrix} b_1 & b_2 & \dots & b_p \\ b_1 & b_2 & \dots & b_p \\ \dots & \dots & \dots & \dots \\ b_1 & b_2 & \dots & b_p \end{pmatrix} = B \end{matrix} \quad (6)$$

Dimana b , adalah derajat keanggotaan dari v_i . Nilai Hasil evaluasi dari B menjadi parameter perhitungan A (bobot tiap faktor evaluasi).

1.3. Studi Kasus Online Learning & Collaboration Learning

Pada buku ini, penerapan fuzzy matematika dicontohkan dengan perhitungan efektifitas metode belajar berbantu *online learning* dikombinasikan dengan *collaboration learning*. *Online learning* adalah sebuah metode pembelajaran kuliah yang tidak menggunakan cara-cara tradisional seperti tatap muka secara fisik di ruang kelas, dan diskusi menggunakan lisan di kelas. Sedangkan menurut (Stuqin, 2011) kuliah *online* adalah sebuah piranti lunak yang membantu belajar berbasis web yang dapat digunakan untuk menyebarkan materi belajar melalui internet. Definisi-definisi tersebut merujuk pada sebuah kondisi, dimana kondisi ini bisa diterapkan di zaman ini yang tidak mungkin dilakukan pada zaman-zaman sebelumnya. Kondisi yang dimaksud adalah berkembangnya teknologi informasi yang sangat cepat, dan pemanfaatannya di segala lini kehidupan.

Online learning dilakukan menggunakan perangkat *hardware* dan *software*, dimana setiap pelajar diwajibkan memiliki perangkat *hardware* untuk mendukung pembelajaran. Sementara *software* yang digunakan tersedia di internet, yaitu *e-learning* dan *online class room*.

Fitur yang dimanfaatkan dari *e-learning* adalah pemberian tugas online, forum online, dan pemberian materi online. Sementara fitur yang dimanfaatkan dari *online class* adalah pembelajaran sinkronus jarak jauh, rekaman, dan pengalihan presenter. Sejatinya *online classroom* adalah perangkat *online meeting*, yang dialihgunakan untuk ruang kelas online.

1.3.1. Pemetaan Masalah

Hasil yang didapat dari *online class room* dijadikan bahan perhitungan fuzzy *mathematics* adalah nilai tugas & kuis, dan hasil kuis. Kuis antara lain berisi nilai evaluasi dari mahasiswa mengenai piranti bantu mengajar, evaluasi video mengajar, konten bahan pelajaran dari dosen, dan lain sebagainya. Kuis ini akan menjadi *Evaluation factor set* dari fuzzy *mathematics*. Sementara keanggotaan *linguistic variabel* didapat dari pertimbangan nilai mahasiswa.

Berikut adalah contoh isi kuis untuk mahasiswa. Kuis berbasis skala likert.

Tabel 1. Pertanyaan Kuis

NO	PERTANYAAN	SKOR				
		1	2	3	4	5
Teknis E-Learning						
1	E-Learning mudah untuk diakses secara <i>realtime</i> dari rumah/kampus/kantor					
2	Jaringan pada environment E-Learning stabil					
3	Kesiapan untuk belajar menggunakan <i>online classroom</i> sebagai kelas tambahan kuliah					
Saran pengembangan teknis E-Learning						
Antarmuka E-Learning						
1	E-Learning mudah digunakan					
2	Fasilitas/fitur E-Learning memadai sebagai sarana belajar secara <i>realtime</i>					
3	Petunjuk penggunaan E-Learning mencukupi					
Saran pengembangan antarmuka E-Learning						
Materi E-Learning						
1	Materi yang diunggah ke E-Learning sesuai dengan tujuan perkuliahan					
2	E-learning dapat menggantikan kuliah kelas dengan seimbang					
3	Materi yang didapat di E-Learning sama dengan yang didapat pada pertemuan tatap muka					
4	Tugas yang diunggah ke E-Learning sesuai dengan materi yang diajarkan					
Saran pengembangan materi E-Learning						

Collaboration Learning		SKOR			
		1	2	3	4
1	Belajar berkelompok sangat mendukung pemahaman dalam proses belajar mengajar				
2	Memiliki semangat untuk belajar bersama teman kelompok				
3	Teman kelompok sangat mendukung untuk memperdalam materi yang dibahas				
4	Keberagaman pemikiran teman sekelompok memperkaya materi kuliah yang dibahas				
5	Dosen/fasilitator sangat berperan dalam membantu dan mengembangkan belajar kelompok				
6	Belajar kelompok sangat efektif dalam proses belajar mengajar				
7	Penerapan sistem belajar berkelompok membuat lebih cepat mengerti materi kuliah				
8	Belajar kelompok memberikan rasa puas				
Saran pengembangan Collaboration Learning					

1.3.1.1. Pemetaan Evaluation Factor

Hasil survey berbasis skala likert dijadikan komponen dalam himpunan evaluation factor berdasarkan jumlah pertanyaan dalam survey, dan normalisasi nilai evaluasi tiap pertanyaan. Persamaan normalisasi dari hasil survey ini adalah

$$u_i = 100 \cdot \frac{v_i - l_i}{m_i - l_i} \quad (7)$$

Dimana

u_i : hasil normalisasi nilai pada pertanyaan ke- i

v_i : nilai yang didapat pada pertanyaan ke- i

l_i : nilai yang mungkin didapat didapat pada pertanyaan ke- i jika semua peserta memilih skala 1 (sangat tidak setuju), atau bisa disebut juga nilai minimum yang mungkin didapat

m_i : nilai yang mungkin didapat didapat pada pertanyaan ke- i jika semua peserta memilih skala 5 (sangat setuju), atau bisa disebut juga nilai maksimum yang mungkin didapat.

Untuk mendapatkan nilai v adalah dengan menjumlahkan hasil perkalian dari banyaknya pemilih terhadap skala yang bersangkutan.

$$v = n_1 \cdot 1 + n_2 \cdot 2 + n_3 \cdot 3 + n_4 \cdot 4 + n_5 \cdot 5 \quad (8)$$

Dimana

n_1 : banyaknya pemilih skala 1

n_2 : banyaknya pemilih skala 2

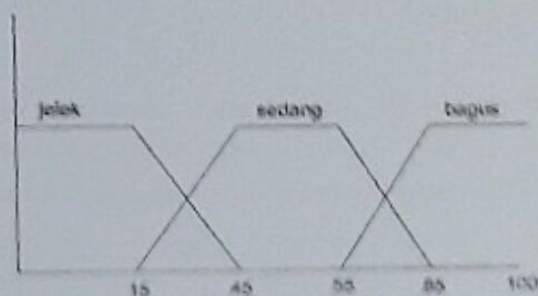
n_3 : banyaknya pemilih skala 3

n_4 : banyaknya pemilih skala 4

n_5 : banyaknya pemilih skala 5

1.3.1.2. Perhitungan Fuzzy Membership Function

Fungsi keanggotaan yang digunakan dalam studi kasus ini terdiri dari 3, yaitu jelek, sedang, dan bagus. Pertimbangan dari penggunaan 3 fungsi adalah untuk penyederhanaan penilaian akhir secara kuantitatif, berdasarkan keanggotaan dari tiap evaluation factor. Ilustrasi fungsi keanggotaan ini dapat dilihat pada gambar berikut.



Gambar 1. Menentukan Fungsi untuk hasil skala likert

Berdasarkan Gambar 1, persamaan fungsi keanggotaan jelek, sedang, dan bagus bisa dibuat. Fungsi keanggotaan jelek dihitung dengan persamaan berikut,

$$\mu_j = \begin{cases} 1 & \text{jika } x \leq 15 \\ 45 - x & \text{jika } 15 < x < 45 \\ 0 & \text{jika } x \geq 45 \end{cases} \quad (9)$$

Fungsi keanggotaan sedang dihitung dengan persamaan berikut,

$$\mu_s = \begin{cases} 0 & \text{jika } x \leq 15 \text{ atau } x \geq 85 \\ \frac{x - 15}{30} & \text{jika } 15 < x < 45 \\ \frac{85 - x}{30} & \text{jika } 45 < x < 85 \end{cases} \quad (10)$$

Fungsi keanggotaan bagus dihitung dengan persamaan berikut,

$$\mu_b = \begin{cases} 0 & \text{jika } x \leq 15 \text{ atau } x \geq 85 \\ \frac{x - 55}{30} & \text{jika } 55 < x < 85 \\ 1 & \text{jika } x \geq 85 \end{cases} \quad (11)$$

1.3.1.3 Penentuan Bobot Faktor Evaluasi

Bobot faktor evaluasi adalah sebanyak anggota evaluation factor, dalam hal ini 19. Jika semua bobotnya dijumlahkan harus bernilai 1 sebagaimana persamaan berikut

$$\sum_{i=1}^{19} \delta_i = 1 \quad (12)$$

Nilai tugas dan kuis memiliki bobot tertinggi, sebab nilai tugas dan kuis yang didapat mahasiswa, berdasarkan pertimbangan, lebih objektif dibanding hasil survey mahasiswa. Sebab kecenderungan pengisian survey dari pribadi mahasiswa memiliki tingkat subjektivitas yang tinggi, dan faktor yang mempengaruhinya ada banyak, antara lain nilai akhir yang didapat, kesiapan diri mahasiswa dalam kuliah, pengaruh dari teman, dan lain sebagainya. Nilai tugas dan kuis memiliki bobot 0,2. Sisanya, yaitu bobot masing-masing hasil survey,

$$\text{dibagi rata, yaitu } \frac{0,8}{18} = 0,044$$

Vektor bobot ini akan dikalikan dengan matrix hasil fuzzyfikasi evaluation factor, menghasilkan keanggotaan untuk nilai jelek, sedang, dan bagus terhadap penerapan hasil perkuliahan berbantu online.

$$[M_e] = [\delta] \times [M_e] \quad (13)$$

dimana

$[M_e]$: matrix hasil evaluasi

$[\delta]$: vektor bobot mendatar

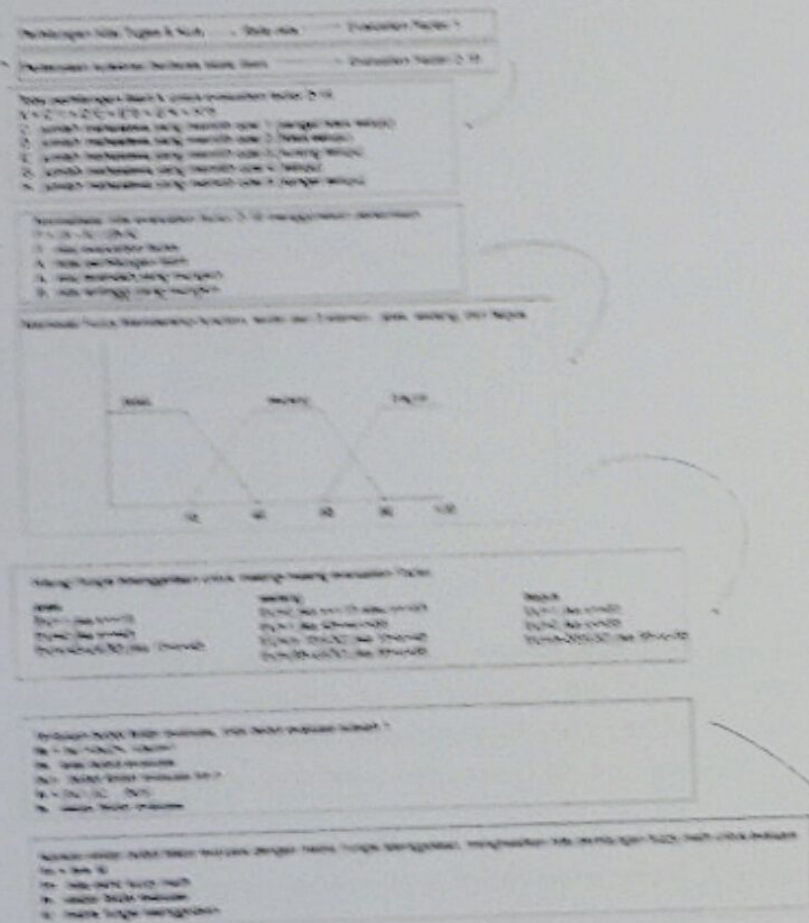
$[M_e]$: matrix fuzzyfied evaluation factor

1.3.1.4 Alur Perhitungan Fuzzy Mathematics

Perhitungan fuzzy mathematics dimulai dari perhitungan nilai rata-rata tugas dan kuis, dan perhitungan hasil kuisioner berbasis skala likert. Selanjutnya konversikan hasil kuisioner skala likert ke evaluation factor dengan menggunakan membership function dari linguistik variabel yang telah disusun. Kemudian tentukan bobot evaluasi untuk masing-masing faktor evaluasi, dan kalikan bobot evaluasi tersebut dengan faktor evaluasi. Hasil berupa

keanggotaan untuk himpunan jelek, sedang, dan Bagus. Lebih jelas mengenai alur perhitungan, bisa perhatikan gambar berikut,

Gambar 2. Alur Perhitungan Langkah-langkah dari 2016 untuk memperoleh anggota kelas fuzzy hasil



Gambar 2. Langkah-langkah perhitungan Fuzzy Mathematics

3.3.2. Pemecahan Menggunakan Fuzzy Mathematics

Dari hasil perhitungan awal, didapat nilai tugas dan kuis adalah 70.76, dan hasil tes/cuisiner adalah pada tabel berikut.

Tabel 2. Hasil Kuis/cuisiner

No Pertanyaan	SKOR				
	1	2	3	4	5
Teknis					
1	0	1	5	20	23
2	0	5	31	24	1
3	1	0	13	33	5
Antarmuka					
1	1	0	3	41	18
2	1	5	18	31	5
3	0	1	18	37	5
Manajemen Learning					
1	0	0	4	30	18
2	2	5	26	26	5
3	2	5	13	34	7
4	0	1	6	44	10
Collaboration Learning					
1	0	0	5	38	17
2	0	2	6	37	15
3	0	1	9	38	12
4	0	0	6	38	16
5	0	0	7	30	23
6	0	0	8	37	15
7	0	1	12	33	14
8	1	3	18	30	10

Menggunakan skala likert, total peserta yang memilih poin 1 – 5 adalah

Tabel 3. Jumlah pemilih untuk masing-masing pilihan

Keterangan	Pemilih
Sangat tidak Setuju	8
Tidak Setuju	36
Kurang Setuju	206
Setuju	619
Sangat Setuju	221
Total	1090

Jika dikonversikan dalam bentuk presentase adalah sebagai berikut.

Tabel 4. Jumlah pemilih masing-masing pilihan dalam bentuk persen

Keterangan	dalam %
Sangat tidak setuju	0.73
Tidak Setuju	3.28
Kurang Setuju	18.76
Setuju	56.38
Sangat Setuju	20.13

Di sini dapat disimpulkan secara sederhana, bahwa penerapan belajar metode online learning dikombinasikan dengan collaboration learning, berdasarkan skala likert, cukup efektif dan diminati oleh mahasiswa, terlihat dari jumlah pemilih setuju adalah 56.38%, sementara yang tidak setuju dan sangat tidak setuju masing-masing hanya 3.28% dan 0.73%.

1.3.2.1. Perhitungan Evaluation Factor Set

Hasil evaluasi berupa kuesioner harus dinormalisasikan terlebih dahulu sebelum defuzzifikasi. Untuk normalisasi perlu mencari kemungkinan nilai minimum dan maksimum untuk masing-masing pertanyaan. Persamaannya adalah sebagai berikut,

$$\min = 1 \cdot \sum_{i=1}^5 n_i \quad (14)$$

$$\max = 5 \cdot \sum_{i=1}^5 n_i \quad (15)$$

Dimana n_i adalah nilai ke- i yang dipilih untuk sebuah pertanyaan. Contohnya pada pertanyaan teknis 1, masing-masing pemilih 1,2,3,4,dan 5 adalah 0,1,8,29,23. Maka nilai min-nya adalah $1 \cdot (0+1+8+29+23) = 61$, dan nilai max-nya adalah $5 \cdot (0+1+8+29+23) = 305$.

Selanjutnya dihitung nilai likert untuk masing-masing pertanyaan dengan persamaan berikut,

$$\text{nilai_likert} = \sum_{i=1}^5 i \cdot n_i \quad (16)$$

Masih menggunakan contoh di atas, nilai likertnya adalah $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 8 + 4 \cdot 29 + 5 \cdot 23 = 257$. Nilai ini harus dinormalisasi sehingga bernilai antara 0-100. Berikut adalah persamaan normalisasinya,

$$\text{normalisasi_likert} = 100 \cdot \frac{\text{nilai_likert} - \min}{\max - \min} \quad (17)$$

Contoh hasil perhitungan pada kasus ini adalah

$$\text{normalisasi_likert} = 100 \cdot \frac{257 - 61}{305 - 61} = 80.33 \quad (18)$$

Untuk kasus teknis 1, didapat hasil normalisasi likert 80.33. Perhitungan ini dilakukan untuk semua pertanyaan kuesioner, didapat hasil pada tabel berikut.

Tabel 5. Hasil Normalisasi Likert

Pertanyaan	1	2	3	4	5	min	max	Nilai	Hasil Normalisasi
1	0	1	8	29	23	61	305	257	80.33
2	0	5	31	24	1	61	305	204	58.61
3	1	6	13	33	8	61	305	224	66.8

1	1	0	3	41	16	61	305	254	79.1
2	1	5	18	31	6	61	305	219	64.75
3	0	1	18	37	5	61	305	229	68.85
1	0	0	4	39	18	61	305	258	80.74
2	2	5	23	26	5	61	305	210	61.07
3	2	5	13	34	7	61	305	222	65.98
4	0	1	6	44	10	61	305	246	75.82
1	0	0	5	38	17	60	300	252	80
2	0	2	6	37	15	60	300	245	77.08
3	0		9	38	12	60	300	241	75.42
4	0	0	6	38	16	60	300	250	79.17
5	0	0	7	30	23	60	300	256	81.67
6	0	0	8	37	15	60	300	247	77.92
7	0	1	12	33	14	60	300	240	75
8	1	3	16	30	10	60	300	225	68.75

Selanjutnya, hasil normalisasi tersebut, digabungkan dengan nilai rata-rata tugas & kuis, dan dihitung fungsi keanggotaannya untuk nilai jelek, bagus, dan sedang. Fungsi keanggotaan fuzzy yang digunakan dalam perhitungan bisa dilihat pada persamaan 8,9,10.

Contoh perhitungan untuk pertanyaan teknis 1 yang hasil normalisasinya adalah 80.33 adalah sebagai berikut :

Nilai keanggotaan jelek

$\mu_j = 0$ karena nilai 80.33 lebih besar dari 45.

Nilai keanggotaan sedang

$\mu_s = \frac{85 - 80.33}{30} = 0.156$ karena nilai 80.33 berada antara 55 dan 85.

Nilai keanggotaan bagus

$$\mu_b = \frac{80.33 - 55}{30} = 0.84 \text{ karena nilai } 80.33 \text{ berada antara } 55 \text{ dan } 85$$

Seluruh faktor evaluasi dihitung keanggotaannya untuk nilai jelek, sedang, dan bagus, sehingga menghasilkan Tabel 6 berikut.

Tabel 6. Hasil evaluasi fuzzy

EVALUATION FACTORS SET	Nilai Normalisasi	Nilai Keanggotaan		
		jelek	sedang	bagus
u1 nilai rata-rata mahasiswa	70.79	0	0.47666667	0.52333333
u2 kemampuan menguasai elearning	89.14	0	0.17666667	0.82333333
u3 kestabilan jaringan elearning	58.61	0	0.87666667	0.12333333
u4 kesigapan online class room	68.8	0	0.68666667	0.31333333
u5 kemampuan menggunakan elearning	79.1	0	0.17666667	0.82333333
u6 fasilitas elearning cukup memadai	64.75	0	0.47	0.53
u7 penyuguh elearning	96.01	0	0.18666667	0.81333333
u8 bahan kuliah yang diunggah ke elearning	89.76	0	0.167	0.833
u9 elearning sebagai pengganti kuliah	61.07	0	0.79666667	0.20333333
u10 keamanan materi elearning dengan tetap muka	85.98	0	0.44	0.56
u11 tugas elearning sama dengan di kelas	75.82	0	0.308	0.692
u12 belajar kelompok mendukung pemahaman	80	0	0.18666667	0.81333333
u13 belajar kelompok menimbulkan semangat belajar	77.08	0	0.308	0.692
u14 belajar kelompok mendukung memperdalam materi	75.42	0	0.21333333	0.78666667
u15 keragaman pikiran dalam kelompok	79.17	0	0.18666667	0.81333333
u16 peran dosen/fasilitator dalam belajar kelompok	81.67	0	0.111	0.889
u17 belajar kelompok efektif dalam proses belajar mengajar	77.92	0	0.215	0.785
u18 penerapan belajar kelompok membuat cepat mengerti kuliah	75	0	0.21333333	0.78666667
u19 kepuasan belajar kelompok	68.75	0	0.34666667	0.65333333

1.3.2.2. Bobot Faktor Evaluasi dan Hasil Akhir

Sebagaimana yang telah dijelaskan pada Subbab 1.3.1.3 tentang penentuan bobot faktor evaluasi, yaitu untuk nilai tugas & kuis mendapat poin 0.2, dan sisanya dibagi rata dengan nilai 0.044, dapat dilihat pada Tabel 7 berikut.

Tabel 7. Hasil Evaluasi Fuzzy

EVALUATION FACTORS SET	Bobot Faktor Evaluasi
u1	0.2
u2	0.044444
u3	0.044444

1.5. Latihan Fuzzy Mathematics

Tentukanlah apakah sebuah kuliah sudah berjalan efektif atau belum. Nilai dari tugas & kuis kuliah tersebut adalah 70,76, dan bobot faktor evaluasinya adalah 0,1. Sementara bobot faktor evaluasi hasil kuisionernya berbagi rata. Berikut adalah hasil kuesioner dari mata kuliah tersebut.

No Pertanyaan	SKOR				
	1	2	3	4	5
Teknis					
1	0	2	13	52	28
2	0	6	45	42	2
3	2	8	26	51	8
Antarmuka					
1	1	1	8	66	19
2	1	8	27	51	8
3	0	2	29	59	5
Materi Elearning					
1	0	0	6	65	24
2	3	6	39	40	7
3	2	5	24	54	10
4	0	2	10	70	13
Collaboration Learning					
1	0	0	10	59	25
2	0	2	11	58	23
3	0	1	16	60	17
4	0	0	11	54	29
5	1	0	10	51	32
6	0	1	19	57	17
7	0	1	25	49	19

	8	2	3	28	49	12
Online Class Room						
1	0	2	4	17	5	
2	0	0	4	17	7	
3	0	0	2	22	4	
4	0	1	3	21	3	
5	0	0	5	14	9	
6	0	0	5	18	5	
7	0	2	5	19	2	
8	1	0	3	18	6	

Hitunglah :

- Jumlah pemilih untuk masing-masing pilihan : sangat tidak setuju, tidak setuju, kurang setuju, setuju, sangat setuju!
- Jumlah pemilih masing-masing pilihan dalam persen!
- Nilai skala likert dari masing-masing pertanyaan!
- Nilai keanggotaan fuzzy masing-masing evaluation factor (nilai jelek, sedang, dan bagus)!
Gunakan membership function fuzzy pada subbab 3.1.2 untuk menghitung nilai keanggotaan fuzzy.
- Hasil akhir evaluasi fuzzy!

Selanjutnya, tentukan, berdasarkan hasil akhir evaluasi fuzzy, apakah kuliah tersebut berjalan baik dan efektif? Jelaskan pula alasan Anda!

BAB 2

LINEAR VECTOR QUANTIZATION

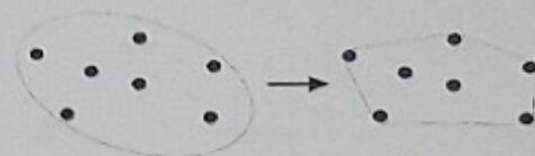
2.1. Pendahuluan

Linear Vector Quantization (LVQ) atau Kuantisasi Vektor Lajar merupakan salah satu metode klasifikasi yang berkembang dari proses clustering. Keberadaannya muncul bersamaan dengan metode clustering *SOM (Self Organizing Map)* yang diusulkan oleh Kohonen (Hu, Palreddy, & Tompkins, 1997). Perbedaannya adalah *SOM* merupakan unsupervised method, yaitu metode yang berjalan tanpa petunjuk apakah suatu data masuk ke dalam kelompok tertentu atau tidak.

SOM hanya berpegangan pada seberapa jauh perbedaan antara satu data dengan data lain. Di sini *SOM* hanya mampu mengelompokkan data ke dalam beberapa kelompok dengan meminimalisir jarak antar data dalam satu kelompok. Kuatnya ikatan data terhadap satu kelompok ditentukan dengan besarnya bobot data tersebut terhadap kelompoknya. Misalkan adalah bobot data ke- i pada waktu ke- t , maka bobot untuk waktu $t+1$ di-update sesuai dengan persamaan berikut (Hu, Palreddy, & Tompkins, 1997):

$$m_i(t+1) = m_i(t) + h_c(i) [x(t) - m_i(t)] \quad (19)$$

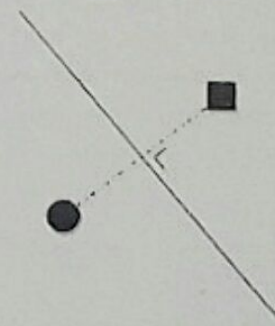
Dimana $h_c(i)$ adalah neighborhood kernel, yang merupakan ukuran ketetanggaan data ke- i dengan semua data tetangganya. Sementara itu *LVQ* merupakan metode klasifikasi *supervised*, di mana training data diberi label sesuai dengan kelasnya masing-masing. *LVQ* menggunakan prinsip *Convex Hull* dan *Diagram Voronoi*. *Convex Hull* adalah sebuah ruang lingkup vertex berbentuk polygon yang tidak memungkinkan adanya area cekung di bidang n dimensi (Berg, Cheong, Kreveld, & Overmars, 2008). Gambar 3 adalah contoh ilustrasi dari *convex hull*.



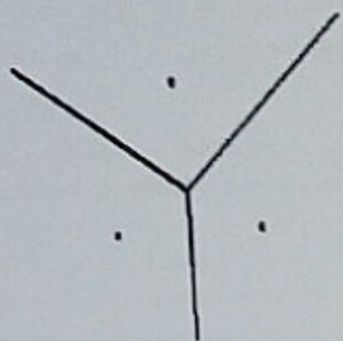
Gambar 3. Convex Hull

Di Gambar 3 terlihat vertex-vertex dalam *Convex Hull* berada dalam lingkup *polygon* yang tertutup secara ketat oleh vertex-vertex pembatas yang berada pada titik-titik sudut *polygon*. Vertex-vertex itu merupakan ilustrasi dari data input yang tersebar dalam ruang *convex*.

Diagram *Voronoi* adalah diagram yang dihimpun dari verteks penyusun *Delanay Triangle*. Verteks ini berada di dalam area diagram, yang wilayahnya berbatasan dengan verteks lain. Batasan antara 2 buah verteks berjarak sama terhadap dinding pembatasnya secara *orthogonal*. Batasan 2 buah verteks bisa dilihat pada gambar 4, ilustrasi diagram *voronoi* sendiri bisa dilihat pada gambar 5.

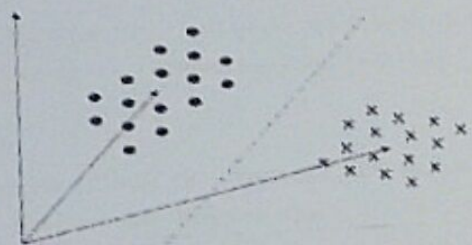


Gambar 4. Dinding pembatas 2 verteks



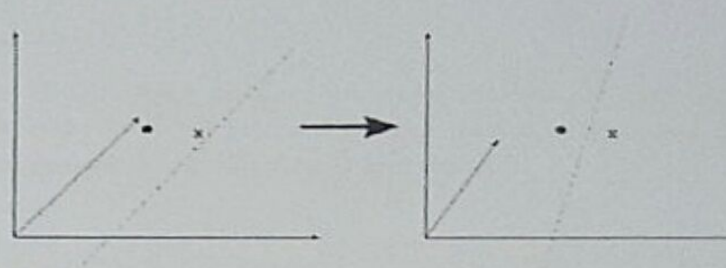
Gambar 5. Diagram Voronoi dengan 3 titik. Dibuat menggunakan (Brutell)

Ilustrasi pembagian kelas wilayah *LVQ* adalah Diagram *Voronoi* yang digambarkan pada gambar 5, dimana masing-masing verteks adalah titik pusat dari *LVQ*. Verteks pada Diagram *Voronoi* merupakan representasi dari *centroid*, atau vektor label pada *LVQ*. Verteks anggota tersebar dalam wilayah yang dikuasai oleh verteks pusat tersebut.



Gambar 6. Klasifikasi data berdasarkan vektor linear

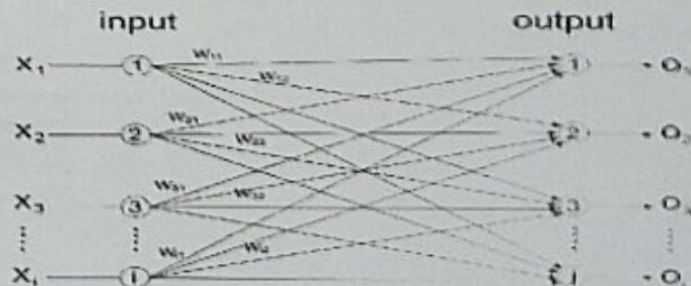
Galat klasifikasi terjadi ketika terdapat vektor data yang masuk ke dalam sebuah kelas yang bukan seharusnya. Contoh data *A* terdeteksi masuk ke kelas 1, padahal sesuai label, seharusnya masuk ke kelas 2. Untuk meminimalisir kesalahan klasifikasi, algoritma *LVQ* mengubah *boundary* antar *cluster* dari kelas yang berbeda dengan mengubah pusat *clustering*. Proses inilah yang disebut dengan *learning vector quantization (LVQ)*. Metode ini disebut "*learning vector quantization*" karena mirip dengan metode kuantisasi vektor pada kompresi sinyal di bidang pemrosesan sinyal dan komunikasi.



Gambar 7. Perubahan vector classifier pusat clustering

2.2. Arsitektur *Linear Vector Quantization*

Linear Vector Quantization bekerja dengan memetakan vektor input ke dalam kelompok yang bersesuaian dengan kelas input tersebut. Arsitektur *LVQ* berupa *graph* yang terdiri dari sejumlah *node* sebagai input, dan sejumlah *node* sebagai output. Jumlah *node input* bergantung dari banyaknya fitur komponen dari data, sementara jumlah *node output* bergantung dari jumlah kelas label yang dituju. *Node input* dan *output* terhubung oleh *edge* yang merupakan representasi dari vektor bobot. Vektor bobot inilah *centroid* dari masing-masing kelas. Nilai bobot ini akan selalu berubah sebesar *learning rate* hingga tercapai vektor yang bisa mengklasifikasikan input terhadap output yang bersesuaian dengan hasil yang mendekati sempurna. Ilustrasi dari arsitektur *LVQ* dapat dilihat pada Gambar 8.



Gambar 8. Arsitektur *Linear Vector Quantization*

Algoritma *LVQ* berjalan dengan mencari perubahan nilai bobot dan mengupdate nilai bobot pada setiap iterasi. Berikut adalah persamaan untuk mencari perubahan nilai bobot:

$$\Delta w_{ij}(t) = x_{ij} - w_{ij}(t) \quad (20)$$

Dimana x_{ij} adalah node (elemen vektor) ke- i dari data ke- k . $w_{ij}(t)$ adalah bobot dari node ke- i dan kelas ke- j pada waktu t , dan $\Delta w_{ij}(t)$ perubahan bobot untuk node ke- i dan kelas ke- j pada waktu t .

Setelah dilakukan perhitungan perubahan bobot, selanjutnya bobot di-update sesuai dengan perubahan yang didapat dikalikan dengan laju pembelajaran (*learning rate*) yang sudah ditetapkan. Laju pembelajaran merupakan koefisien yang bisa diubah oleh pengguna untuk kalibrasi kecepatan belajar dan akurasi. Semakin besar laju pembelajaran, semakin cepat proses belajar namun semakin rendah akurasi. Laju pembelajaran yang umum digunakan adalah 0.1. Berikut adalah persamaan untuk meng-update nilai bobot:

$$w_{ij}(t+1) = w_{ij}(t) \pm \alpha \cdot \Delta w_{ij} \quad (21)$$

dimana $w_{ij}(t)$ adalah bobot dari node ke- i dan kelas ke- j pada waktu t , $w_{ij}(t+1)$ adalah bobot dari node ke- i dan kelas ke- j pada waktu $t+1$, α adalah laju pembelajaran, dan $\Delta w_{ij}(t)$ adalah perubahan bobot untuk node ke- i dan kelas ke- j pada waktu t .

Dalam Algoritma LVQ, Perhitungan bobot terus dilakukan selama iterasi, dan berakhir ketika batas iterasinya tercapai, atau bobot sudah stabil.

2.3. Sample Case

Misalkan ada 12 buah data yang ingin dikelompokkan ke dalam 2 kelas. Dari keseluruhan data tersebut 6 data digunakan sebagai data latih, dan 6 data sebagai data uji. Sementara class terbagi dua, yaitu kelas X dan kelas O. Data latih dan uji hanya masuk ke dalam salah satu class di antara 2. Tidak ada data yang masuk ke dalam 2 kelas secara bersamaan. Berikut adalah kelima data latih beserta labelnya

Data Latih 1 : (1,3) / X

Data Latih 2 : (6,8) / O

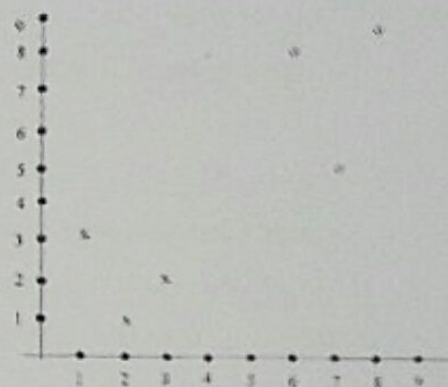
Data Latih 3 : (2,1) / X

Data Latih 4 : (7,5) / O

Data Latih 5 : (3,2) / X

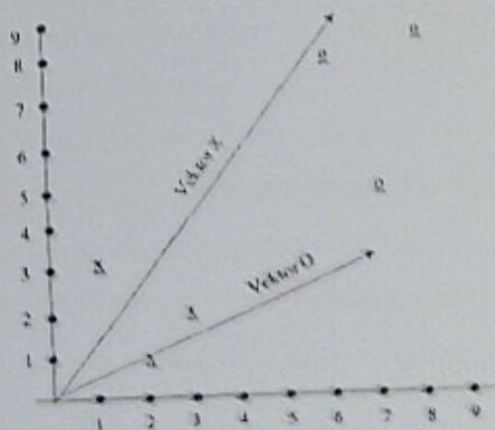
Data Latih 6 : (8,9) / O

Jika digambarkan dalam koordinat kartesius, persebaran data tersebut dapat dilihat pada Gambar 9. Data latih yang masuk ke class X dilambungkan dengan label x, dan data latih yang masuk ke class O dilambungkan dengan label o.



Gambar 9. Data Latih

Kemudian tentukan vektor linear yang mewakili masing-masing class. Untuk mempermudah penamaan berilah label vektor tersebut dengan vektor X dan vektor O. Saat ini penentuan nilai vektor dilakukan dengan sembarang. Misalkan vektor X adalah (6,9) dan vektor Y adalah (7,3). Penentuan nilai vektor untuk mengoptimalkan proses *learning* diberikan pada subbab berikutnya. Ilustrasi dari vektor X dan vektor O di persebaran data latih dapat dilihat pada Gambar 10.



Gambar 20 Ilustrasi vektor X dan O

Selanjutnya, mulailah perhitungan untuk memposisikan ulang vektor X dan O. Tentukan *learning rate* untuk kecepatan proses pembelajaran. Dalam hal ini *learning rate* adalah seberapa jauh vektor X dan O berubah posisi selama jalannya algoritma. Lakukan perhitungan jarak vektor X terhadap semua data X, dan vektor Y terhadap semua data Y. Setiap kali jarak didapat, vektor dipindahkan sejauh *learning rate* ke arah data tersebut. Lakukan perhitungan dalam iterasi hingga batas iterasi terpenuhi, atau pengujian klasifikasi telah memenuhi batas galat yang ditoleransi. Pada setiap iterasi untuk setiap vektor lakukan tahap berikut:

1. Hitung selisih antara data dengan vektor *class*, dapatkan delta vektor *class* (persamaan 20).
2. Tambahkan vektor *class* dengan delta yang sudah dikalikan *learning rate* (persamaan 21).
3. Gunakan vektor hasil perhitungan langkah 2 untuk menghitung perubahan dengan data berikutnya.
4. Setelah 1 iterasi selesai, lakukan pengujian hasil klasifikasi menggunakan data training, dan lihat prosentase keberhasilan klasifikasinya. Pengujian dengan mengukur kedekatan data latih dengan vektor *class* menggunakan *eulidean distance*. Jika sudah lebih kecil dari toleransi, iterasi dihentikan, dan didapatkan vektor *class* yang sesuai dengan data *training*.

Learning Rate : 0.1

Error Tolerance : 0.1

Iterasi 1

Perhitungan vektor X

Delta vektor X terhadap data latih 1 : $(1,3) - (6,9) = (-5,-6)$

Perubahan vektor X : $(6,9) + 0.1 * (-5,-6) = (6,9) + (-0.5,-0.6) = (5.5,8.4)$

Bawalah hasil perhitungan di atas ke perhitungan selanjutnya

Delta vektor X terhadap data latih 3 : $(2,1) - (5.5,8.4) = (-3.5,-7.4)$

Perubahan vektor X : $(5.5,8.4) + 0.1 * (-3.5,-7.4) = (5.5,8.4) + (-0.35,-0.74) = (5.15,7.66)$

Bawalah hasil perhitungan di atas ke perhitungan selanjutnya

Delta vektor X terhadap data latih 5 : $(1,3) - (5.15,7.66) = (-4.15,-4.66)$

Perubahan vektor X : $(5.15,7.66) + 0.1 * (-4.15,-4.66) = (5.15,7.66) + (-0.415,-0.466) = (4.735,7.194)$

Perhitungan vektor O

Delta vektor O terhadap data latih 2 : $(6,8) - (7,3) = (-1,5)$

Perubahan vektor O : $(7,3) + 0.1 * (-1,5) = (7,3) + (-0.1,0.5) = (6.9,3.5)$

Bawalah hasil perhitungan di atas ke perhitungan selanjutnya

Delta vektor O terhadap data latih 4 : $(7,5) - (6.9,3.5) = (0.1,1.5)$

Perubahan vektor O : $(6.9,3.5) + 0.1 * (0.1,1.5) = (6.9,3.5) + (0.01,0.15) = (6.91,3.65)$

Bawalah hasil perhitungan di atas ke perhitungan selanjutnya

Delta vektor O terhadap data latih 6 : $(8,9) - (6.91,3.65) = (1.09,5.35)$

Perubahan vektor O : $(6.91,3.65) + 0.1 * (1.09,5.35) = (6.91,3.65) + (0.109,0.535) = (7.019,4.185)$

Perhitungan galat

Total data adalah 6, sehingga galat bernilai $n/6$ dimana n adalah banyaknya kesalahan dalam estimasi pengelompokan berdasarkan jarak.

Perhitungan bermula dari jumlah kesalahan=0.

Jarak antara vektor X dengan data latih 1: Euclidean distance ((4.735,7.194), (1,3))

$$= \sqrt{(4.735 - 1)^2 + (7.194 - 3)^2} = \sqrt{13.950225 + 17.589636} = \sqrt{31.539861} = 5.62$$

Jarak antara vektor O dengan data latih 1: ((7.019,4.185), (1,3))

$$= \sqrt{(7.019 - 1)^2 + (4.185 - 3)^2} = \sqrt{36.228361 + 1.404225} = \sqrt{37.632586} = 6.13$$

Berdasarkan jarak, data latih 1 lebih dekat ke vektor X dibanding vektor O, sehingga data latih 1 masuk ke vektor X. Hal ini sesuai dengan label sebenarnya dari data latih 1 (yaitu X), sehingga nilai kesalahan tetap 0.

Jarak antara vektor X dengan data latih 2: Euclidean distance ((4.735,7.194), (6,8))

$$= \sqrt{(4.735 - 6)^2 + (7.194 - 8)^2} = \sqrt{1.600225 + 0.649636} = \sqrt{2.249861} = 1.5$$

Jarak antara vektor O dengan data latih 2: ((7.019,4.185), (6,8))

$$= \sqrt{(7.019 - 6)^2 + (4.185 - 8)^2} = \sqrt{36.228361 + 1.404225} = \sqrt{37.632586} = 6.13$$

Berdasarkan jarak, data latih 2 lebih dekat ke vektor X dibanding vektor O, sehingga data latih 1 masuk ke vektor X. Hal ini bertentangan dengan label yang seharusnya, yaitu masuk ke vektor O, sehingga nilai kesalahan bertambah 1 menjadi 1.

Jarak antara vektor X dengan data latih 3: Euclidean distance ((4.735,7.194), (2,1))

$$= \sqrt{(4.735 - 2)^2 + (7.194 - 1)^2} = \sqrt{7.480225 + 38.365636} = \sqrt{45.845861} = 6.77$$

Jarak antara vektor O dengan data latih 3: ((7.019,4.185), (2,1))

$$= \sqrt{(7.019 - 2)^2 + (4.185 - 1)^2} = \sqrt{25.190361 + 10.144225} = \sqrt{35.334586} = 5.94$$

Berdasarkan jarak, data latih 3 lebih dekat ke vektor X dibanding vektor O, sehingga data latih 3 masuk ke vektor X. Hal ini sesuai dengan label sebenarnya dari data latih 3 (yaitu X), sehingga nilai kesalahan tetap 1.

Lakukan perhitungan nilai kesalahan, kemudian rata-ratakan nilai kesalahan tersebut. Jika di bawah batas ambang *error tolerance*, hentikan proses pembelajaran, dan vektor class telah terbentuk.

2.4. Optimalisasi Inisialisasi Bobot Vektor

Untuk mempercepat proses learning LVQ, kita bisa menginisiasi nilai awal dari vektor *class*. Dalam hal ini vektor *class* ditentukan dari data latih yang mewakili vektor *class* tersebut. Persamaan untuk inisialisasi bobot vektor *class* berdasarkan data training adalah sebagai berikut:

$$v_{c,d} = \frac{\sum_{n=1}^n a_{n,c,d}}{n} \quad (22)$$

Dimana $v_{c,d}$ adalah vektor *class* c untuk dimensi ke- d . Kemudian $a_{n,c,d}$ adalah komponen data latih ke- n untuk *class* c dimensi ke- d .

Pada studi kasus subbab 2.3, untuk mencari inisial vektor *class* X adalah sebagai berikut. Untuk vektor *class* X :

$$v_x = \frac{\text{Data latih 1} + \text{Data latih 3} + \text{Data latih 5}}{3}$$

$$v_x = \frac{(1,3) + (2,1) + (3,2)}{3} = \frac{(6,6)}{3} = (2,2)$$

Kemudian untuk vektor *class* O :

$$v_o = \frac{\text{Data latih 2} + \text{Data latih 4} + \text{Data latih 6}}{3}$$

$$v_o = \frac{(6,8) + (7,5) + (8,9)}{3} = \frac{(21,22)}{3} = (7,7.33)$$

2.5. Ringkasan LVQ

LVQ atau *Linear Vector Quantization* merupakan salah satu metode klasifikasi dalam sistem pakar berbasis *supervised learning*. LVQ memiliki beberapa komponen perhitungan, antara lain vektor linear, data latih, data uji, *learning rate*, dan *error tolerance*. LVQ mampu mengklasifikasikan data berdasarkan fitur-fiturnya, dan sangat bagus dan cepat untuk data yang memiliki garis pemisah fitur berupa bidang linear. Untuk mempercepat proses perhitungan, class LVQ perlu diinisialisasi dengan sebuah formula sehingga membentuk vektor *class* yang mendekati hasil pembelajaran akhir.

2.6. Latihan LVQ

Pada studi kasus subbab 2.3, silakan lakukan proses learning LVQ dengan inialisasi vektor class pada subbab 2.4. Perhatikan, berapa kali looping dibutuhkan untuk mencapai hasil latih dengan tingkat *error rate* di bawah 0.1, dan bagaimana hasil pembagian class pada data uji menggunakan hasil pembelajaran tersebut!

DAFTAR PUSTAKA

- Beutel, A. (n.d.). Interactive Voronoi Diagram Generator with WebGL. *Interactive Voronoi Diagram Generator with WebGL*. Retrieved from <http://alexbeutel.com/webgl/voronoi.html>
- Hu, Y. H., Palreddy, S., & Tompkins, W. J. (1997, September). A patient-adaptable ECG beat classifier using a mixture of experts approach. *IEEE Transactions on Biomedical Engineering*, 44(9), 891-900.
- Kresnha, P. E., Susilowati, E., & Ambo, S. N. (2016). Analisis Efektifitas Kuliah Berbasis Online Untuk Meningkatkan Kemajuan Belajar Mahasiswa Di Jurusan Informatika Universitas Muhammadiyah Jakarta. *Konferensi Nasional Teknologi Informasi dan Komunikasi (KNASTIX)* (pp. 14-19). Yogyakarta: Universitas Kristen Duta Wacana.
- Shujin, M. (2011). Online Course Quality Factor Analysis Based on Fuzzy Mathematics. *IT in Medicine and Education (ITME)* (pp. 344-346). Cuangzhou: IEEE.
- Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control* 8, 338-353.

SISTEM PAKAR

STUDI KASUS DAN PEMECAHANNYA

Fuzzy merupakan salah satu cabang ilmu komputer, lebih tepatnya bidang expert system. Fuzzy digunakan untuk pemecahan masalah secara cepat dan relatif tepat, sebagaimana manusia menentukan keputusan dalam masalah yang sedang dihadapi. Tidak harus tepat 100%, namun hasil cukup memuaskan, dan masih berada di wilayah "bisa diterima". Dalam perkembangan selanjutnya, setelah fuzzy bisa diterima di dunia kecerdasan buatan, muncul berbagai teori tentang fuzzy. Dimulai dari sekedar perubahan himpunan crisp menjadi himpunan yang didalamnya terdapat elemen yang memiliki tingkat keanggotaan tertentu terhadap himpunan tersebut, hingga modifikasi berbagai metode yang dimasukkan fuzzy didalamnya, seperti FLVQ (Fuzzy Linear Vector Quantization), FNN (Fuzzy Neural Network), FGA (Fuzzy Genetic Algorithms), dan sebagainya.

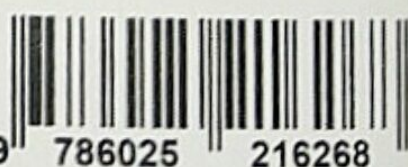
Pada pembahasan buku ini, Anda akan mempelajari mengenai sejarah singkat logika fuzzy dan cara penyelesaian masalah menggunakan fuzzy mathematics, dengan studi kasus penilaian evaluasi belajar kuliah berbasis online dan collaboration learning. Pada buku ini juga dibahas tentang Linear Vector Quantization (LVQ) yang merupakan salah satu metode klasifikasi dalam sistem pakar berbasis supervised learning. LVQ memiliki beberapa komponen perhitungan, antara lain vektor linear data latih, data uji, learning rate, dan error tolerance. LVQ mampu mengklasifikasikan data berdasarkan fiturnya, dan sangat bagus dan cepat untuk data yang memiliki garis pemisah fitur berupa bidang linear. Untuk mempercepat proses perhitungan, class LVQ perlu diinisialisasi dengan sebuah formula sehingga membentuk vektor class yang mendekati hasil pembelajaran akhir.



Canting Mas Anyar

Perum. Agatama Regency Banguntapan A8
Yogyakarta. Telp. 082116448037

ISBN 978-602-52162-6-8



9 786025 216268