TUGAS AKHIR – KS 141501

IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS : PT. ALIF DUTA PERSADA).

IMPLEMENTATION OF K-MEANS CLUSTERING AND LRFM MODEL FOR CUSTOMER SEGMENTATION (CASE STUDY : PT. ALIF DUTA PERSADA)

ADAM ARDIANSYAH
NRP 052114 40000 026

Dosen Pembimbing
Edwin Riksakomara, S.Kom., M.T.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

TUGAS AKHIR – KS141501

IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS : PT. ALIF DUTA PERSADA)

Adam Ardiansyah
NRP 05211440000026

Dosen Pembimbing :
Edwin Riksakomara, S.Kom, M.T.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

FINAL PROJECT – KS141501

IMPLEMENTATION OF K-MEANS CLUSTERING AND LRFM MODEL FOR CUSTOMER SEGMENTATION (CASE STUDY : PT. ALIF DUTA PERSADA)

Adam Ardiansyah
NRP 05211440000026

Supervisor :
Edwin Riksakomara, S.Kom, M.T.

DEPARTMENT OF INFORMATION SYSTEMS
Faculty of Information and Communication Technology
Sepuluh Nopember Institute of Technology
Surabaya 2018

LEMBAR PENGESAHAN
IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS PT. ALIF DUTA PERSADA)

TUGAS AKHIR
Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

ADAM ARDIANSYAH
NRP. 05211440000026

Surabaya, 17 Juli 2018

KEPALA
DEPARTEMEN SISTEM INFORMASI

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 196503101991021001

LEMBAR PERSETUJUAN

IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS PT. ALIF DUTA PERSADA)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:
ADAM ARDIANSYAH
NRP. 05211440000026

Disetujui Tim Penguji: Tanggal Ujian : 05 Juli 2018
Periode Wisuda : September 2018

Edwin Riksakomara, S.Kom., M.T. **(Pembimbing I)**

Ahmad Mukhlason, S.Kom, M.Sc, Ph.D. **(Penguji I)**

Faizal Mahananto, S.Kom., M. Eng., Ph.D. **(Penguji II)**

IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM MODEL UNTUK SEGMENTASI PELANGGAN (STUDI KASUS : PT. ALIF DUTA PERSADA)

Nama Mahasiswa : Adam Ardiansyah
NRP : 05211440000026
Departemen : Sistem Informasi
Pembimbing I : Edwin Riksakomara, S.Kom, M.T.

ABSTRAK

Persaingan merupakan hal yang tidak bisa dihindari, kerugian dan keuntungan akan selalu ada dalam dunia bisnis. Pihak yang mampu menguasai persaingan akan terus mendapat keuntungan dan yang tersisih akan sulit mempertahankan keberlangsungannya. Hal ini memaksa perusahaan untuk lebih bisa menarik pelanggan dan mempertahankannya agar mampu melanjutkan bisnisnya. Salah satu metode agar dapat unggul dalam hal tersebut adalah dengan mengimplementasikan CRM (customer relationship management) agar perusahaan dapat lebih baik mengenali dan mempertahankan pelanggan mereka.

PT. Alif Duta Persada, perusahaan distributor PT. Unilever Indonesia, menyadari hal tersebut sehingga merasa perlu untuk memanfaatkan segmen-segmen pelanggannya. Namun proses segmentasi yang berjalan tidak optimal karena dilakukan secara manual tiap 1-2 bulan sekali pada seribu lebih pelanggannya. Bahkan untuk penentuan pelanggan yang berhak menerima pembagian bonus selalu diserahkan pada sales-sales mereka, yang berarti hal tersebut akan sangat bersifat subjektif. Hal-hal tersebut memicu buruknya penerapan CRM perusahaan, sehingga menjadi tidak tepat sasaran dan hasilnya kurang optimal.

Untuk menyelesaikan masalah tersebut solusi yang dapat dilakukan adalah melakukan segmentasi pelanggan menggunakan metode clustering secara otomatis. Segmentasi dilakukan menggunakan metode clustering K-means, dengan menggunakan metode Elbow untuk menentukan nilai K yang tepat. Dalam proses clustering digunakan variabel LRFM sebagai tolak ukur dalam penggambaran perilaku pelanggan saat melakukan transaksi. Dan selanjutnya hasil clustering tersebut divisualisasikan agar memudahkan dalam menganalisa hasil segmentasi.

Penelitian menghasilkan dua segmen pelanggan. Jumlah dua segmen ini diperoleh dari keluaran metode Elbow. Hasil clustering diperoleh berdasarkan bobot yang telah ditentukan dari pihak perusahaan untuk keempat variabel. Hasil nilai CLV (Customer Lifetime Value) untuk segmen 1 adalah 0.138, sedangkan untuk segmen 2 adalah 0.250. Tingkat akurasi yang didapatkan berdasarakan SSE (Sum Squared Error) untuk nilai K=2 adalah 1.546808.

Dengan adanya segmentasi ini diharapkan bisa memudahkan PT. Alif Duta Persada untuk melakukan segmentasi pelanggan secara lebih efisien dan dapat menentukan pembagian bonus pada pelanggan berdasarkan performa pelanggan.

***Kata kunci* : Segmentasi Pelanggan, *clustering*, K-means, model LRFM**

IMPLEMENTATION OF K-MEANS CLUSTERING AND LRFM MODEL FOR CUSTOMER SEGMENTATION (CASE STUDY : PT. ALIF DUTA PERSADA)

Student Name	**: Adam Ardiansyah**
NRP	**: 05211440000026**
Department	**: Sistem Informasi**
Supervisor I	**: Edwin Riksakomara, S.Kom, M.T.**

ABSTRACT

Competition is inevitable in business, losses and benefits will always coexist in the business world. The side that capable of mastering the competition will continue to get the benefit and those who get left behind will find it difficult to maintain their sustainability. This situation forces the company to be better at attract customers and keeps them in order to be able to survive. One method to excel in this case is to implement CRM (customer relationship management) so that companies can better understand and retain their customers.

PT. Alif Duta Persada, a distributor company of PT. Unilever Indonesia, realizes this so that it feels the need to take advantage of its customer segments. But the present segmentation process is not optimal because all this time the process was done manually by human for every 1-2 months. Moreover the process to determine the customers who are eligible to receive a bonus was always assigned to their salesperson, which means it will be very subjective. These things trigger a poor corporate CRM implementation, so that the impact is program's target is not exactly right and the results are less than optimal.

The solution to solve the problem is to segment the customer using the clustering method automatically. Segmentation is done using K-means clustering method, by using Elbow method to determine the right value of K. In the clustering process LRFM variable was used as a benchmark to describe the customer behavior when making transactions. And after that the clustering results are visualized in order to facilitate the analyzing process.

This research produces two customer segments. The number of segments is obtained from the Elbow method's output. The clustering results are obtained based on the predetermined weight of the company for the four variables mentioned earlier. The CLV (Customer Lifetime Value) result for segmen 1 is 0.138, and for segmen 2 is 0.250. The accuracy level obtained based on SSE (Sum Squared Error) for value K=2 is 1.546808.

This segmentation's result is expected to facilitate PT. Alif Duta Persada to segment customers more efficiently and can determine the distribution of bonuses to customers based on customer performance.

Keyword: customer segmentation, clustering, K-means, LRFM model

KATA PENGANTAR

Puji dan syukur penulis tuturkan ke hadirat Allah SWT, Tuhan Semesta Alam yang telah memberikan kekuatan dan hidayah-Nya kepada penulis sehingga penulis mendapatkan kelancaran dalam menyelesaikan tugas akhir dengan judul: **IMPLEMENTASI *K-MEANS CLUSTERING* DAN MODEL LRFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS : PT. ALIF DUTA PERSADA)**, yang merupakan salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya.

Terima kasih penulis sampaikan kepada pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa materiil maupun moril demi tercapainya tujuan pembuatan tugas akhir ini. Tugas akhir ini tidak akan terwujud tanpa bantuan dan dukungan dari berbagai pihak yang sudah meluangkan waktu, tenaga dan pikirannya. Secara khusus penulis ingin menyampaikan ucapan terima kasih yang sebanyak-banyaknya kepada:

1. Allah SWT yang telah menganugerahkan rahmat dan berkahnya sehingga Tugas Akhir ini dapat diselesaikan.
2. Kedua orang tua, adik, kakak, dan seluruh keluarga yang senantiasa memberikan dukungan materi, semangat dan doa untuk penulis.
3. Pihak PT. Alif Duta Persada Probolinggo, yang telah memberikan kesempatan bagi penulis untuk melaksanakan penelitian Tugas Akhir.
4. Bapak Dr. Ir. Aris Tjahyanto, M.Kom, selaku Kepala Departemen Sistem Informasi FTIK ITS, yang telah menyediakan dukungan untuk penelitian mahasiswa.

5. Bapak Edwin Riksakomara, S.Kom., M.T. selaku dosen pembimbing yang telah meluangkan waktunya untuk memberi masukan, arahan dan bimbingan untuk penelitian Tugas Akhir ini.
6. Seluruh Bapak dan Ibu dosen pengajar beserta karyawan Departemen Sistem Informasi, FTIK ITS Surabaya yang telah memberikan ilmu, pengalaman berharga dan bantuan kepada penulis selama masa perkuliahan.
7. Rekan-rekan pejuang wisuda 118 laboratorium RDIB yang memberikan bantuan dan semangat.
8. Teman-teman OSIRIS yang menjadi keluarga tempat berbagi susah dan senang selama penulis menempuh perkuliahan.
9. Berbagai pihak yang tidak bisa disebutkan satu persatu yang telah turut serta menyukseskan penulis dalam menyelesaikan tugas akhir.

Penyusunan laporan ini masih jauh dari kata sempurna sehingga penulis menerima adanya kritik maupun saran yang membangun untuk perbaikan di masa yang akan datang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

Surabaya, Juli 2018
Penulis,

Adam Ardiansyah

DAFTAR ISI

ABSTRAK v
ABSTRACT vii
KATA PENGANTAR ix
DAFTAR ISI xi
DAFTAR GAMBAR xvi
DAFTAR TABEL xviii
DAFTAR SCRIPT xx
1. BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 4
1.3 Batasan Masalah 4
1.4 Tujuan 5
1.5 Manfaat 5
1.6 Relevansi 6
2. BAB II TINJAUAN PUSTAKA 7
2.1 Studi Sebelumnya 7
2.2 Dasar Teori 9
2.2.1 PT. Alif Duta Persada 10
2.2.2 Customer Relationship Management 10
2.2.3 Segmentasi Pelanggan 11
2.2.4 Penggalian Data 12
2.2.5 Clustering 13
2.2.6 K-Means 14

2.2.7 Model LRFM 16
2.2.8 Min-max normalization 17
2.2.9 Metode *Elbow* 18
2.2.10 Analytical Hierarchy Process (AHP) 19
2.2.11 Customer Lifetime Value 22
2.2.12 Sum Squared Error (SSE) 23
3. BAB III METODOLOGI PENELITIAN 25
3.1. Metodologi 25
3.1.1 Identifikasi Masalah 26
3.1.2 Studi Literatur 26
3.1.3 Pengumpulan Data 27
3.1.4 Praproses Data 27
3.1.5 Pembobotan LRFM 29
3.1.6 Penghitungan CLV 29
3.1.7 Clustering Data 29
3.1.8 Pengujian Kualitas *Cluster* 30
3.1.9 Visualisasi *Cluster* 30
3.1.10 Analisis Segmentasi 30
3.1.11 Penyusunan Laporan Tugas Akhir 31
4. BAB IV PERANCANGAN 33
4.1. Pengumpulan Data 33
4.2 Praproses Data 33
4.3 Pembobotan LRFM 34
4.4 Penentuan jumlah *cluster* 35

4.5 Rancangan visualisasi 35
4.5.1 Use Case Diagram 35
4.5.2 Use Case Description 36
4.5.3 Desain Antarmuka 37
5. BAB V IMPLEMENTASI 42
5.1 Pra Proses Data 42
5.1.1 Data Cleaning 42
5.1.2 Pemodelan LRFM 44
5.1.3 Normalisasi 46
5.2 Pembobotan LRFM 48
5.3 Perhitungan CLV 49
5.4 Proses Clustering 49
5.4.1 Penentuan Jumlah K 50
5.4.2 Clustering dengan K-means 52
5.5 Pengujian Kualitas *Clustering* 52
5.6 Pembuatan Visualisasi 54
5.6.1 Halaman Pertama 54
5.6.2 Halaman Visualisasi 3 Dimensi 58
5.6.3 Halaman Visualisasi Lokasi 61
6 BAB VI HASIL PEMBAHASAN 65
6.1 Pembobotan LRFM dengan AHP 65
6.2 Perhitungan CLV 68
6.3 Proses *Clustering* 69
6.3.1 Penentuan Nilai K 69

6.3.2 *Clustering K-means* 71
6.3.3 Pengujian Kualitas *Clustering* 71
6.3.4 Hasil *Cluster* 72
6.4 Analisis Hasil Clustering 72
6.4.1 Analisis Karakteristik Segmen 1 73
6.4.2 Analisis Karakteristik Segmen 2 74
6.4.3 Perbandingan Karakteristik Segmen 75
6.5 Analisis Visualisasi Clustering 75
6.5.1 Grafik *Bar Chart* 76
6.5.2 Grafik *Radar* 77
6.5.3 Grafik *Box-plot* 78
6.5.4 Grafik *Scatter Plot* 2D 82
6.5.5 Grafik *Scatter Plot* 3D 88
6.5.6 Grafik *Stacked Bar* 91
6.5.7 Grafik *Donut Chart* 93
6.6 Analisis Perbandingan 93
6.6.1 Perbandingan Jumlah Segmen Dengan Bobot 94
6.6.2 Perbandingan Jumlah Segmen Tanpa Bobot ..97
6.7 Kesimpulan Analisis 100
7. BAB VII KESIMPULAN DAN SARAN 103
7.1. Kesimpulan 103
7.2. Saran 104
DAFTAR PUSTAKA 105
BIODATA PENULIS 109

LAMPIRAN A A-1
LAMPIRAN B B-1
LAMPIRAN C C-1
LAMPIRAN D D-1
LAMPIRAN E E-1
LAMPIRAN F..... F-1

DAFTAR GAMBAR

Gambar 2.1 Grafik Metode Elbow 18
Gambar 2.2 Model Hierarkis 20
Gambar 3.1 Metodologi Pengerjaan 25
Gambar 4.1 Use Case Diagram 36
Gambar 4.2 Halaman Utama 38
Gambar 4.3 Halaman Pertama Visualisasi 39
Gambar 4.4 Halaman Visualisasi 3D 40
Gambar 4.5 halaman Visualisasi Lokasi 41
Gambar 6.1 hasil metode Elbow 71
Gambar 6.2 Bar Chart Anggota Segmen 76
Gambar 6.3 Perbandingan Nilai 4 variabel 77
Gambar 6.4 Grafik Radar 78
Gambar 6.5 Box-plot Length 79
Gambar 6.6 Box-plot Recency 80
Gambar 6.7 Box-plot Frequency 81
Gambar 6.8 Box-plot Monetary 82
Gambar 6.9 Scatter 2D LR 83
Gambar 6.10 Scatter Plot 2D LF 84
Gambar 6.11 Scatter Plot LM 85
Gambar 6.12 Scatter Plot RF 86
Gambar 6.13 Scatter Plot RM 87
Gambar 6.14 Scatter Plot FM 88
Gambar 6.15 Scatter Plot LRF 89
Gambar 6.16 Scatter Plot LRM 90
Gambar 6.17 Scatter Plot LFM 90
Gambar 6.18 Scatter Plot RFM 91
Gambar 6.19 Stacked Bar Lokasi Cluster 92
Gambar 6.20 Donut Chart Lokasi Cluster 93
Gambar 6.21 Persebaran Lokasi Pelanggan 2 segmen 95
Gambar 6.22 Persebaran Lokasi 3 Segmen 96
Gambar 6.23 Persebaran Lokasi 4 Segmen 96

Gambar 6.24 Persebaran 3 Segmen Tanpa Bobot.....99
Gambar 6.25 Persebaran 4 Segmen Tanpa Bobot.....99
Gambar 6.26 Persebaran 5 Segmen Tanpa Bobot.....100

DAFTAR TABEL

Tabel 2.1 Penelitian Sebelumnya 7
Tabel 2.2 Penjabaran Variabel LRFM.....16
Tabel 2.3 Preferensi Penilaian Pairwise Comparison.....20
Tabel 2.4 Random Index 22
Tabel 4.1 Use Case Description melihat tabel data hasil clustering 36
Tabel 5.1 Penjelasan Atribut 42
Tabel 5.2 Sebagian Hasil Pemodelan LRFM 46
Tabel 5.3 Rentang Nilai Atribut 47
Tabel 6.1 Hasil Kuesioner Responden 1 65
Tabel 6.2 Hasil Kuesioner Responden 2 65
Tabel 6.3 Hasil Kuesioner Responden 3 65
Tabel 6.4 Hasil Rata-Rata Penilaian Responden.....66
Tabel 6.5 Normal Comparison Matrix 66
Tabel 6.6 Weighted Sum Vector 67
Tabel 6.7 Consistency Vector.....67
Tabel 6.8 Sebagian Hasil Perhitungan CLV Pelanggan 68
Tabel 6.9 Nilai Error Setiap K.....69
Tabel 6.10 Perbandingan Error.....70
Tabel 6.11 Contoh Hasil Clustering 72
Tabel 6.12 Karakteristik Segmen 1 73
Tabel 6.13 Karakteristik Segmen 2 74
Tabel 6.14 Perbandingan Karakteristik Segmen 75
Tabel 6.15 Perbandingan Jumlah Anggota.....94
Tabel 6.16 Karakteristik Segmen Setiap K 94
Tabel 6.17 Perbandingan Jumlah Anggota.....97
Tabel 6.18 Karakteristik Segmen tanpa Bobot.....97
Tabel B.1 Use Case Melihat visualisasi grafik bar chart, radar chart, dan box plot B-1
Tabel B.2 Use Case Melihat visualisasi grafik scatter plot 3D B-2

Tabel B.3 Use Case Melihat visualisasi data pelanggan berdasarkan lokasi (grafik stacked bar dan donut chart) B-2
Tabel C.1 Pemodelan LRFM C-1
Tabel D.1 Tabel Normalisasi LRFM..... D-1
Tabel E.1 Tabel Normalisasi E-1

DAFTAR SCRIPT

script 5.1 Algoritma Ekstraksi Data 45
script 5.2 Normalisasi Data 48
script 5.3 Menghitung CLV.....49
script 5.4 Elbow Method 51
script 5.5 K-means Clustering.....52
script 5.6 SSE 53
script 5.7 Grafik Bar Chart.....55
script 5.9 Radar Chart.....56
script 5.10 Grafik Box-plot 58
script 5.11 Scatter Plot 3D.....61
script 5.12 Menentukan Kecamatan Tiap Pelanggan 62
script 5.13 Visualisasi Stacked Bar 63
script 5.14 Visualisasi Donut Chart.....64
script F.1 Script kode Keseluruhan F-1

BAB I
PENDAHULUAN

Pada bab pendahuluan akan dijelaskan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan relevansi terhadap pengerjaan tugas akhir. Berdasarkan uraian pada bab ini, harapannya gambaran umum permasalahan dan pemecahan masalah pada tugas akhir dapat dipahami.

1.1 Latar Belakang

Persaingan merupakan hal yang tidak dapat dihindarkan dalam dunia bisnis. Dalam persaingan tersebut tidak dapat dipungkiri lagi bahwa akan selalu ada kerugian dan keuntungan. Maka dari itu demi menghindari kerugian akibat persaingan bisnis maka sebuah perusahaan perlu mengenali pelanggannya lebih dekat melalui sebuah metode untuk mengelola hubungan pelanggan, atau yang biasa dikenal dengan istilah *Customer Relationship Management* (CRM). Manajemen hubungan pelanggan telah menjadi prioritas utama bagi banyak perusahaan. Dalam banyak pasar kompetitif, bisnis berinvestasi secara substansial untuk mengimplementasikan manajemen hubungan pelanggan (CRM) [1]. Salah satu fungsi utama dari pendekatan dengan CRM adalah mengenali lebih dekat siapa saja orang-orang yang menjadi pelanggan perusahaan. Pendekatan ini kemudian memberi jalan untuk menargetkan pemasaran, di mana subset atau segmen konsumen menerima pesan pemasaran yang disesuaikan dengan segmen pasar mereka. Pendekatan yang lebih selektif ini menghasilkan penghematan biaya dan tingkat respons yang meningkat [2]. Hanya saja dalam penelitian ini titik beratnya bukan pada *target marketing* namun pengelompokan pelanggan untuk PT. Alif Duta Persada.

PT. Alif Duta Persada merupakan sebuah perusahaan distributor yang berada di kawasan Kota Probolinggo. Perusahaan ini adalah distributor resmi dari PT. Unilever Indonesia untuk daerah Probolinggo dan sekitarnya. PT. Unilever Indonesia merupakan perusahaan yang terus tumbuh penjualannya, tercatat untuk pertengahan tahun 2017 lalu penjualannya terus meningkat hingga mencapai angka Rp 40 triliun [3]. Dengan menjadi distributor perusahaan sebesar itu maka PT. Alif Duta Persada sendiri harus memenuhi beberapa syarat yang telah ditentukan, salah satunya adalah pengelompokan pelanggan yang dimiliki berdasarkan omzet untuk pelanggan tersebut serta pemberian bonus pelanggan yang diberikan tiap beberapa waktu sekali. Permasalahan yang terjadi selama ini adalah pihak PT. Alif Duta Persada mengelompokkan pelanggannya secara manual setiap satu atau dua bulan sekali, dan pendistribusian bonus yang diberikan PT. Unilever Indonesia selalu diserahkan pada tiap-tiap *sales* yang mengelolanya sehingga sangat tidak terstandar dan rawan untuk diselewengkan.

Untuk menyelesaikan permasalahan tersebut solusi yang dapat dilakukan adalah dengan mengotomatisasi proses segmentasi pelanggan dan melakukan *clustering* berdasarkan beberapa faktor untuk menentukan siapa saja pelanggan yang pantas dan sesuai untuk menerima bonus sebagai bentuk CRM perusahaan. Hal itu berguna untuk memastikan bahwa salah satu bentuk CRM tersebut tepat guna dan diharapkan efektif untuk meningkatkan kepuasan pelanggan dan meningkatkan pendapatan yang diperoleh dari pelanggan tersebut.

Cluster analysis adalah Analisis cluster pengelompokan objek data berdasarkan informasi yang hanya ditemukan pada data yang menggambarkan objek dan hubungannya. Tujuannya

adalah bahwa objek dalam suatu kelompok serupa (atau terkait) satu sama lain dan berbeda dari (atau tidak terkait dengan) objek dalam kelompok lain [4]. Studi-studi tentang segmentasi pasar yang berbeda sudah pernah dilakukan sebelumnya, salah satu yang telah dilakukan adalah *clustering* pada kelompok pelancong yang homogen. Pengelompokan ini telah terbukti kegunaannya dalam menentukan kebijakan yang lebih spesifik dan efektif yang bertujuan untuk mempromosikan perubahan perilaku dan keberlanjutan sistem transportasi [5].

Pada dasarnya banyak metode yang bisa digunakan untuk melakukan segmentasi pelanggan salah satunya adalah penggalian data pada riwayat transaksi pembelian pelanggan pada periode tertentu. Kemudian dari data transaksi tersebut dapat dirumuskan beberapa variabel yang dapat ditentukan menjadi kriteria dalam melakukan segmentasi pelanggan. Variabel yang umum dijadikan tolak ukur adalah model RFM (*Recency, Frequency,* dan *Monetary),* model tersebut adalah metode *customer value analysis* yang telah banyak digunakan untuk mensegmentasi pelanggan [6]. Reinartz dan Kumar dalam bukunya [7] menyatakan bahwa model RFM tidak dapat membedakan mana pelanggan yang memiliki hubungan jangka panjang atau jangka pendek dengan perusahaan. Atas dasar tersebut maka dikembangkan model RFM dengan menambahkan variabel baru, salah satunya adalah penambahan *Length* (L) sebagai pertimbangan untuk mengungkapkan loyalitas pelanggan sehingga terbentuk model LRFM yang merupakan ekstensi dari model RFM.

Algoritma *clustering* yang akan digunakan dalam tugas akhir ini adalah *K-Means*. *K-Means* merupakan salah satu metode dalam *cluster analysis* yang tergolong sederhana dan telah banyak digunakan. Metode ini membagi semua objek ke dalam

K kelompok berdasarkan jaraknya, sehingga setiap objek akan masuk dalam kelompok yang paling dekat dengan posisinya. Jarak dari tiap objeknya dihitung menggunakan *Euclidean Distance*.

Dengan adanya penelitian ini diharapkan hasilnya dapat mendukung pihak PT. Alif Duta Persada dalam memudahkan proses segmentasi pelanggan secara otomatis sehingga memudahkan salah satu proses CRM perusahaan untuk menentukan pembagian bonus.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, berikut adalah rumusan masalah yang menjadi permasalahan utama dalam penelitian ini:

1. Bagaimana implementasi metode *K-means* pada segmentasi pelanggan berdasarkan model LRFM?
2. Bagaimana akurasi hasil segmentasi menggunakan metode *clustering K-means*?
3. Bagaimana visualisasi dapat mempermudah analisis segmentasi pelanggan?
4. Bagaimana analisis segmentasi dilakukan terhadap hasil *clustering*?

1.3 Batasan Masalah

Dalam penelitian ini, terdapat beberapa batasan masalah penelitian yaitu:

1. Data yang akan diolah adalah data transaksi penjualan dari PT. Alif Duta Persada pada rentang waktu 1 Desember 2017 hingga 30 Januari 2018
2. Variabel yang digunakan sebagai parameter adalah jarak waktu pertama kali pembelian (*length*), jarak waktu pembelian terakhir (*recency*), frekuensi pembelian, dan

total biaya pembelian pelanggan yang mengacu pada metode LRFM
3. Perangkat lunak yang digunakan sebagai alat bantu *clustering* adalah Jupyter Notebook
4. Visualisasi *cluster* yang digunakan berbasis python

1.4 Tujuan

Dalam penelitian ini, tujuan yang ingin dicapai adalah :

1. Menghasilkan segmentasi pelanggan dengan metode *clustering K-means* dan berdasarkan model LRFM
2. Mengetahui akurasi dari segmentasi yang telah dihasilkan oleh metode *K-means*
3. Membuat visualisasi *clustering* pelanggan untuk mempermudah analisa hasil segmentasi
4. Menghasilkan analisa segmentasi dari *clustering* yang telah dilakukan

1.5 Manfaat

Manfaat yang diharapkan dapat dihasilkan dari pengerjaan tugas akhir ini adalah :

Bagi Peneliti

Penelitian ini dapat memberikan pengetahuan mengenai proses *clustering* menggunakan metode *K-means* pada data transaksi pelanggan.

Bagi Instansi Terkait

Manfaat penelitian ini adalah untuk membantu PT. Alif Duta Persada mengotomatisasi proses *clustering* pelanggan berdasarkan algoritma dan metode yang telah dibuat. Selain itu juga membantu menentukan siapa saja pelanggan yang berhak

mendapatkan bonus yang diberikan secara objektif berdasarkan performa pelanggan, sebagai bentuk manajemen hubungan pelanggan yang baik dan tepat sasaran.

1.6 Relevansi

Relevansi penelitian ini terhadap ruang lingkup Sistem Informasi berada pada Laboratorium Rekayasa Data dan Intelejensi Bisnis dengan topik penggalian data dan *clustering*. Penelitian ini merupakan implementasi dari berbagai mata kuliah, yaitu: Sistem Cerdas, Penggalian Data dan Analitika Bisnis, serta Manajemen Rantai Pasok dan Hubungan Pelanggan.

BAB II
TINJAUAN PUSTAKA

Pada bab ini akan dijabarkan mengenai referensi yang terkait dengan penelitian tugas akhir ini beserta teori yang digunakan dalam tugas akhir ini.

2.1 Studi Sebelumnya

Penelitian-penelitian sebelumnya dapat menjadi referensi dan dasar penelitian tugas akhir ini. Penelitian terdahulu yang digunakan sebagai dasar penelitian tugas akhir ini tertera pada tabel 2.1 :

Tabel 2.1 Penelitian Sebelumnya

Judul Penelitian 1	*K-Means and cluster models for cancer signature* [8]
Penulis; Tahun	Zura Kakushadze, Willie Yu; 2017
Deskripsi Umum	Penelitian ini membahas pengelompokan penyakit kanker. Hingga saat ini metode yang dapat diterima untuk tanda-tanda kanker adalah *nonnegative matrix factorization* (NMF), yang mana matriks NMF tersebut digunakan sebagai acuan untuk proses *clustering* menggunakan K-means. Hasil yang diperoleh melalui teknik NMF yang ditingkatkan adalah Kanker Hati, Kanker Paru dan Karsinoma Sel Ginjal tampaknya tidak memiliki struktur pengelompokan. Hal tersebut bagus karena dapat belajar sesuatu yang penting tentang jenis kanker tersebut. Namun, buruknya adalah dari sudut pandang terapi. Karena jenis kanker ini tampaknya memiliki sedikit kesamaan dengan orang lain,

	kemungkinan mereka akan membutuhkan terapi khusus.
Keterkaitan dengan Tugas Akhir	Referensi dalam memahami *clustering* dan *K-means* serta contoh alur berfikir kode program untuk melakukan *clustering* terhadap pelanggan.
Judul Penelitian 2	*Integrating RFM model and Cluster for Student Loan Subsudy Valuation*[9]
Penulis; Tahun	Zhao Dan; 2008
Deskripsi Umum	Penelitian ini melakukan *clustering* terhadap mahasiswa di perguruan tinggi dan universitas. *Clustering* ini bertujuan untuk membantu pengalokasian dana dari pemerintah kepada mahasiswa yang membutuhkan. Dengan mempertimbangkan model RFM dan pembobotan menggunakan AHP sebagai identifikasi utama mahasiswa yang membutuhkan. Hanya saja pada penelitian ini nilai *recency* diubah menjadi rasio (rata-rata harga makanan non-gandum/ biaya pokok rata-rata). Hasil akhirnya terbentuk 6 *cluster* dengan nilai rerata RFM terendah pada *cluster* 6. Hasil penelitian ini dapat mengidentifikasi mahasiswa yang membutuhkan dan perlu pinjaman subsidi pendidikan.
Keterkaitan dengan Tugas Akhir	Referensi dalam memahami konsep RFM beserta aplikasi metode *analytic hierarchy process*. Juga sebagai contoh dalam pengimplementasian variabel RFM beserta *K-means* dalam *clustering*.
Judul Penelitian 3	IMPLEMENTASI *FUZZY C-MEANS* DAN MODEL RFM UNTUK SEGMENTASI PELANGGAN (STUDI KASUS : PT. XYZ)
Penulis; Tahun	Denny Bintang Saputra; 2017

Deskripsi Umum	Pengimplementasian metode *fuzzy c-means* terhadap data transaksi pelanggan untuk *clustering* dengan tujuan melakukan segmentasi pelanggan PT. XYZ. Permasalahan yang dialami PT. XYZ adalah tidak adanya strategi promosi spesifik yang bisa ditentukan untuk kelompok atau *cluster* tertentu karena tidak mengenali karakteristik pelanggannya. Segmentasi dilakukan dengan metode *clustering fuzzy c-means*, serta model RFM yang dapat menggambarkan perilaku pelanggan dalam melakukan transaksi. Lalu hasil *clustering* tersebut divisualisasikan untuk memudahkan analisis segmentasi. Hasilnya didapat 3 segmen pelanggan dengan karakteristik yang berbeda-beda, dan dengan hasil tersebut perusahaan dapat menerapkan strategi pemasaran yang tepat untuk setiap segmen sesuai dengan karakteristiknya. Dan hasil segmentasi tersebut divisualisasikan dalam grafik *scatter plot* dan *box-plot.* Serta ditampilkan juga peringkat tertinggi hingga terendah pelanggan berdasarkan *customer lifetime value*.
Keterkaitan dengan Tugas Akhir	Sebagai referensi untuk melakukan segmentasi pelanggan menggunakan *clustering* serta sebagai panduan dalam menentukan metodologi pengerjaan tugas akhir.

2.2 Dasar Teori

Bagian dasar teori ini memberikan gambaran umum tentang landasan penjabaran tugas akhir, dimana bagian ini menjelaskan teori-teori yang berkaitan dengan tugas akhir ini.

2.2.1 PT. Alif Duta Persada

PT. Alif Duta Persada merupakan perusahaan distributor resmi bagi PT. Unilever Indonesia untuk wilayah Kota Probolinggo dan sekitarnya. Perusahaan ini telah mulai beroperasi semenjak tahun 2012. PT. Alif Duta Persada sebagai distributor dari PT. Unilever Indonesia, tbk bertanggung jawab untuk memasarkan produk-produk dari PT. Unilever Indonesia, tbk (hanya khusus pada kategori *Personal Care* dan *Home Care Food*) untuk seluruh wilayah Kota dan Kabupaten Probolinggo.

PT. Alif Duta Persada memiliki divisi marketing yang bertugas untuk membantu proses pemasaran dan pendistribusian produk-produk *Personal Care* dan *Home Care Food* dari PT. Unilever Indonesia, tbk. Contoh beberapa brand produk yang didistribusikan PT. Alif Duta Persada untuk *Personal Care* adalah AXE, Close Up, Dove, LUX, Ponds, Rexona, Sunsilk, Treseme, Vaseline, Citra, Clear, Fair & Lovely, dan Lifebuoy, sedangkan untuk *homecare and food* adalah Rinso, Molto, Sunlight, Wipol, Super Pell, Vixal, Cif, Blueband, Sariwangi, dan Bango.

2.2.2 Customer Relationship Management

Istilah manajemen hubungan pelanggan mencakup semua konsep yang digunakan oleh perusahaan dalam kaitannya dengan klien mereka termasuk menangkap, menyimpan dan menganalisis informasi tentang pelanggan, sambil mempertimbangkan privasi dan keamanan data. Ini adalah strategi bisnis yang mempengaruhi proses, budaya dan teknologi suatu organisasi untuk mengoptimalkan pendapatan dan meningkatkan nilainya dengan memahami dan memenuhi kebutuhan konsumen individu [10].

Maka dapat disimpulkan pengertian CRM adalah pendekatan perusahaan untuk memahami dan mempengaruhi perilaku

pelanggan melalui komunikasi yang bermakna untuk meningkatkan akuisisi pelanggan, retensi pelanggan, loyalitas pelanggan, dan profitabilitas pelanggan. Tujuan CRM adalah untuk menjalin hubungan yang lebih dekat dan lebih dalam dengan pelanggan serta untuk memaksimalkan customer's lifetime value [11]. Menurut Francis Buttle terdapat 4 tipe CRM, yaitu [12] :

a) *Strategic* CRM
   CRM *strategic* adalah strategi bisnis yang berpusat pada pelanggan yang menyasar pada memenangkan dan menjaga pelanggan yang menguntungkan bagi perusahaan.
b) *Operational* CRM
   CRM *operational* berfokus pada otomasi proses-proses yang berhadapan langsung dengan pelanggan seperti penjualan, pemasaran dan pelayanan pelanggan.
c) *Analytical* CRM
   CRM *analytical* berfokus pada *intelligent mining* untuk data yang berkaitan dengan pelanggan demi tujuan strategis ataupun taktikal.
d) *Collaborative* CRM
   CRM *collaborative* mengaplikasikan teknologi pada lintas batas organisasi dengan maksud untuk mengoptimalkan perusahaan, rekan dan nilai pelanggan.

2.2.3 Segmentasi Pelanggan

Konsep segmentasi pelanggan awalnya dikembangkan oleh pakar pemasaran Amerika, Wendell R. Smith, pada pertengahan tahun 1950. Segmentasi pelanggan mengacu pada klasifikasi pelanggan berdasarkan nilai, permintaan, preferensi dan faktor lain dalam strategi organisasi, model bisnis dan

target pasar yang jelas. Model segmentasi pelanggan adalah model yang dibangun dengan mengklasifikasikan pelanggan sesuai dengan standar tertentu pada variabel segmentasi yang dipilih [13]. Sedangkan segmentasi pasar adalah proses membagi pasar menjadi subset yang lebih homogen atau kurang homogen sehingga memungkinkan untuk menciptakan proposisi nilai yang berbeda [12].

Singkatnya, segmentasi yang efektif memungkinkan perusahaan menentukan kelompok pelanggan mana yang harus mereka coba layani dan cara terbaik memposisikan produk dan layanan untuk masing-masing kelompoknya [14]. Terdapat dua macam model segmentasi pelanggan berbasis pola konsumsi, yaitu model RFM dan model *customer value matrix* [13]. Penelitian tugas akhir ini menggunakan pengembangan model RFM yaitu model LRFM.

2.2.4 Penggalian Data

Pengertian sederhana penggalian data atau yang populer dikenal dengan *data mining* mengacu pada ekstraksi atau penggalian pengetahuan dari data yang berjumlah banyak[15]. Sebagian orang menganggap penggalian data sebagai sinonim untuk *Knowledge Discovery from Data* atau KDD, sedangkan sebagian lainnya menganggap penggalian data adalah salah satu tahap esensial dalam KDD.
Proses KDD sendiri merupakan rangkaian iteratif dengan langkah-langkah seperti berikut [15] :

1. *Data Cleaning* : untuk menghilangkan *noise* dan inkonsistensi dari data.
2. *Data Integration* : dimana beberapa sumber data dikombinasikan.

3. *Data Selection* : dimana data yang relevan dengan analisis yang akan dilakukan diambil dari basis data.
4. *Data Transformation* : dimana data ditransformasi menjadi bentuk yang sesuai untuk melakukan *mining.*
5. *Data Mining* : proses esensial dimana metode-metode intelijen diterapkan dalam rangka mengekstrak pola dalam data.
6. *Pattern Evaluation* : untuk mengidentifikasi pola yang benar-benar menarik yang merepresentasikan pengetahuan berdasarkan beberapa ukuran yang menarik.
7. *Knowledge Presentation* : dimana visualisasi dan teknik representasi pengetahuan digunakan untuk menampilkan pengetahuan yang telah digali.

Terdapat dua kategori pada penyelesaian masalah menggunakan Penggalian Data, yaitu *supervised learning* dan *unsupervised learning*[16]. *Supervised learning* merupakan pembelajaran menggunakan label dan *unsupervised learning* merupakan pembelajaran tanpa label. Kasus *clustering* merupakan tipe pembelajaran tanpa label atau masuk ke dalam kategori *unsupervised learning*.

2.2.5 Clustering

Clustering atau *cluster analysis* telah lazim digunakan dalam setiap disiplin ilmu yang melibatkan analisis data multivariat. Secara operasional definisi *clustering* dapat dinyatakan sebagai berikut: Jika terdapat sejumlah n objek, temukan kelompok berjumlah K berdasarkan ukuran kemiripan, sehingga kemiripan antara objek pada kelompok yang sama bernilai tinggi sedangkan kemiripan antara objek dalam kelompok yang berbeda rendah [17]. Tujuan dari *cluster analysis* sendiri adalah

untuk menemukan pengelompokan alami dari serangkaian pola, titik atau objek. Teknik dalam *clustering* harus memenuhi dua kriteria, pertama adalah tiap kelompok adalah homogen; objek yang berada dalam kelompok yang sama pasti mirip satu sama lain, dan yang kedua tiap grup/kelompok harus berbeda dengan kelompok lain atau objek dalam kelompok satu harus berbeda dengan semua kelompok lain [18].

Algoritma Clustering secara luas dapat dibagi menjadi dua kelompok: *hierarchical* dan *partitional*. Algoritma pengelompokan hirarkis secara rekursif menemukan *cluster* yang bersarang baik dalam mode *agglomerative* (dimulai dengan setiap titik data dalam kelompok tersebut, lalu menggabungkan kelompok cluster yang paling mirip secara berturut-turut untuk membentuk hirarki *cluster*) atau dalam mode *divisive (top-down*) (dimulai dengan semua titik data dalam satu *cluster* dan secara rekursif membagi setiap cluster menjadi kelompok-kelompok yang lebih kecil). Dibandingkan dengan algoritma clustering hirarkis, algoritma clustering parsial menemukan semua cluster secara bersamaan sebagai partisi data dan tidak memaksakan struktur hirarkis. Algoritma hirarkis yang paling terkenal adalah single-link dan complete-link; Algoritma partikular yang paling populer dan paling sederhana adalah *K-means* [17].

2.2.6 K-Means

K-means merupakan salah satu algoritma untuk melakukan *clustering* yang telah dikenalkan lebih dari 50 tahun yang lalu dan masih banyak digunakan [17]. Algoritma clustering *K-means* ini bersifat iteratif, numerik, tidak deterministik dan masuk dalam kategori unsupervised learning. Algoritma ini awalnya diusulkan oleh MacQueen dan dikembangkan oleh Hartigan dan Wong [19]. Ide mendasar dari metode ini adalah

membagi semua objek ke dalam K kelompok dan menentukan nilai rata-rata (*mean*) dari tiap kelompoknya yang kemudian disebut *centroid*. Yang setelahnya maka tiap objek dihitung jaraknya terhadap *centroid,* yang umumnya dihitung menggunakan *euclidean distance*. Tujuan dari *K-means* adalah untuk meminimalkan *sum squred error* (SSE) dari kesemua kelompok yang ada. Langkah-langkah utama algoritma *K-means* adalah sebagai berikut [17] :

1. Tentukan partisi awal dengan sejumlah K *cluster*; ulangi langkah kedua dan ketiga hingga anggota *cluster* menjadi stabil.
2. Buat partisi baru dengan menempatkan tiap objek pada pusat *cluster (centroid)* terdekat.
3. Hitung pusat *cluster (centroid)* yang baru.

Yang perlu diingat pada langkah-langkah diatas adalah pada langkah pertama, yaitu ulangi langkah kedua dan ketiga hingga anggota *cluster* menjadi stabil, atau dalam artian bahwa anggota *cluster* tidak akan berubah lagi ketika langkah kedua dan ketiga dilakukan.

Untuk menghitung jarak pada tiap objek digunakan *Euclidean Distance* dengan formula seperti berikut :

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2} \qquad [15]$$

Keterangan :

X_1, X_2 = objek satu dan objek dua

x_{1i}, x_{2i} = nilai objek pada tiap dimensi atau sumbunya

2.2.7 Model LRFM

Model LRFM merupakan pengembangan dari model yang sudah ada yaitu RFM (*recency, frequency, monetary*) dengan penambahan variabel *length*. Pengembangan ini dilakukan karena beberapa hal, salah satunya adalah fakta bahwa model RFM tidak bisa membedakan pelanggan mana yang memiliki hubungan jangka panjang dan jangka pendek dengan perusahaan [7]. Padahal loyalitas pelanggan bergantung pada durasi hubungan antara pelanggan dan perusahaan, maka dari itu perlu ditambahkan sebuah varuabel untuk mengukur tingkat loyalitas yang telah dibangun oleh pelanggan. Itulah mengapa beberapa literatur lebih memilih menggunakan model LRFM untuk melakukan segmentasi terhadap pelanggan suatu perusahaan.

Tabel 2.4 menjelaskan penjabaran untuk masing-masing variabel dalam model LRFM [20] :

Tabel 2.2 Penjabaran Variabel LRFM

Variabel	**Definisi**
Length (L)	Jarak waktu (hari) antara transaksi pertama dengan transaksi terakhir dalam interval desember 2017 hingga januari 2018. Semakin besar jarak waktunya maka semakin besar nilai L.
Recency (R)	Jarak waktu dari terakhir kali pelanggan melakukan transaksi hingga waktu saat ini, semakin kecil jarak waktunya maka semakin besar nilai R.
Frequency (F)	Merupakan rata-rata jumlah transaksi yang dilakukan dalam satu periode tertentu, semakin besar jumlah transaksi yang dilakukan maka semakin besar nilai F.

Variabel	Definisi
Monetary (M)	Adalah total uang yang dikeluarkan pelanggan dalam periode tertentu, semakin besar nilainya maka semakin besar nilai M.

2.2.8 Min-max normalization

Normalisasi merupakan kata yang dicetuskan oleh E. F. Codd pada 1972, normalisasi merupakan tahap dalam praproses data. Menormalisasikan data masukan akan membantu mempercepat tahap pembelajaran dan juga mengatasi masalah perbedaan jangkauan nilai dari setiap atribut [21].

Min-Max Normalization melakukan transformasi linear pada data original menjadi bentuk data dengan interval yang diinginkan. Dengan kata lain metode ini merubah nilai data menjadi nilai dalam skala minimal dan maksimal yang diinginkan. Berikut formula untuk melakukan m*in-max normalization* :

$$v' = new_{min} + (new_{max} - new_{min}) \times (\frac{v - A_{min}}{A_{max} - A_{min}})$$ [21]

Keterangan :

v’ = nilai yang baru hasil normalisasi
v = nilai awal sebelum dinormalisasi
new_{min} = nilai minimum pada skala yang baru
new_{max} = nilai maksimum pada skala yang baru

Pada penelitian ini akan digunakan skala data dalam rasio angka nol hingga satu (0-1), sehingga formula diatas dapat disederhanakan. Berikut formula yang telah disederhanakan :

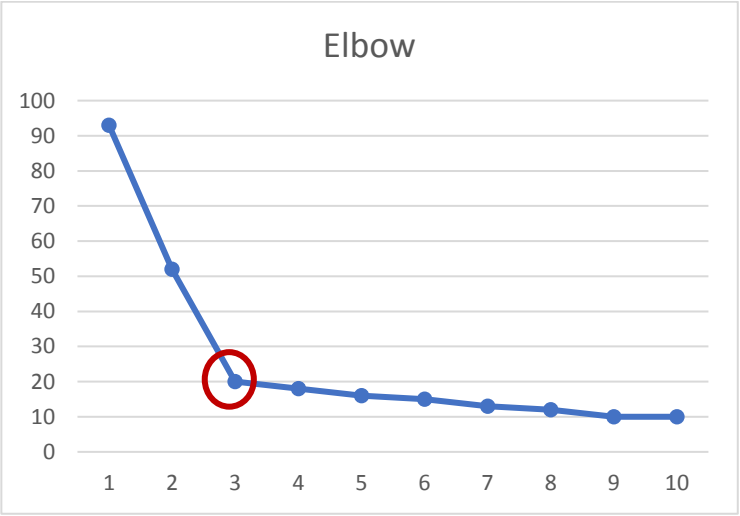
$$v' = \frac{v - A_{min}}{A_{max} - A_{min}}$$ [15]

Keterangan :

v’	= nilai yang baru hasil normalisasi
v	= nilai awal sebelum dinormalisasi
A_{min}	= nilai minimum atribut A
A_{max}	= nilai maksimum atribut A

2.2.9 Metode *Elbow*

Metode *Elbow* merupakan salah satu metode yang bersifat visual untuk menentukan jumlah *cluster* yang tepat, terutama pada algoritma *K-means*. Ide dasarnya adalah memulai dengan nilai k = 2, dan terus menambahkan 1 nilai di setiap langkahnya, lalu hitunglah jumlah *cluster* dan error yang menyertai di setiap nilai K tersebut. Pada beberapa nilai K error akan turun secara drastis, dan setelah itu error akan mencapai masa stabil pada suatu titik. Maka itulah nilai K yang baik [22].



Gambar 2.1 Grafik Metode Elbow

Gambar 2.1 grafik metode *elbow* menunjukkan nilai K (jumlah *cluster*) pada sumbu x dan nilai SSE sebagai tolak ukur kesalahan pada sumbu y. Dapat dilihat bahwa *error* menurun drastis ketika k berubah dari 1 ke 2 dan juga ke 3, namun setelah itu *error* hanya menurun perlahan setelah melewati nilai 3. Sehingga pada gambar diatas terlihat seperti bentuk ’siku’ pada nilai k=3, dan itulah nilai k yang akan digunakan.

2.2.10 Analytical Hierarchy Process (AHP)

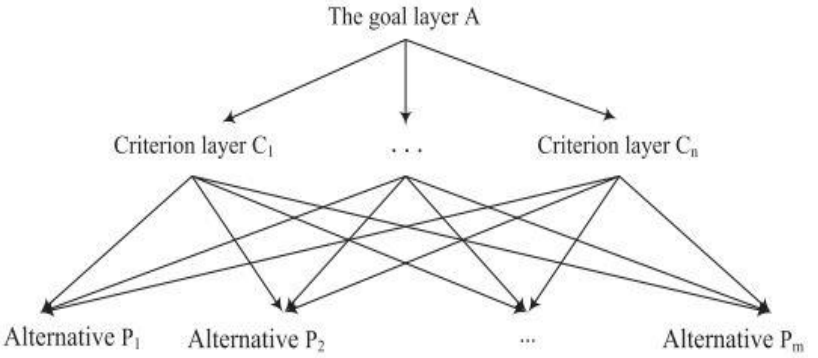
Banyak metode telah dikembangkan untuk membantu dan memfasilitasi aktivitas pembuatan keputusan yang kompleks. Salah satunya adalah metode *analytical hierarchy process* (AHP) yang dikembangkan oleh Thomas Saaty, dengan menurunkan vektor prioritas dari matriks *pairwise comparisons*. Metode AHP ini pertama kali muncul pada tahun 1972, namun deskripsi lengkap tentang model tersebut baru tersedia pada tahun 1980 [23]. Hingga saat ini metode AHP merupakan yang paling banyak dipelajari dan diutilisasi dari semua metode *Multi-criteria decision analysis* (MCDA). *Analytic Hierarchy Process* (AHP) adalah teori pengukuran umum. AHP digunakan untuk mendapatkan skala relatif dari angka absolut dalam perbandingan diskret dan kontinyu dalam struktur hirarkis bertingkat[24]. AHP adalah tipikal metode rekayasa sistem yang mengubah analisis kualitatif menjadi analisis kuantitatif [25].

Secara umum terdapat tiga langkah untuk menentukan bobot atau kepentingan pada pembuatan keputusan [25] :

1. *Hierarchical Model*

   Dengan membuat model hierarki maka masalah dalam pembuatan keputusan menjadi berbentuk hierarkis dan kompleksitas masalah dapat diuraikan. Permasalahan diuraikan mulai dari yang umum hingga mendetil dan dinyatakan dalam beberapa tingkatan. Model hierarkis

ini biasanya terdiri dari tiga lapisan. Lapisan puncak (1) mewakili tujuan umum untuk menentukan urutan kepentingan; lapisan tengah (2) berisi kriteria yang mempengaruhi tujuan dan digunakan untuk mengevaluasi alternatif pilihan; lapisan terbawah (3) yang berisi alternatif untuk mencapai tujuan.



Gambar 2.2 Model Hierarkis

2. *Pairwise comparison judgement matrix*
   Dalam metode AHP ini matriks penilaian dibuat dengan membandingkan dua elemen secara berpasangan. *Pairwise comparison* (perbandingan berpasangan) digunakan untuk menentukan kepentingan relatif pada tiap kriteria. Sang pembuat keputusan harus mengungkapkan opininya tentang nilai dari satu perbandingan berpasangan pada satu waktu, dengan skala 1-9 seperti tabel pada 2.3 yang menjelaskan preferensi penilaian [25] :

Tabel 2.3 Preferensi Penilaian Pairwise Comparison

Tingkat preferensi	Definisi
1	Sama pentingnya

Tingkat preferensi	Definisi
3	Sedikit lebih penting
5	Jelas lebih penting
7	Jauh lebih penting
9	Mutlak lebih penting
2, 4, 6, 8	Nilai tengah antara dua nilai yang berdekatan

Setelah terbentuk tabel *pairwise comparison* selanjutnya nilai dirata-rata lalu dinormalisasikan sehingga membentuk tabel baru yang bernama *normal comparison matrix*.

3. *Ranking and consistency check*

   Setelah membentuk matriks perbandingan, nilai kepentingan relatif dari tiap elemen dengan elemen lain dapat diperoleh. Namun ketika banyak perbandingan berpasangan dilakukan, beberapa inkonsistensi kemungkinan bakal muncul. Tujuan dari adanya matriks pengujian konsistensi adalah untuk memeriksa konsistensi dari setiap pertimbangan yang diberikan dan menghindari hasil yang saling bertentangan atau tidak konsisten. Penilaian dalam matriks dapat dikatakan konsisten jika *consistency ratio* (CR) bernilai kurang dari 10%. Namun sebelumnya perlu menghitung *consistency index* (CI) terlebih dahulu. Berikut formula untuk menghitung CI dan CR :

   $CI = \frac{\lambda_{max} - n}{n-1}$ [25]

   $CR = \frac{CI}{RI}$ [25]

   Keterangan :

   λ_{max} = nilai eigenvalue maksimal

   n = banyaknya variabel

   RI = *random consistency index*

CR = *consistency ratio*

CI = *consistency index*

Nilai RI disesuaikan dengan jumlah variabel, dan mengacu pada tabel 2.4

Tabel 2.4 Random Index

n	2	3	4	5	6	7	8	9	10
RI	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.51

Terdapat tiga keluaran yang bisa dihasilkan oleh proses AHP, yaitu [26]:

Peringkat secara keseluruhan, yang dapat membantu memahami bagaimana setiap pelanggan dibandingkan dengan yang lain.

Pengukuran akan konsistensi dari preferensi para pembuat keputusan secara keseluruhan.

Sebuah fasilitas untuk melakukan analisis sensitivitas, yang mampu memberikan informasi tentang hubungan sebab-akibat antara beberapa faktor.

2.2.11 Customer Lifetime Value

Nilai dari seorang pelanggan sebanding dengan nilai yang pelanggan berikan pada perusahaan selama masa hidupnya. Beberapa studi menunjukkan bahwa kontribusi masa lalu dari pelanggan tidak selalu merefleksikan nilai masa depannya pada perusahaan [7]. *Customer lifetime value* (CLV) memperhitungkan total kontribusi finansial (pendapatan minus pengeluaran) dari pelanggan dalam masa hidupnya pada perusahaan sehingga dapat merefleksikan keuntungan masa depan pelanggan tersebut[6]. CLV sendiri didefinisikan sebagai jumlah keuntungan yang didapat perusahaan dari pelanggan selama masa transaksi setelah dikurangi dengan total biaya untuk memikat, menjual, dan melayani pelanggan dengan

memperhitungkan nilai waktu dari uang. Atau dapat disimpulkan bahwa CLV secara umum adalah nilai sekarang dari semua profit di masa depan yang didapat dari pelanggan pada saat memiliki hubungan dengan perusahaan [27].
Formula untuk menghitung nilai CLV sendiri adalah sebagai berikut :

$$CLV = NL \times WL + NR \times WR + NF \times WF + NM \times WM$$

Sedangkan untuk menghitung CLV dalam cluster digunakan formula sebagai berikut :

$$CLV = W_L \times C_L^j + W_R \times C_R^j + W_F \times C_F^j + W_M \times C_M^j \quad [6]$$

Keterangan :

CLV	= nilai CLV yang dicari
NL	= nilai normalisasi variabel *Length*
WL	= nilai bobot variabel *Length*
NR	= nilai normalisasi variabel *Recency*
WR	= nilai bobot variabel *Recency*
NF	= nilai normalisasi variabel *Frequency*
WF	= nilai bobot variabel *Frequency*
NM	= nilai normalisasi variabel *Monetary*
WM	= nilai bobot variabel *Monetary*
$C_L^j, C_R^j, C_F^j, C_M^j$	= nilai rata-rata dari tiap variabel tersebut

2.2.12 Sum Squared Error (SSE)

Sum Squared Error (SSE) merupakan salah satu metode untuk mengukur kesalahan atau *error* dalam suatu penelitian. Singkatnya SSE merupakan jumlah *error* yang dikuadratkan. Algoritma *K-means* mengelompokkan sekumpulan data sehingga fungsi SSE dapat diminimalkan. Error yang didapatkan biasanya diukur dengan menggunakan *Euclidean Distance* antara titik data atau objek dengan titik mean pada

cluster [28]. Sehingga SSE yang dihitung merupakan jumlah dari selisih titik data tiap objek dengan nilai *mean* pada *cluster*-nya yang dikuadratkan.

Misalkan μ_k adalah *mean* dari kelompok k (Ck) maka *sum squared error* untuk semua *cluster* dapat dinyatakan sebagai berikut :

$SSE = \sum_{k=1}^{K} \sum_{Xi \in Ck} ||X_i - \mu_k||^2$ [17]

Keterangan :

μ_k = mean dari *cluster* k

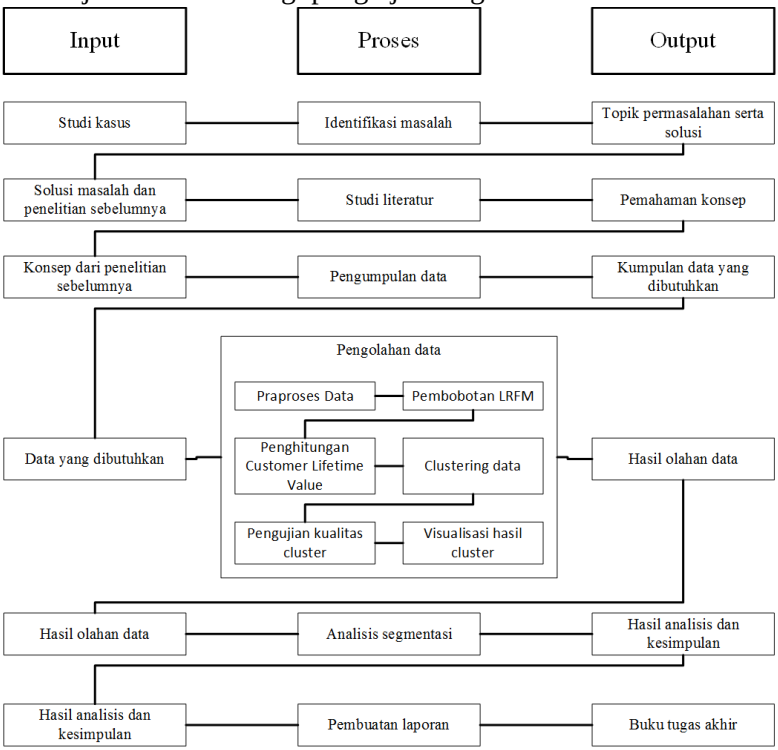
X_i = nilai objek ke-i pada *cluster* k

BAB III
METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan tentang metodologi yang akan digunakan dengan deskripsi setiap tahapannya.

3.1. Metodologi

Metodologi digunakan sebagai panduan dalam pengerjaan tugas akhir agar terstruktur dan sistematis. Gambar 3.1 menunjukkan metodologi pengerjaan tugas akhir :



Gambar 3.1 Metodologi Pengerjaan

Penelitian dimulai dengan mengidentifikasi masalah yang perlu ditemukan solusinya. Lalu dilakukan studi literatur untuk

memperoleh referensi terkait penelitian ini. Selanjutnya adalah pengumpulan data untuk mendapatkan data-data yang diperlukan atau dalam penelitian ini data tersebut berupa data transaksi pelanggan. Kemudian dilakukan praproses data agar data siap untuk diolah lebih lanjut. Setelah dilakukan praproses data ditransformasikan menjadi empat variabel yaitu *length*, *recency*, *frequency*, dan *monetary*, yang selanjutnya dilakukan pembobotan pada setiap variabel LRFM tersebut. Pembobotan tersebut dilakukan dengan metode *Analytical Hierarchy Process* sehingga menghasilkan bobot yang sesuai. Selanjutnya dilakukan penghitungan CLV yang merupakan indeks bagi setiap pelanggan untuk menunjukkan seberapa bernilainya pelanggan bagi perusahaan. Selanjutnya dilakukan *clustering* pelanggan dengan menggunakan metode *K-means*, yang kemudian hasil *clustering* tersebut divisualisasikan dan dianalisis karakter tiap segmennya.

3.1.1 Identifikasi Masalah

Pada tahap awal ini dilakukan analisis permasalahan yang akan diselesaikan pada tugas akhir ini. Studi kasus yang diambil adalah PT. Alif Duta Persada, dimana permasalahan yang diangkat adalah segmentasi pelanggan perusahaan. Setelah selesai mengidentifikasi masalah selanjutnya adalah penetapan rumusan masalah, batasan masalah, tujuan, dan manfaat.

3.1.2 Studi Literatur

Pada tahap kedua ini dilakukan pencarian dan pengumpulan referensi terkait penelitian terdahulu yang dapat menunjang pengerjaan tugas akhir. Literatur yang dibahas adalah terkait dengan topik penggalian data, *clustering*, *K-means*, metode LRFM, dan segmentasi pelanggan yang berhubungan dengan tugas akhir. Studi literatur dilakukan berdasarkan narasumber dengan teknik wawancara, buku, dan penelitian sebelumnya.

3.1.3 Pengumpulan Data

Pada tahap ketiga ini dilakukan pengumpulan data-data yang diperlukan untuk penelitian tugas akhir, terutama data yang berkaitan dengan proses *clustering*. Data-data yang dikumpulkan adalah sebagai berikut :

1. Data pelanggan yang pernah melakukan transaksi
2. Data transaksi penjualan produk pada pelanggan dari Desember 2017 hingga Januari 2018

Data transaksi yang diperoleh terdapat beberapa atribut penting yang krusial untuk proses *clustering* ini seperti tanggal transaksi, nama outlet, nama produk, total kuantitas, dan biaya total. Dari data transaksi tersebut sudah memadai untuk dilakukan *clustering* pelanggan berdasarkan data histori transaksi produk. Selain pengumpulan data, dilakukan juga wawancara pada narasumber terkait kondisi perusahaan yang berkaitan dengan segmentasi sebagai pendukung dalam penelitian tugas akhir.

3.1.4 Praproses Data

Tahap praproses ini dibutuhkan agar data mentah yang telah dikumpulkan dapat diolah sesuai dengan kriteria yang diinginkan. Masukan dari tahap ini adalah data yang telah berhasil dikumpulkan sebelumnya. Tahap ini meliputi tiga proses yaitu *data cleaning*, pemodelan LRFM, dan normalisasi.

3.1.4.1 Data Cleaning

Data cleaning merupakan subproses untuk melakukan pembersihan terhadap data mentah yang dimiliki. Data yang perlu dibersihkan adalah data yang kosong, negatif, dan tidak sesuai kriteria. Pembersihan data bertujuan untuk menghasilkan data yang bersih dan siap untuk dilakukan *clustering*.

3.1.4.2 Pemodelan LRFM

Pemodelan LRFM merupakan subproses untuk membentuk variabel *length*, *recency*, *frequency*, dan *monetary* dari data transaksi yang telah di *cleaning*. Pemodelan variabel-variabel tersebut antara lain :

a. *Length* : variabel ini diperoleh dari selisih waktu antara transaksi yang pertama kali dilakukan pelanggan dengan transaksi terakhirnya. Untuk menghitung nilai dari variabel ini digunakan atribut tanggal transaksi.
b. *Recency* : variabel ini diperoleh dari selisih waktu transaksi yang terakhir kali dilakukan oleh masing-masing pelanggan per tanggal 1 Februari 2018. Untuk menghitung nilai dari variabel ini digunakan atribut tanggal transaksi.
c. *Frequency* : variabel ini diperoleh dari rata-rata jumlah transaksi yang dilakukan oleh masing-masing pelanggan dengan mengagregasikan semua data berdasarkan pelanggan. Nilai diperoleh dari jumlah transaksi yang dilakukan dibagi dengan jumlah periode (2 bulan).
d. *Monetary* : variabel ini diperoleh dari jumlah total pengeluaran pelanggan dalam melakukan transaksi untuk masing-masing pelanggan. Untuk menghitung nilai dari variabel ini digunakan atribut total harga net.

3.1.4.3 Normalisasi

Normalisasi merupakan subproses untuk menormalkan nilai data agar seimbang antar satu variabel dengan variabel lainnya. Semua variabel perlu untuk dilakukan normalisasi, karena jika tidak maka variabel *monetary* akan memiliki rentang nilai yang

jauh melebihi variabel lainnya dan semua variabel tidak memiliki rentang nilai yang seimbang.

3.1.5 Pembobotan LRFM

Pada tahap ini dilakukan pembobotan terhadap keempat variabel dengan menggunakan metode AHP. Penilaian bobot dilakukan dengan memberikan kuesioner kepada manajer atau para pimpinan bagian marketing PT. Alif Duta Persada, sebanyak minimal 3 responden. Dari perhitungan AHP tersebut diperoleh nilai bobot untuk keempat variabel. Setelah itu dilakukan uji konsistensi terhadap nilai bobot yang diperoleh.

3.1.6 Penghitungan CLV

Tahap selanjutnya adalah menghitung indeks CLV untuk tiap pelanggan, yang mana CLV dihitung dari nilai keempat variabel LRFM pelanggan yang telah dikalikan dengan bobotnya. Indeks CLV kemudian didapat dengan menjumlahkan hasil perkalian tersebut. Luaran dari tahap ini adalah nilai CLV bagi masing-masing pelanggan.

3.1.7 Clustering Data

Tahapan selanjutnya adalah melakukan *clustering* terhadap data pelanggan. Sebelum clustering dilakukan ditentukan terlebih dulu nilai K sebagai jumlah *cluster* yang akan dibentuk. Untuk menentukan nilai K tersebut digunakan metode *Elbow*. Setelah mengetahui nilai K *clustering* dilakukan dengan menggunakan algoritma *K-means*. *Clustering* ini dilakukan untuk mengelompokkan pelanggan yang memiliki kemiripan tinggi sehingga dapat mengetahui ciri-ciri atau perilaku dari kelompok pelanggan yang diinginkan. Perilaku yang dimaksud disini adalah terkait keempat variabel yang jadi acuan, yaitu *length, recency, frequency,* dan *monetary*.

3.1.8 Pengujian Kualitas *Cluster*

Setelah selesai melakukan *clustering* selanjutnya adalah tahap pengujian kualitas *cluster* yang telah terbentuk. Masukan dari tahap ini adalah hasil *clustering* yang telah dilakukan sebelumnya. Pengujian ini dilakukan dengan menguji validitas berdasarkan SSE, yang diperoleh dari jumlah jarak semua objek terhadap titik pusat *cluster* di tiap *cluster*-nya. Nilai SSE yang semakin kecil menunjukkan kualitas *clustering* yang semakin baik.

3.1.9 Visualisasi *Cluster*

Pada tahap ini dilakukan visualisasi terhadap hasil *clustering* dengan media *dashboard*. Visualisasi dilakukan menggunakan bahasa pemrograman python, visualisasi ini bertujuan untuk memudahkan dalam pembacaan hasil *clustering* agar dapat menganalisis segmentasi pelanggan.

3.1.10 Analisis Segmentasi

Analisis hasil *clustering* bertujuan untuk menggambarkan bagaimana pelanggan-pelanggan dikelompokkan dalam beberapa *cluster*. Analisis segmentasi dilakukan untuk menjelaskan ciri atau kemiripan anggota dalam satu segmen, serta perbandingan dengan segmen lainnya. Analisis ini juga bertujuan untuk mengidentifikasi segmen mana yang lebih menjadi prioritas dibandingkan segmen lain. Pemrioritasan ini diperoleh dari tinggi rendahnya nilai dari keempat variabel pada keseluruhan anggota *cluster*. Lebih lanjut lagi ditambahkan juga analisis mengenai aspek lokasi/geografis yang dikaitkan dengan hasil *cluster*. Selain itu juga akan dilakukan pemetaan pelanggan berdasarkan wilayah asalnya. Dalam tahap ini juga dilakukan penjelasan mengenai hasil visualisasi dashboard yang telah dibuat.

3.1.11 Penyusunan Laporan Tugas Akhir

Tahap terakhir dalam penelitian ini adalah penyusunan laporan tugas akhir sebagai dokumentasi terhadap keseluruhan proses dan hasil dalam tugas akhir ini. Pendokumentasian ini dituangkan dalam sebuah buku Tugas Akhir (TA) dengan mengikuti format yang telah ditetapkan oleh laboratorium Rekayasa Data dan Intelegensi Bisnis (RDIB) serta yang berlaku di Departemen Sistem Informasi.

Halaman ini sengaja dikosongkan

BAB IV
PERANCANGAN

Bab ini menjelaskan tentang rancangan dari tugas akhir yang terdiri dari proses pengumpulan data, praproses data, pemberian bobot LRFM, penentuan jumlah cluster serta rancangan visualisasi.

4.1. Pengumpulan Data

Pengumpulan data dilakukan secara langsung pada perusahaan PT. Alif Duta Persada di Kota Probolinggo. Data yang digunakan sendiri adalah riwayat transaksi penjualan dengan rentang waktu dari tanggal 1 Desember 2017 hingga 30 Januari 2018 serta data daftar pelanggan yang melakukan transaksi tersebut. Selanjutnya dilakukan juga wawancara untuk menggali permasalahan yang ada terkait dengan proses CRM untuk pelanggan khususnya dalam hal segmentasi. Data diperoleh dalam bentuk dokumen excel. Data transaksi yang diperoleh berjumlah 78.262 baris data.

4.2 Praproses Data

Sebelum dapat diolah, data mentah harus diproses dalam tahap praproses data, sehingga data mentah siap untuk diolah dan dapat menghasilkan keluaran yang diharapkan. Tahapan dalam praproses data antara lain :

1. Pemilihan data untuk menentukan kolom yang akan digunakan sebagai variabel penentu *cluster*. Atribut yang dibutuhkan mengacu pada model LRFM, yaitu *length, recency, frequency,* dan *monetary*.
2. Pembersihan data untuk menghilangkan data yang kosong, nilai atribut yang tidak sesuai dengan ketentuan, atau beberapa data yang tidak berkaitan. Pembersihan ini dilakukan agar hasil yang didapat lebih baik dan tingkat kesalahan yang dihasilkan lebih kecil.

3. Pemodelan variabel LRFM, untuk mendapatkan nilai variabel LRFM yang diperoleh dari beberapa atribut yang tersedia.
    a. Variabel *length* diperoleh dari selisih waktu antara transaksi pertama dan transaksi terakhir dalam periode waktu tertentu, dalam penelitian ini maka dari 1 Desember 2017 hingga 30 Januari 2018. Nilai ini diperoleh dari atribut tanggal faktur (*invoice date*).
    b. Variabel *recency* diperoleh dari selisih waktu antara transaksi terakhir yang dilakukan pelanggan dengan tanggal batas yang ditentukan, dalam penelitian ini yaitu tanggal 1 Februari 2018. Nilai ini diperoleh dari atribut tanggal faktur (*invoice date*).
    c. Variabel *frequency* diperoleh dari jumlah transaksi yang dilakukan dibagi dengan lama periode waktu yang ditentukan. Nilai ini diperoleh dari jumlah atribut nomor faktur (*invoice number*).
    d. Variabel *monetary* diperoleh dari jumlah total uang yang dikeluarkan pelanggan selama periode tersebut. Nilai ini didapatkan dari akumulasi atribut harga pada catatan transaksi pelanggan.
4. Normalisasi data untuk menyetarakan skala nilai dari setiap variabel agar menjadi berimbang. Metode yang digunakan adalah *min-max normalization*.

4.3 Pembobotan LRFM

Tahap selanjutnya adalah menentukan bobot untuk setiap variabel LRFM. Bobot yang dihasilkan menentukan tingkat kepentingan dari tiap variabel dibandingkan dengan variabel lainnya. Bobot ini dihasilkan dari proses pengisian kuesioner oleh pihak terkait dari perusahaan PT. Alif Duta Persada Probolinggo. Responden yang mengisi ketiga kuesioner ini adalah *sales manager*, admin, dan direktur. Kuesioner yang diberikan berisikan data responden, petunjuk pengisian kuesioner, penjelasan variabel LRFM, skala preferensi, contoh

pengisian, dan daftar pertanyaan. Rancangan kuesioner secara lengkap dapat dilihat pada Lampiran A.

4.4 Penentuan jumlah *cluster*

Sebelum dilakukan proses *clustering*, tahap sebelumnya adalah penentuan nilai *k* atau jumlah *cluster* yang akan dibuat. Dalam penentuan nilai *k* ini, metode yang digunakan adalah metode *Elbow*. Hasil dari metode *Elbow* adalah grafik yang menunjukkan nilai error pada setiap nilai *k*. Nilai *k* dipilih ketika ada penurunan nilai error yang drastis pada suatu nilai *k*, diikuti dengan nilai yang stabil pada error selanjutnya. Dalam aplikasi yang dirancang, penentuan jumlah *k* ini dilakukan setelah pengguna mengunggah data transaksi yang akan diolah. Untuk menghitung nilai *k* ini menggunakan algoritma pemrograman python.

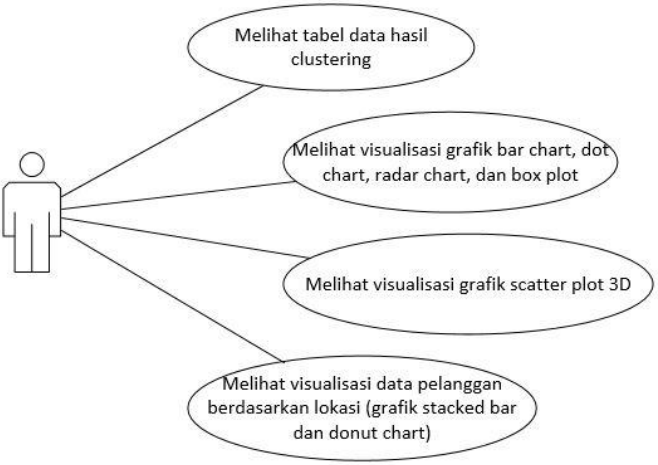
4.5 Rancangan visualisasi

Rancangan visualisasi hasil dari proses *clustering* meliputi penyusunan grafik-grafik yang digunakan, *use case diagram, use case description* dan desain antarmuka yang menjelaskan alur penggunaan aplikasi visualisasi.

4.5.1 Use Case Diagram

Diagram ini dibuat untuk menampilkan grafik yang menjelaskan karakteristik untuk setiap *cluster* yang telah dibentuk. *Use case diagram* tersaji pada gambar 4.1, dan terbagi dalam 4 *use case* utama, yaitu :

1. Melihat tabel data hasil *clustering*
2. Melihat visualisasi grafik *bar chart, radar chart*, dan *box plot*
3. Melihat visualisasi grafik *scatter plot 3D*
4. Melihat visualisasi data pelanggan berdasarkan lokasi (grafik *stacked bar* dan *donut chart*)



Gambar 4.1 Use Case Diagram

4.5.2 Use Case Description

Setiap *use case* yang telah dibuat dideskripsikan menjadi *use case description*. *Use case* melihat tabel data hasil *clustering* dideskripsikan menjadi *use case description* pada tabel 4.1. Untuk tujuh *use case description* yang lain dapat dilihat pada Lampiran B.

Tabel 4.1 Use Case Description melihat tabel data hasil clustering

UC-1 Melihat tabel data hasil *clustering*		
Purpose	Mengetahui hasil proses *clustering* pelanggan secara langsung	
Overview	Dimulai ketika pengguna mengakses aplikasi	
Actors	Pengguna aplikasi	
Pre Condition	Data hasil *clustering* telah dimasukkan pada direktori aplikasi	
Post Condition	Tabel hasil *clustering* ditampilkan	
Typical Course Event	***Actor***	***System***
	1. Membuka aplikasi	2. Mengambil data hasil *clustering*

UC-1 Melihat tabel data hasil *clustering*		
		pada direktori yang telah ditentukan 3. Menampilkan data hasil *clustering* dalam bentuk tabel
Alternate Flow Event	-	
Exceptional Flow Event	Aplikasi tertutup karena *error*	

4.5.3 Desain Antarmuka

a) Halaman utama aplikasi

Halaman utama aplikasi menampilkan hasil proses *clustering* yang telah dilakukan sebelumnya beserta dua tombol untuk melakukan *clustering* ulang dan proses visualisasi.

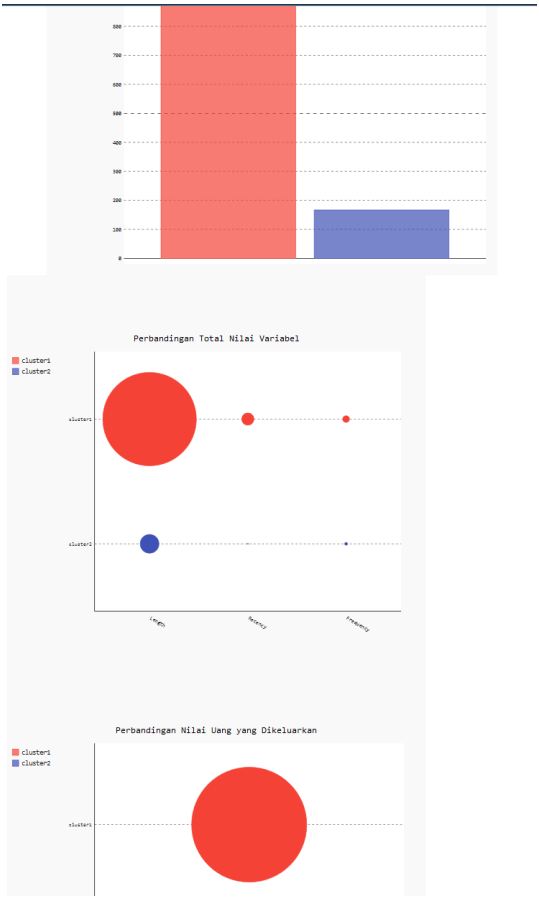
	nama_outlet	recency	monetary	frequency	length	clv	cluster	kecamatan
0	88	6	5.940295e+05	4.5	56	0.158445	0	Kotaanyar
1	371	6	2.258207e+06	5.5	56	0.174961	0	Kotaanyar
2	2 PUTRI	2	1.140490e+06	3.0	52	0.142525	0	Gending
3	27 WANDA, TK	8	9.224550e+05	3.5	49	0.138503	0	Gending
4	3F	7	9.766172e+05	3.5	49	0.139546	0	Kraksaan
5	9, TK	6	2.075380e+05	4.0	50	0.144500	0	Kraksaan
6	AA JAYA	5	7.635918e+05	4.0	49	0.146028	0	Gending
7	AA MAKMUR	1	1.152983e+06	4.0	56	0.159157	0	Maron
8	AAN , BU	3	4.377324e+06	9.0	56	0.222940	1	Pajarakan
9	ABADI	5	4.258201e+05	3.5	49	0.139426	0	Gending
10	ABADI JAYA II	3	3.606029e+06	7.5	56	0.203944	1	Kraksaan
11	ABC	2	6.997044e+06	9.0	57	0.233774	1	Gading
12	ABD HADI	1	6.567067e+06	15.5	58	0.305510	1	Maron
13	ABD KAMAL	6	1.879199e+05	3.0	49	0.132296	0	Paiton
14	ABD KARIM , HAJI	3	1.652730e+06	3.5	49	0.145259	0	Pajarakan
15	ABD RAHMAN	7	3.025632e+05	4.0	49	0.142759	0	Kraksaan
16	ABD ROHIM/MU ADI	1	9.254233e+06	4.0	55	0.185098	0	Kraksaan
17	ABD. JALIL	6	9.462947e+05	3.5	50	0.141504	0	Krejengan
18	ABDUL HALIM	6	2.532347e+05	4.0	56	0.151833	0	Paiton
19	ABDUL HANAN	1	3.409431e+06	5.0	56	0.177657	0	Banyu Anya

Gambar 4.2 Halaman Utama

Untuk mengakses hasil visualisasi maka perlu di klik tombol visualisasi.

b) Halaman Pertama Visualisasi

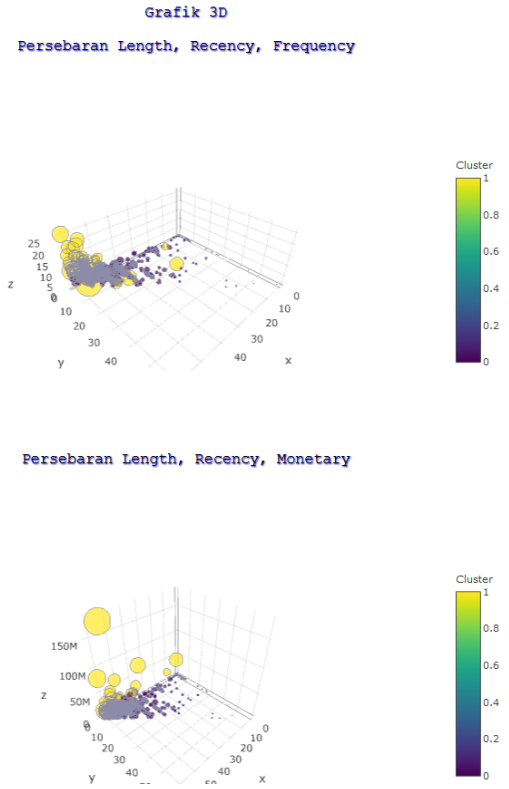
Halaman pertama visualisasi menampilkan beberapa grafik sekaligus. Mulai dari *bar chart*, *radar*, hingga *box plot*. Tampilan halaman pertama visualisasi dapat dilihat pada gambar 4.3.



Gambar 4.3 Halaman Pertama Visualisasi

c) Halaman Visualisasi 3D

Halaman selanjutnya merupakan visualisasi untuk menampilkan persebaran data variabel dalam bentuk tiga dimensi. Halaman ini sengaja dibuat untuk menganalisa performa pelanggan dengan melihat hasil nilai setiap pelanggan pada grafik *scatter plot 3D*. Tampilan halaman visualisasi tiga dimensi ini dapat dilihat pada gambar 4.4.



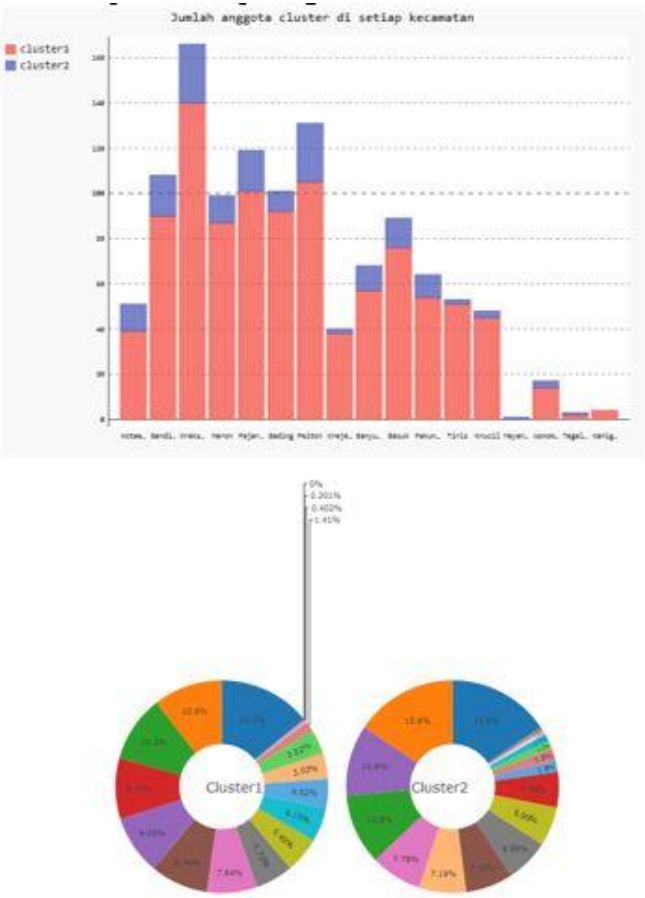
Gambar 4.4 Halaman Visualisasi 3D

Pada halaman tersebut terdapat empat grafik *scatter plot 3D* yaitu meliputi grafik perbandingan antara variabel *length*, *recency*, *frequency* lalu *length*, *recency*, *monetary* lalu *length*, *frequency*, *monetary* dan *recency*, *frequency*, *monetary*.

d) Halaman Visualisasi Lokasi

Halaman selanjutnya adalah visualisasi untuk menampilkan hubungan *cluster* pelanggan dengan lokasi tiap pelanggan. Grafik yang ditampilkan disini adalah *stacked* bar dan *donut chart*. Lokasi didapat dari alamat pelanggan yang kemudian

dicari kecamatan dari alamat tersebut. Untuk halaman visualisasi lokasi dapat dilihat pada gambar 4.5.



Gambar 4.5 halaman Visualisasi Lokasi

Halaman ini sengaja dikosongkan

BAB V
IMPLEMENTASI

Pada bab ini menjelaskan proses implementasi berdasarkan dari rancangan yang dibuat yaitu pra proses data, pembobotan LRFM, perhitungan CLV, proses *clustering*, uji performa *clustering* dan proses pembuatan visualisasi.

5.1 Pra Proses Data

Praproses data dilakukan dengan tujuan agar data mentah yang telah dikumpulkan dapat diolah sesuai dengan kriteria dan proses yang diinginkan. Masukan tahap ini adalah data transaksi yang sebelumnya berhasil diperoleh. Tahap praproses data ini meliputi *data cleaning*, pemodelan LRFM dan normalisasi.

5.1.1 Data Cleaning

Pembersihan data dilakukan untuk menghilangkan data yang tidak valid atau kurang sesuai nilainya. Sebelum itu, dipilih terlebih dahulu atribut yang penting berkaitan dengan model LRFM. Dari 20 atribut yang ada, 5 atribut terpilih akan digunakan untuk proses selanjutnya. Penjelasan mengenai atribut keseluruhan dan atribut yang terpilih dapat dilihat pada tabel 5.1.

Tabel 5.1 Penjelasan Atribut

Atribut	**Penjelasan**
DistID	Id pengenal untuk distributor pada pelanggan. (**Tidak digunakan**)
Salesman	Id untuk tiap sales yang bertugas. (**Tidak digunakan**)
SalesmaneNama	Nama salesman yang bertugas (**Tidak digunakan**)
INVNumber	Nomor invoice/faktur tanda pembelian. (**digunakan**)
INVDate	Tanggal dibuatnya *invoice*

Atribut	Penjelasan
	(**digunakan**)
Outlet	- (**Tidak digunakan**)
Outlet Name	Nama Outlet/ pelanggan (**digunakan**)
OutletAddress	Alamat outlet berada (**digunakan**)
SKUCode	Kode produk yang dijual (**Tidak digunakan**)
ProductName	Nama produk yang dijual (**Tidak digunakan**)
Case	Satuan jumlah kardus yang terjual (**Tidak digunakan**)
Dozen	Satuan (jumlah) lusin yang terjual (**Tidak digunakan**)
Pieces	Satuan barang yang terjual (**Tidak digunakan**)
Weight	Berat total barang yang dibeli (**Tidak digunakan**)
TotalQuantity(PCS)	Total kuantitas barang, akumulasi dari case, dozen, dan pieces (**Tidak digunakan**)
PriceCase	- (**Tidak digunakan**)
GSV	Harga kotor, belum ditambahkan pajak dan dikurangi diskon (**Tidak digunakan**)
Discount	Potongan harga yang diberikan (**Tidak digunakan**)
Tax	Besaran pajak yang harus dibayar (**Tidak digunakan**)
Net	Harga bersih yang harus dibayarkan. (**digunakan**)

Dari semua atribut tersebut, maka lima atribut yang digunakan adalah INVNumber, INVDate, Outlet Name, OutletAddress, dan Net.

Setelah terpilih atribut yang akan diolah selanjutnya dilakukan penghapusan penghapusan data yang tidak diperlukan dan tidak valid pada baris dan kolom yang ada. Data yang tidak valid berupa baris data yang menunjukkan total kuantitas bernilai negatif, harga bernilai negatif, hingga beberapa data dimana total kuantitas bernilai 0 tetapi terdapat harga. Selain itu semua juga terdapat data yang dianjurkan oleh admin PT. Alif Duta Persada untuk dihiraukan saja, sehingga setelah dilakukan pembersihan tersisa 75.608 baris data.

5.1.2 Pemodelan LRFM

Pemodelan variabel LRFM dilakukan agar nilai-nilai dari setiap variabel dapat ditentukan guna dijadikan masukan untuk proses *clustering*. Pemodelan LRFM dilakukan dengan menggunakan algoritma pemrograman python pada atribut-atribut terkait dari data excel yang telah diperoleh.

Algoritma yang digunakan untuk menghitung nilai dari keempat variabel dapat dilihat pada script 5.1.

```
1. LRFM=pd.DataFrame(columns=[['nama_outlet',
2. 'recency','length','monetary','frequency']])
3.
4. def LRFM_awal(df):
5.     n=0
6.     nama_outlet=pd.Series(df['Outlet Name'].unique())
7.      end_date = df['INVDate'].max() +
       dt.timedelta(days=1)
8.
9.      global LRFM
10.    for a in nama_outlet:
11.            ab=df[df['Outlet Name']==a]
12.          jml_inv=len(ab['INVNumber'].unique())
```

```
13.           ab['recent'] = pd.Series(end_date-
      ab['INVDate'])
14.           global LRFM
15.
16.           LRFM.loc[n,'recency']=ab['recent'].min()
17.           LRFM.loc[n,'nama_outlet']=a
18.           LRFM.loc[n,'length']=ab['recent'].max()-
      ab['recent'].min()
19.
       LRFM.loc[n,'monetary']=ab['harga_net'].sum()
20.            LRFM.loc[n,'frequency']=jml_inv/2
21.            n+=1
```

script 5.1 Algoritma Ekstraksi Data

Hal pertama yang dilakukan adalah membaca data file excel yang diunggah oleh pengguna, lalu menginisiasinya dalam sebuah variabel untuk diolah lebih lanjut.

Nilai variabel L adalah selisih antara transaksi pertama dan terakhir pelanggan dalam periode waktu tersebut. Atribut yang digunakan untuk menghitung ini adalah 'INVDate' yang menunjukkan waktu terjadinya transaksi. Nilai L diperoleh dengan mengurangkan tanggal transaksi maksimal (terbaru) dikurangi dengan tanggal transaksi minimal (terlama).

Nilai variabel R adalah selisih waktu dari hari terakhir pelanggan melakukan transaksi dengan hari yang dijadikan acuan, pada penelitian ini adalah tanggal 1 Februari 2018. Tanggal acuan ini diperoleh dari tanggal maksimal yang terdapat pada data ditambahkan dengan 1 hari, yang mana nilai itu tersimpan dalam variabel end_date. Maka untuk menghitung *recency* dengan cara mengurangkan end_date dengan tanggal transaksi lalu dipilih nilai minimumnya.

Nilai variabel F adalah jumlah transaksi yang dilakukan dibagi dengan periode waktunya, dalam kasus ini maka dibagi 2

karena periode data yang digunakan adalah 2 bulan. Nilai F diperoleh dengan menghitung jumlah transaksi yang diperoleh dari jumlah nomor faktur (INVNumber) yang berbeda dibagi dengan dua.

Nilai variabel M merupakan jumlah uang yang dikeluarkan oleh pelanggan pada keseluruhan transaksinya. Nilai M diperoleh dari penjumlahan atribut harga_net untuk setiap baris data dari pelanggan tersebut.

Setelah algoritma dieksekusi maka diperoleh 1.162 baris data yang berisi nama pelanggan serta nilai untuk variabel LRFM dari tiap pelanggan. Untuk melihat sebagian hasil dari pemodelan LRFM dapat dilihat pada tabel 5.1, sedangkan hasil pemodelan secara lengkap dapat dilihat pada Lampiran C.

Tabel 5.2 Sebagian Hasil Pemodelan LRFM

nama_outlet	recency	monetary	frequency	length
[88.0]	6	594029.5	4.5	56
[371.0]	6	2258207	5.5	56
['2 PUTRI']	2	1140490	3	52
['27 WANDA, TK']	8	922455	3.5	49
['3F']	7	976617.2	3.5	49
['9, TK']	6	207538	4	50
['AA JAYA']	5	763591.8	4	49

5.1.3 Normalisasi

Normalisasi data bertujuan untuk membuat nilai dari tiap variabel menjadai dalam rasio nilai yang setara. Normalisasi dilakukan menggunakan algoritma pemrograman python.
Proses ini dilakukan pada data variabel LRFM yang telah terbentuk, lalu diolah menggunakan metode *min-max normalization* dengan skala 0 sampai 1. Penjelasan detil untuk

nilai minimum dan maksimum setiap variabel dapat dilihat pada tabel 5.2.

Tabel 5.3 Rentang Nilai Atribut

Variabel	maksimum	minimum
L	61	0
R	59	1
F	26	0.5
M	178526080	4620

Operasi perhitungan *min-max* dilakukan berdasarkan rumus yang telah dijelaskan pada Bab II. Nilai normalisasi didapatkan dengan mengurangi nilai saat ini dengan nilai minimal variabel tersebur, kemudian dibagi dengan selisih antara nilai maksimum dan minimum variabel tersebut. Perhitungan detil dapat dilihat pada script 5.2.

```
1.      panjang=len(LRFM)
2.
3.      for b in np.arange(panjang):
4.          #untuk R = 1-R, karena semakin kecil
    semakin baik
5.          norm_lrfm.loc[b,'recency']=1-
    (LRFM.loc[b,'recency']-
    LRFM['recency'].min())/(LRFM['recency'].max()-
    LRFM['recency'].min())
6.
    norm_lrfm.loc[b,'nama_outlet']=LRFM.loc[b,'nama_
    outlet']
7.
    norm_lrfm.loc[b,'length']=(LRFM.loc[b,'length']-
    LRFM['length'].min())/(LRFM['length'].max()-
    LRFM['length'].min())
8.
    norm_lrfm.loc[b,'monetary']=(LRFM.loc[b,'monetar
    y']-
    LRFM['monetary'].min())/(LRFM['monetary'].max()-
    LRFM['monetary'].min())
```

```
9.
    norm_lrfm.loc[b,'frequency']=(LRFM.loc[b,'freque
    ncy']-
    LRFM['frequency'].min())/(LRFM['frequency'].max(
    )-LRFM['frequency'].min())
```

script 5.2 Normalisasi Data

Perlakuan khusus ditujukan untuk variabel R, dimana semakin kecil nilai variabel menunjukkan semakin baik nilai tersebut, yang mana hal itu berbeda dengan ketiga variabel lainnya dimana semakin besar nilai maka juga semakin baik nilai tersebut. Maka khusus untuk variabel R nilainya adalah 1 dikurangi dengan hasil nilai normalisasi. Untuk hasil lengkap tabel normalisasi dapat dilihat pada Lampiran D.

5.2 Pembobotan LRFM

Melalui proses pengisian kuesioner dari tiga responden, didapatkan bobot untuk preferensi atau kepentingan dari keempat variabel LRFM. Data hasil pengisian kuesioner tersebut kemudian dirata-rata untuk diolah pada tahap selanjutnya.

Setelah itu dilakukan normalisasi matriks dengan cara menjumlahkan total nilai dari setiap kolom lalu membagi setiap nilai pada sel tersebut dengan jumlah kolomnya. Selanjutnya bobot diperoleh dengan merata-rata nilai kriteria di tiap barisnya. Tabel baru yang dihasilkan disebut sebagai *normal comparison matrix*.

Selanjutnya adalah melakukan uji konsistensi untuk memastikan bahwa data yang dihasilkan oleh kuesioner telah bersifat konsisten atau tidak. Uji konsistensi dilakukan dengan mengalikan bobot tiap variabel dengan matriks rata-rata hasil kuesioner, kemudian menghasilkan 4 baris angka yang disebut *weighted sum vector*. Lalu menghitung *consistency vector* dengan cara membagi tiap baris dari *weighted sum vector*

dengan bobot untuk tiap variabel. Selanjutnya adalah mencari nilai eigen maksimum dengan cara merata-rata hasil dari *consistency vector*, lalu mencari nilai *consistency index* menggunakan rumus yang telah dijelaskan pada Bab II. Dan terakhir menghitung *consistency ratio* berdasarkan tabel *random index*. Hasil dari *consistency ratio* merupakan tolak ukur untuk menentukan data tersebut konsisten atau tidak.

5.3 Perhitungan CLV

Perhitungan CLV dilakukan untuk mengetahui nilai dari suatu pelanggan terhadap perusahaan. Data yang dibutuhkan adalah data LRFM dari tiap pelanggan dan hasil bobot untuk setiap variabel LRFM. CLV diperoleh dengan mengambil nilai LRFM masing-masing pelanggan yang kemudian dikalikan dengan bobot di tiap variabel tersebut. Kemudian, hasil perkalian tersebut dijumlahkan, hasil inilah yang disebut sebagai CLV, dan dapat membantu perusahaan untuk melakukan pemeringkatan pelanggan serta sebagai salah satu bahan analisis segmentasi. Kode untuk menghitung CLV ditampilkan pada script 5.3.

```
1.  nilai_clv=np.empty(panjang)
2.  def clv(df):
3.      for ab in np.arange(panjang):
4.          nilai_clv[ab]=
    float(df.loc[ab,'recency'])+
    float(df.loc[ab,'frequency'])+
    float(df.loc[ab,'monetary'])+
    float(df.loc[ab,'length'])
5.      return nilai_clv
```

script 5.3 Menghitung CLV

5.4 Proses Clustering

Proses *clustering* dilakukan untuk menghasilkan *cluster* yang beranggotakan pelanggan-pelanggan dengan karakteristik atau kebiasaan yang mirip. Masukan dari proses ini adalah data nilai

variabel LRFM seluruh pelanggan yang telah dikalikan dengan bobot dari setiap variabel. Dalam proses ini terdapat dua subproses, yaitu penentuan jumlah K dan *clustering* itu sendiri dengan menggunakan metode *K-Means*. Keseluruhan proses *clustering* dilakukan dengan algoritma pemrograman python menggunakan IDE Jupyter Notebook.

5.4.1 Penentuan Jumlah K

Data hasil dari proses sebelumnya, yaitu nilai variabel LRFM semua pelanggan yang telah dikalikan bobot untuk tiap variabel, digunakan sebagai masukan untuk menentukan nilai K atau jumlah *cluster* yang ingin dihasilkan. Metode yang digunakan dalam tahap ini adalah *elbow method*, dengan mengandalkan grafik dua dimensi mengenai nilai error yang dihasilkan. Pada sumbu X grafik tersebut menunjukkan nilai K, sedangkan sumbu Y merepresentasikan nilai SSE yang diperoleh pada setiap jumlah K.

Dalam penentuan nilai K ini, rentang K yang digunakan dimulai dari 2 hingga 10. Kemudian ditampilkan grafik yang akan merepresentasikan nilai SSE untuk setiap nilai K. Selanjutnya adalah menentukan nilai K terbaik dengan memperhatikan gambaran grafik serta penurunan nilai SSE yang dihasilkan. Kode untuk proses ini ditampilkan pada script 5.4.

```
1.      def elbow():
2.      norm_lrfm['recency']= norm_lrfm['recency']
    *bobotR
3.      norm_lrfm['frequency']=
    norm_lrfm['frequency'] *bobotF
4.      norm_lrfm['monetary']= norm_lrfm['monetary']
    *bobotM
5.      norm_lrfm['length']= norm_lrfm['length']
    *bobotL
6.
7.      global error_elbow
8.      for tiap in np.arange(9):
```

```
9.
    error_elbow[tiap]=hitung_error(norm_lrfm,tiap+2)
10.
11.     return analisa(error_elbow)
12.
13. def hitung_error(df, k):
14.     #kmeans clustering
15.     kmeans =
    KMeans(n_clusters=k).fit(norm_lrfm.drop(['nama_o
    utlet','clv'],axis=1, level=0))
16.
17.     #mengambil centroid tiap cluster
18.     global centroid
19.     centroid=kmeans.cluster_centers_
20.     global label
21.     label=kmeans.labels_
22.
23.     #∑_(k=1)^K▒∑_(Xi∈Ck)▒〖||X_i-μ_k ||〗^2
24.     for ab in
    np.arange(len(norm_lrfm['recency'])):
25.         error[ab][0] =
    np.square(np.absolute(df.loc[ab,'recency']-
    centroid[label[ab]][0]))
26.         error[ab][1] =
    np.square(np.absolute(df.loc[ab,'length']-
    centroid[label[ab]][1]))
27.         error[ab][2] =
    np.square(np.absolute(df.loc[ab,'monetary']-
    centroid[label[ab]][2]))
28.         error[ab][3] =
    np.square(np.absolute(df.loc[ab,'frequency']-
    centroid[label[ab]][3]))
29.
30.         ttl_error[ab] =
    error[ab][0]+error[ab][1]+error[ab][2]+error[ab]
    [3]
31.     return np.sum(ttl_error)
```

script 5.4 Elbow Method

5.4.2 Clustering dengan K-means

Setelah mendapatkan nilai K terbaik, maka selanjutnya dilakukan *clustering*. Masukan dari proses ini sama seperti pada proses sebelumnya yaitu data nilai variabel LRFM dari seluruh pelanggan yang telah dikalikan dengan bobot dari tiap variabel. Metode *clustering* yang digunakan adalah *K-Means.* Setelah dataset dimasukkan, proses *clustering* dijalankan dengan menggunakan nilai K yang telah ditentukan. Metode *K-Means* akan menentukan posisi keanggotaan dari tiap pelanggan, yaitu pada *cluster* mana pelanggan tersebut berada. Hasil *clustering* kemudian disimpan dalam bentuk file *excel* sebagai keluarannya. Kode yang digunakan untuk proses ini ditampilkan pada script 5.5.

```
1.  def cluster(jumlah):
2.      norm_lrfm['clv']=pd.Series(clv(norm_lrfm),
    index=norm_lrfm.index)
3.
4.      global hasil_akhir
5.
6.      kmeans =
    KMeans(n_clusters=jumlah).fit(norm_lrfm.drop(['n
    ama_outlet','clv'],axis=1, level=0))
7.      hasil_cluster=kmeans.labels_
```

script 5.5 K-means Clustering

5.5 Pengujian Kualitas *Clustering*

Setelah *clustering* dilakukan, dihasilkan *cluster* atau segmen yang membagi semua pelanggan menurut kebiasaannya. Untuk mengetahui akurasi dari hasil *clustering* yang dilakukan, diperlukan uji kualitas *clustering.* Pengujian ini menggunakan metode SSE (*Sum Squared Error*). Data yang diperlukan sebagai masukan adalah data nilai variabel LRFM semua pelanggan yang telah dikalikan dengan bobot setiap variabelnya. Algoritma SSE menghitung kuadrat selisih setiap

objek data pelanggan dengan titik pusat *cluster* di tiap dimensinya, kemudian menjumlahkan total selisih keseluruhan. Proses perhitungan SSE menggunakan script 5.6.

```
1.  def hitung_error(df, k):
2.      #kmeans clustering
3.      kmeans =
    KMeans(n_clusters=k).fit(norm_lrfm.drop(['nama_o
    utlet','clv'],axis=1, level=0))
4.
5.      #mengambil centroid tiap cluster
6.      global centroid
7.      centroid=kmeans.cluster_centers_
8.      global label
9.      label=kmeans.labels_
10.
11.     for ab in
    np.arange(len(norm_lrfm['recency'])):
12.
13.         error[ab][0] =
    np.square(np.absolute(df.loc[ab,'recency']-
    centroid[label[ab]][0]))
14.         error[ab][1] =
    np.square(np.absolute(df.loc[ab,'length']-
    centroid[label[ab]][1]))
15.         error[ab][2] =
    np.square(np.absolute(df.loc[ab,'monetary']-
    centroid[label[ab]][2]))
16.         error[ab][3] =
    np.square(np.absolute(df.loc[ab,'frequency']-
    centroid[label[ab]][3]))
17.
18.         ttl_error[ab] =
    error[ab][0]+error[ab][1]+error[ab][2]+error[ab]
    [3]
19.
20.     return np.sum(ttl_error)
```

script 5.6 SSE

5.6 Pembuatan Visualisasi

Proses visualisasi dilakukan menggunakan algoritma pemrograman python serta *framework* Flask. Untuk membuat visualisasi berbentuk *dashboard* ini diperlukan beberapa *library*, mulai dari Plotly, Pygal, hingga Pyplot. Penjelasan lebih rinci mengenai isi dari setiap halaman *dashboard* akan dijelaskan pada setiap aktivitas berikut.

5.6.1 Halaman Pertama

Halaman pertama menampilkan delapan grafik, yaitu dua *bar chart, radar chart,* dan empat *box-plot chart*.

5.6.1.1 Tampilan *Bar Chart*

Untuk menampilkan grafik *bar chart,* dibutuhkan data masukan hasil dari proses *clustering* dan pemodelan LRFM. Data tersebut kemudian diagregasikan berdasar kelompok *cluster*. Pada grafik pertama sumbu X merupakan tipe atau nomor *cluster,* sedangkan sumbu Y menunjukkan jumlah pelanggan dalam *cluster* tersebut. Sedangkan pada grafik kedua sumbu X menunjukkan variabel yang dibandingkan dan sumbu Y menunjukkan total nilai variabel tersebut. *Library* yang digunakan adalah *pygal*, sedangkan proses visualisasi *bar chart* dapat dilihat pada script 5.7 dan 5.8.

```
1.  import pygal
2.  from pygal.style import DefaultStyle
3.
4.  @app.route('/dashboard', methods=['GET',
    'POST'])
5.  def visual(k=None):
6.      d=np.empty([k, 5])
7.      e=np.empty([k, 4])
8.
9.      for a in range(k):
10.         b=(hasil_akhir['cluster']==a)
11.         d[a, 0]=len(hasil_akhir[b])
```

```
12.         d[a, 1] =
    hasil_akhir.loc[b,'length'].sum()
13.         d[a, 2] =
    hasil_akhir.loc[b,'recency'].sum()
14.         d[a, 3] =
    hasil_akhir.loc[b,'frequency'].sum()
15.         d[a, 4] =
    hasil_akhir.loc[b,'monetary'].sum()
16.     try:
17.         bar=pygal.Bar()
18.         bar.title = 'jumlah anggota cluster'
19.         for ab in range(k):
20.             bar.add('cluster'+str(ab+1),d[ab,
    0])
21.         bar_data=bar.render_data_uri()
```

script 5.7 Grafik Bar Chart Pertama

```
22. bar_chart = pygal.Bar(x_label_rotation=30,
    style=custom_style)
23.         bar_chart.title = 'Perbandingan Total
    Nilai Variabel'
24.         bar_chart.x_labels = ['Length',
    'Recency', 'Frequency','monetary(juta)']
25.         for ab in range(k):
26.             bar_chart.add('cluster'+str(ab+1),
    d[ab, 1:5])
27. dot_data=bar_chart.render_data_uri()
```

script 5.8 Grafik Bar Kedua

5.6.1.2 Tampilan Radar Chart

Untuk menampilkan grafik *radar chart*, dibutuhkan data masukan hasil dari proses *clustering* dan variabel LRFM. Data setiap variabel LRFM kemudian dihitung rata-ratanya berdasar kelompok *cluster* sehingga diketahui rata-rata nilai tiap variabel dari tiap *cluster*. Pada grafik ini sumbu X merupakan nilai variabel yang dibandingkan yaitu variabel *monetary* dan *recency*, sedangkan sumbu Y menunjukkan variabel *length* dan

frequency. *Library* yang digunakan adalah *pygal*, sedangkan proses visualisasi *radar chart* dapat dilihat pada script 5.9.

```
1.  import pygal
2.  from pygal.style import DefaultStyle
3.
4.  @app.route('/dashboard', methods=['GET',
    'POST'])
5.  def visual(k=None):
6.      d=np.empty([k, 5])
7.      e=np.empty([k, 4])
8.
9.      for a in range(k):
10.         b=hasil_akhir[hasil_akhir['cluster']==a]
11.         e[a, 0] =
    hasil_akhir.loc[b,'length'].mean()
12.         e[a, 1] =
    hasil_akhir.loc[b,'recency'].mean()
13.         e[a,
    2]=hasil_akhir.loc[b,'frequency'].mean()
14.         e[a,
    3]=hasil_akhir.loc[b,'monetary'].mean()/1000000
15.
16.     radar =
    pygal.Radar(dynamic_print_values=True,
    style=DefaultStyle( value_font_family=
    'googlefont:Raleway',
    value_font_size=30,
    value_colors=('white',)))
17.         radar.title = 'Rata-rata nilai variabel
    dari tiap cluster'
18.         radar.x_labels = ['Length', 'Recency',
    'Frequency', 'Monetary (juta)']
19.         for ab in range(k):
20.             radar.add('cluster'+str(ab+1), e[ab,
    0:4])
21.         radar_data=radar.render_data_uri()
```

script 5.9 Radar Chart

5.6.1.3 Tampilan Box-plot Chart

Grafik ini ditampilkan dalam bentuk *box plot* yang menggambarkan nilai matematis dari keempat variabel di setiap segmen. *Box plot* mendeskripsikan nilai minimal, nilai maksimal, kuartil I, kuartil II, dan kuartil III tiap variabel pada tiap *cluster*. *Box plot* pada tiap *cluster* memiliki warna berbeda antar satu dengan yang lainnya. Data yang dibutuhkan pada proses ini adalah data nilai LRFM tiap pelanggan dan hasil *cluster*. Proses visualisasi *box plot chart* dapat dilihat pada Script 5.10.

```
1.  import pygal
2.  from pygal.style import DefaultStyle
3.
4.  @app.route('/dashboard', methods=['GET',
    'POST'])
5.  def visual(k=None):
6.      d=np.empty([k, 5])
7.      e=np.empty([k, 4])
8.  #box plot (persebaran nilai tiap variabel)
9.      box_plot = pygal.Box()
10.     box_plot.title = 'persebaran nilai variabel
    Length'
11.     for ab in range(k):
12.
    b=hasil_akhir[hasil_akhir['cluster']==ab]
13.         box_plot.add('cluster'+str(ab+1),
    np.array(b['length']))
14.     box_data=box_plot.render_data_uri()
15.
16.     box2_plot = pygal.Box()
17.     box2_plot.title = 'persebaran nilai variabel
    Recency'
18.     for ab in range(k):
19.
    b=hasil_akhir[hasil_akhir['cluster']==ab]
```

```
20.         box2_plot.add('cluster'+str(ab+1),
    np.array(b['recency']))
21.     box2_data=box2_plot.render_data_uri()
22.
23.     box3_plot = pygal.Box()
24.     box3_plot.title = 'persebaran nilai variabel
    Frequency'
25.     for ab in range(k):
26.
    b=hasil_akhir[hasil_akhir['cluster']==ab]
27.             box3_plot.add('cluster'+str(ab+1),
    np.array(b['frequency']))
28.         box3_data=box3_plot.render_data_uri()
29.
30.         box4_plot = pygal.Box()
31.         box4_plot.title = 'persebaran nilai
    variabel Monetary'
32.         for ab in range(k):
33.
    b=hasil_akhir[hasil_akhir['cluster']==ab]
34.             box4_plot.add('cluster'+str(ab+1),
    np.array(b['monetary']))
35.         box4_data=box4_plot.render_data_uri()
```

script 5.10 Grafik Box-plot

5.6.2 Halaman Visualisasi 3 Dimensi

Halaman visualisasi 3 dimensi ini berisi empat buah grafik 3 dimensi yang menggambarkan persebaran nilai keempat variabel LRFM secara bergantian. Grafik yang ditampilkan pada halaman ini adalah *scatter 3d chart*.

5.6.2.1 Tampilan grafik *scatter 3 dimensi*

Masukan yang dibutuhkan pada tahap ini adalah data variabel LRFM, *customer lifetime value,* dan *cluster* pelanggan. Keempat grafik yang ditampilkan meliputi grafik perbandingan antara variabel *length, recency, frequency* lalu *length, recency, monetary* lalu *length, frequency, monetary* dan *recency,*

frequency, *monetary*. Proses visualisasi data dapat dilihat pada script 5.11.

```
1.  import plotly
2.  from plotly.offline import download_plotlyjs,
    init_notebook_mode, plot, iplot
3.  import plotly.graph_objs as go
4.
5.  init_notebook_mode(connected=True)
6.
7.  @app.route('/dashboard2', methods=['GET',
    'POST'])
8.  def visual2(k=None):
9.      dfs=hasil_akhir['clv']*50000
10.     trace1 = go.Scatter3d(
11.         x=hasil_akhir['length'][:],
12.         y=hasil_akhir['recency'][:],
13.         z=hasil_akhir['frequency'][:],
14.         text=hasil_akhir['nama_outlet'][:],
15.         mode='markers',
16.         marker=dict(
17.             sizemode='diameter',
18.             sizeref=1162,
19.             size=dfs[:],
20.             color = hasil_akhir['cluster'][:],
21.             colorscale = 'Viridis',
22.             colorbar = dict(title = 'Cluster'),
23.             line=dict(color='rgb(140, 140,
    170)'))
24.     )
25.
26.     trace2 = go.Scatter3d(
27.         x=hasil_akhir['length'][:],
28.         y=hasil_akhir['recency'][:],
29.         z=hasil_akhir['monetary'][:],
30.         text=hasil_akhir['nama_outlet'][:],
31.         mode='markers',
32.         marker=dict(
33.             sizemode='diameter',
```

```
34.             sizeref=1162,
35.             size=dfs[:],
36.             color = hasil_akhir['cluster'][:],
37.             colorscale = 'Viridis',
38.             colorbar = dict(title = 'Cluster'),
39.             line=dict(color='rgb(140, 140,
    170)')
40.         )
41.     )
42.
43.     trace3 = go.Scatter3d(
44.         x=hasil_akhir['length'][:],
45.         y=hasil_akhir['frequency'][:],
46.         z=hasil_akhir['monetary'][:],
47.         text=hasil_akhir['nama_outlet'][:],
48.         mode='markers',
49.         marker=dict(
50.             sizemode='diameter',
51.             sizeref=1162,
52.             size=dfs[:],
53.             color = hasil_akhir['cluster'][:],
54.             colorscale = 'Viridis',
55.             colorbar = dict(title = 'Cluster'),
56.             line=dict(color='rgb(140, 140,
    170)')
57.         )
58.     )
59.
60.     trace4 = go.Scatter3d(
61.         x=hasil_akhir['recency'][:],
62.         y=hasil_akhir['frequency'][:],
63.         z=hasil_akhir['monetary'][:],
64.         text=hasil_akhir['nama_outlet'][:],
65.         mode='markers',
66.         marker=dict(
67.             sizemode='diameter',
68.             sizeref=1162,
69.             size=dfs[:],
70.             color = hasil_akhir['cluster'][:],
71.             colorscale = 'Viridis',
72.             colorbar = dict(title = 'Cluster'),
```

```
            line=dict(color='rgb(140, 140,
    170)')
        )
    )

    graf=plot([trace1], show_link=False,
    output_type='div')
    graf2=plot([trace2], show_link=False,
    output_type='div')
    graf3=plot([trace3], show_link=False,
    output_type='div')
    graf4=plot([trace4], show_link=False,
    output_type='div')

    return render_template('graph2.html',
    grafik=Markup(graf), grafik2=Markup(graf2),
    grafik3=Markup(graf3), grafik4=Markup(graf4))
```

script 5.11 Scatter Plot 3D

5.6.3 Halaman Visualisasi Lokasi

Halaman visualisasi ini khusus dibuat untuk menambah kedalaman anlisa terhadap aspek lokasi. Halaman ini menampilkan persebaran anggota segmen di tiap kecamatan, serta menunjukkan kecamatan-kecamatan mana saja yang menyumbang anggota terbanyak dalam suatu segmen. Grafik yang ditampilkan adalah *stacked bar* dan *donut chart*.

5.6.3.1 Tampilan *Stacked Bar*

Untuk menampilkan grafik *stacked bar* diperlukan data anggota segmen di setiap kecamatan, maka dari itu data pelanggan perlu dilakukan identifikasi terlebih dulu untuk mengelompokkan pelanggan tersebut berada di kecamatan mana. Algoritma untuk mengidentifikasi kecamatan pelanggan dapat dilihat pada script 5.12, sedangkan untuk proses visualisasi grafik dapat dilihat pada script 5.13.

```
#mengisi kecamatan
tempat=pd.read_excel('..\coding template flask\\data_kecamatan\\lokasi.xlsx')

def kecm(df, nama_outl):
    n=0
    o=0
    kcmtn=[len(nama_outl)]
    for ab in nama_outl:
        bc=(df.loc[:,'Outlet Name']==ab)
        de=pd.Series(df.loc[bc,'OutletAddress'])
        ef=(tempat['alamat']==de[o])
        kcmtn.append(tempat.loc[ef,'kecamatan'])
        o+=len(de)
        n+=1
    return(kcmtn[1:])
```

script 5.12 Menentukan Kecamatan Tiap Pelanggan

Algoritma script 5.12 akan membaca alamat outlet dari setiap pelanggan kemudian mencocokkannya dengan data pada sebuah file excel yang berisi data seluruh alamat pelanggan serta kecamatannya. Kemudian data kecamatan tersebut ditambahkan pada kolom yang bernama kecamatan.

```
def visual3():
    k=len(hasil_akhir['cluster'].unique())

kcmtn=pd.Series(hasil_akhir['kecamatan'].unique())
    lokasi=np.empty([k, len(kcmtn)])


    for i in range(k):
        n=0

jml=hasil_akhir[hasil_akhir['cluster']==i]

        for daerah in kcmtn:
            ab=(jml.loc[:,'kecamatan']==daerah)
```

```
13.
    lokasi[i,n]=len(jml.loc[ab,'kecamatan'])
14.             n+=1
15.
16.     #stacked Bar
17.     stacked = pygal.StackedBar()
18.     stacked.title = 'Jumlah anggota cluster di
    setiap kecamatan'
19.     stacked.x_labels = kcmtn
20.     for i in range(k):
21.         stacked.add('cluster'+str(i+1),
    lokasi[i,:])
22.     stacked_data=stacked.render_data_uri()
```

script 5.13 Visualisasi Stacked Bar

Grafik ini menampilkan jumlah anggota pada sumbu Y dan lokasi kecamatan pada sumbu X. Dengan menggabungkan data jumlah kedua segmen dengan membedakan warnanya maka grafik ini dapat mempermudah analisis mengenai karakteristik segmen berkaitan dengan lokasinya.

5.6.3.2 Tampilan *Donut Chart*

Untuk menampilkan grafik ini data yang diperlukan masih sama yaitu jumlah anggota segmen di setiap lokasi kecamatan. Pada dasarnya grafik ini menampilkan perspektif yang berbeda dibanding *stacked bar*, dimana pada grafik ini membagi jumlah anggota di setiap daerah berdasarkan segmennya. Algoritma untuk menampilkan grafik ini dapat dilihat pada script 5.14.

```
1.      fig = {
2.        "data": [
3.          {
4.            "values": lokasi[0,:],
5.            "labels": kcmtn[:],
6.            "domain": {"x": [0, .48]},
7.            "name": "Cluster1",
```

```
8.            "hoverinfo":"label+percent+name",
9.            "hole": .4,
10.           "type": "pie"
11.         },
12.         {
13.           "values": lokasi[1,:],
14.           "labels": kcmtn[:],
15.           "text":"Cluster2",
16.           "textposition":"inside",
17.           "domain": {"x": [.52, 1]},
18.           "name": "Cluster2",
19.           "hoverinfo":"label+percent+name",
20.           "hole": .4,
21.           "type": "pie"
22.         }],
23.       "layout": {
24.             "title":"persebaran anggota
    cluster",
25.             "annotations": [
26.                 {
27.                     "font": {
28.                         "size": 20
29.                     },
30.                 "showarrow": False,
31.                 "text": "Cluster1",
32.                 "x": 0.20,
33.                 "y": 0.5
34.                 },
35.                 {
36.                 "font": {
37.                     "size": 20
38.                 },
39.                 "showarrow": False,
40.                 "text": "Cluster2",
41.                 "x": 0.8,
42.                 "y": 0.5
43.                 }
44.             ]
45.         } }
```

script 5.14 Visualisasi Donut Chart

BAB VI
HASIL PEMBAHASAN

Bab ini menjelaskan hasil dan pembahasan dari proses pembobotan LRFM, perhitungan CLV, *clustering K-Means*, dan uji kualitas clustering. Bab ini juga berisi tentang analisis hasil proses *clustering* dan juga visualisasi hasil *clustering*.

6.1 Pembobotan LRFM dengan AHP

Pembobotan LRFM dilakukan menggunakan metode *Analytical Hierarchy Process* atau AHP. Data yang digunakan sebagai masukan adalah hasil kuesioner yang telah diberikan pada tiga responden yang bersangkutan dari pihak PT. Alif Duta Persada. Hasil rekapan kuesioner dapat dilihat pada tabel 6.1 hingga 6.3.

Tabel 6.1 Hasil Kuesioner Responden 1

Kriteria	recency	frequency	monetary	length
recency	1	0.17	0.11	1.00
frequency	6.00	1	0.25	7.00
monetary	9.00	4.00	1	8.00
length	1.00	0.14	0.13	1

Tabel 6.2 Hasil Kuesioner Responden 2

Kriteria	recency	frequency	monetary	length
recency	1	0.14	0.11	0.33
frequency	7.00	1	0.20	5.00
monetary	9.00	5.00	1	7.00
length	3.00	0.20	0.14	1

Tabel 6.3 Hasil Kuesioner Responden 3

Kriteria	recency	frequency	monetary	length
recency	1	0.17	0.11	0.50

Kriteria	recency	frequency	monetary	length
frequency	6.00	1	0.33	7.00
monetary	9.00	3.00	1	7.00
length	2.00	0.14	0.14	1

Hasil kuesioner tersebut kemudian diolah dengan cara merata-rata semua data pengisian responden terhadap perbandingan empat kriteria LRFM. Hasil rata-rata ini kemudian dijumlahkan pada setiap kolomnya. Hasil rata-rata dapat dilihat pada Tabel 6.4.

Tabel 6.4 Hasil Rata-Rata Penilaian Responden

Kriteria	recency	frequency	monetary	length
recency	1.00	0.16	0.11	0.61
frequency	6.33	1.00	0.25	6.33
monetary	9.00	4.00	1.00	7.33
length	2.00	0.16	0.14	1.00
Jumlah	18.33333	5.320635	1.498016	15.27778

Setelah itu dilakukan normalisasi matriks terhadap jumlah kolom pada tabel hasil rata-rata nilai setiap kriteria atau variabel. Tabel baru hasil normalisasi tersebut disebut sebagai *normal comparisons matrix. Normal comparisons matrix* dapat dilihat pada tabel 6.5. dari tabel ini didapatkan nilai bobot untuk setiap kriteria.

Tabel 6.5 Normal Comparison Matrix

Kriteria	recency	frequency	monetary	length	bobot
recency	0.0545454	0.029833	0.074172	0.04	0.050
frequency	0.3454545	0.187947	0.166887	0.414545	0.279
monetary	0.4909090	0.75179	0.66755	0.48	0.598
length	0.1090909	0.03043	0.091391	0.065455	0.074

Selanjutnya perlu dilakukan uji konsistensi untuk mengetahui apakah data hasil kuesioner telah bersifat konsisten atau tidak. Uji konsistensi dilakukan dengan mengalikan bobot terhadap tabel hasil kuesioner, kemudian hasil tersebut adalah yang disebut *weighted sum vector*. Selanjutnya mencari *consistency vector* dengan cara membagi *weighted sum vector* dengan bobot tiap variabel. Untuk *consistency vector* dan *weighted sum vector* dapat dilihat pada tabel 6.6 dan 6.7.

Tabel 6.6 Weighted Sum Vector

Variabel	Weighted Sum Vector
Recency	0.205551028
Frequency	1.211716832
Monetary	2.702473149
Length	0.300300111

Tabel 6.7 Consistency Vector

Variabel	Consistency Vector
Recency	4.14103111
Frequency	4.347609929
Monetary	4.52249694
Length	4.053101051

Selanjutnya adalah mencari nilai *eigen* maksimum dengan cara merata-rata nilai *consistency vector*, yang kemudian didapatkan nilai 4.266059758. Nilai *eigen* maksimum tersebut kemudian menjadi masukan untuk menghitung CI, dan dengan mengacu pada Bab II maka nilai CI adalah:

$$CI = \frac{4.26606 - 4}{4 - 1} = 0.088687$$

Kemudian dicari *consistency ratio* dengan mengacu pada Bab II, yaitu membagi *CI* dengan *RI*. Sehingga nilai *CI/RI* yang diperoleh adalah :

$$\frac{0.088687}{0.90} = 0.09854$$

Karena nilai *CI/RI* adalah 0.098 atau kurang dari 0.1, maka hasil kuesioner dapat dikatakan bersifat konsisten. Maka dari itu, bobot kriteria telah valid dan dapat digunakan pada perhitungan CLV. Adapun bobot masing-masing kriteria adalah 0.074, 0.050, 0.279, dan 0.598 berturut-turut untuk variabel *length, recency, frequency,* dan *monetary.* Sehingga kriteria *monetary* memiliki bobot terbesar dibanding kriteria lain.

6.2 Perhitungan CLV

Setelah nilai bobot berhasil diperoleh, selanjutnya dilakukan perhitungan untuk CLV. Perhitungan CLV dilakukan dengan mengalikan masing-masing bobot dengan nilai LRFM masing-masing pelanggan, kemudian menjumlahkan keempat hasil perkalian tersebut. Sebagian hasil perhitungan CLV dapat dilihat pada Tabel 6.8, untuk hasil selengkapnya dapat dilihat pada Lampiran E.

Tabel 6.8 Sebagian Hasil Perhitungan CLV Pelanggan

nama_outlet	recency	monetary	frequency	length	clv
88	6	594029.5	4.5	56	0.1584
371	6	2258207	5.5	56	0.175
2 PUTRI	2	1140490	3	52	0.1425
27 WANDA, TK	8	922455	3.5	49	0.1385
3F	7	976617.2	3.5	49	0.1395
9, TK	6	207538	4	50	0.1445
AA JAYA	5	763591.8	4	49	0.146

6.3 Proses *Clustering*

Dalam proses ini akan dijelaskna mengenai hasil dari penentuan nilai K dengan metode *Elbow* dan *clustering* dengan metode *K-Means.*

6.3.1 Penentuan Nilai K

Penentuan nilai K dilakukan dengan menerapkan metode *Elbow* menggunakan algoritma pemrograman python. Hasil perhitungan metode Elbow, ditampilkan dalam plot dua dimensi pada Gambar 6.1. Nilai pada sumbu X adalah jumlah K yaitu 2 hingga 10, sedangkan sumbu Y adalah nilai SSE yang dihasilkan oleh setiap nilai K pada sumbu X. Metode *Elbow* menjelaskan bahwa nilai K yang diambil adalah pada titik dimana terjadi penurunan SSE yang signifikan dan diikuti oleh nilai yang cenderung konstan. Pada plot tersebut dapat dilihat bahwa seiring naiknya nilai K, maka nilai SSE akan semakin menurun. Namun pada plot yang dihasilkan tidak terlihat penurunan SSE yang signifikan, sehingga penentuan nilai K dilakukan dengan menghitung tingkat penurunan nilai SSE. Dan kemudian didapatkan bahwa penurunan nilai SSE yang paling signifikan adalah pada nilai K=2. Untuk hasil lengkap nilai SSE yang didapat untuk tiap nilai K dapat dilihat pada tabel 6.9, sedangkan nilai perbandingan *error* dapat dilihat pada tabel 6.10.

Tabel 6.9 Nilai Error Setiap K

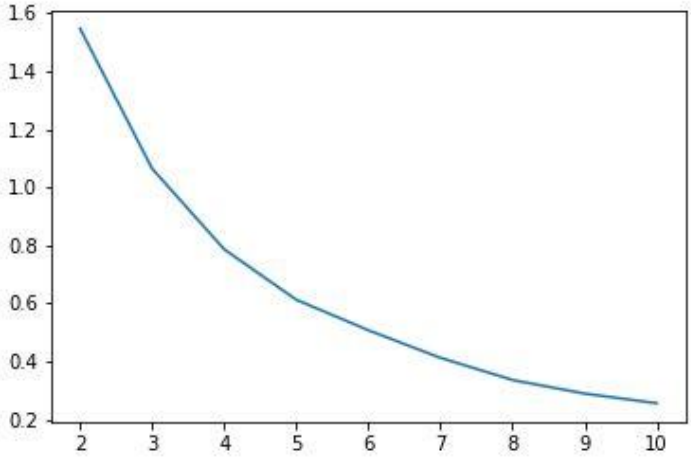
K	Error
2	1.546808
3	1.064151
4	0.786225
5	0.612481

K	Error
6	0.507408
7	0.412281
8	0.335098
9	0.289646
10	0.255385

Perbandingan penurunan *error* dilakukan berdasarkan data pada tabel 6.9, dan dipilih nilai penurunan *error* yang paling signifikan. Data perbandingan *error* disajikan pada tabel 6.10.

Tabel 6.10 Perbandingan Error

K	Perbedaan Error	Perbandingan Error
2	0.48266	**1.7366376**
3	0.27793	1.5996297
4	0.17374	1.6535600
5	0.10507	1.1045466
6	0.09513	1.2324918
7	0.07718	1.6981371
8	0.04545	1.3266237
9	0.03426	



Gambar 6.1 hasil metode Elbow

6.3.2 Clustering K-means

Proses *Clustering K-Means* dilakukan menggunakan algoritma pemrograman python. Luaran yang dihasilkan dari eksekusi *script* adalah pusat setiap *cluster*, serta pengelompokan pelanggan terhadap *cluster* yang telah ditentukan jumlahnya. Nantinya hasil dari pengolahan ini akan disimpan dalam bentuk *.xlsx* agar pengguna aplikasi dapat memiliki data tersebut.

6.3.3 Pengujian Kualitas *Clustering*

Pengujian kualitas dijalankan untuk mengetahui seberapa baik kualitas hasil dari proses *clustering* yang dilakukan. Pengujian dilakukan dengan mengukur *error* dalam bentuk SSE.

Pengujian SSE dilakukan menggunakan algoritma pemrograman python. Hasil pengujian SSE menampilkan nilai *error* total pada hasil *clustering* berdasarkan parameter yang telah digunakan. Nilai SSE yang diperoleh untuk jumlah K=2 adalah 1.5468083.

6.3.4 Hasil *Cluster*

Setelah berhasil menjalankan proses clustering maka keluaran yang dihasilkan adalah pemetaan *cluster* pelanggan. Contoh sebagian hasil *clustering* dapat dilihat pada tabel 6.11.

Tabel 6.11 Contoh Hasil Clustering

nama_outlet	recency	monetary	frequency	length	cluster
88	6	594029.5	4.5	56	0
371	6	2258207	5.5	56	0
2 PUTRI	2	1140490	3	52	0
27 WANDA, TK	8	922455	3.5	49	0
3F	7	976617.2	3.5	49	0
9, TK	6	207538	4	50	0
AA JAYA	5	763591.8	4	49	0
AA MAKMUR	1	1152983	4	56	0
AAN, BU	3	4377324	9	56	1
ABADI	5	425820.1	3.5	49	0
ABADI JAYA II	3	3606029	7.5	56	1
ABC	2	6997044	9	57	1
ABD HADI	1	6567067	15.5	58	1

Dari hasil *clustering* ini didapatkan dua *cluster*, dengan *cluster* pertama berisi 995 anggota dan *cluster* kedua berisi 167 anggota pelanggan.

6.4 Analisis Hasil Clustering

Proses *clustering* menghasilkan dua segmen dengan sejumlah 1.162 pelanggan yang menjadi anggotanya. Masing-masing segmen memiliki karakteristik yang berbeda. Karakteristik setiap segmen dipengaruhi oleh nilai variabel *length, recency, frequency,* dan *monetary*. Keempat variabel tersebut juga mempengaruhi nilai CLV dari tiap pelanggan. Nilai CLV inilah

yang menjadi ukuran tingkat kepentingan tiap pelanggan pada tiap segmen.

6.4.1 Analisis Karakteristik Segmen 1

Segmen pertama memiliki anggota sebanyak 995 pelanggan, dengan rata-rata nilai CLV pada *cluster* tersebut adalah 0.137935535 atau lebih rendah dari segmen yang kedua. *Cluster* ini cenderung merupakan kelompok pelanggan yang lebih kecil atau memiliki performa yang inferior dibandingkan dengan *cluster* kedua. Hal ini dibuktikan dengan lebih rendahnya rata-rata nilai dari keempat variabel pada kelompok pelanggan ini. Bahkan meskipun berjumlah jauh lebih banyak namun jumlah uang yang dikeluarkan secara keseluruhan tetap lebih sedikit dari pada *cluster* kedua, yaitu Rp 1.775.197.831 dibandingkan dengan *cluster* kedua yang bernilai Rp 2.215.374.684. Sedangkan untuk variabel *length* memiliki rata-rata nilai 46.09 hari dengan nilai terendah dari 0 hingga nilai tertinggi 61 hari. Untuk *recency* rata-rata adalah 6 hari, dengan nilai terendah 59 hari dan nilai tertinggi 1 hari. Lalu *frequency* bernilai rata-rata 3.3 kali transaksi tiap bulan, dengan nilai minimum 0.5 dan nilai maksimum 7. Dan terakhir variabel *monetary* memiliki nilai rata-rata 1.78 juta dan nilai minimum Rp 4.620 serta nilai maksimum Rp 24.764.092. Karakteristik segmen 1 lebih lengkap dapat dilihat pada Tabel 6.12.

Tabel 6.12 Karakteristik Segmen 1

Segmen		1
Peringkat Segmen		2
Jumlah Anggota		995
Karakteristik	**Length**	0 - 61 hari (rata-rata : 46.09 hari)
	Recency	1 - 59 hari (rata-rata : 6 hari)

	Frequency	0.5 - 7 kali (rata-rata : 3.3 kali)
	Monetary	Rp. 4.620, - Rp. 24.764.092,- (rata-rata : Rp. 1.784.118,-)

6.4.2 Analisis Karakteristik Segmen 2

Segmen kedua memiliki anggota sebanyak 167 pelanggan, dengan rata-rata nilai CLV pada *cluster* tersebut adalah 0.250248127. Dengan kata lain *cluster* ini cenderung memiliki anggota pelanggan yang lebih superior dari pada *cluster* pertama. Meski berjumlah sedikit namun nilai variabel *monetary* bernilai total sangat masif yaitu Rp 2.215.374.684. Sedangkan untuk variabel *length* memiliki rata-rata nilai 55.3 hari dengan nilai terendah dari 15 hingga nilai tertinggi 61 hari. Untuk *recency* rata-rata adalah 2.75 hari, dengan nilai terendah 20 hari dan nilai tertinggi 1 hari. Lalu *frequency* bernilai rata-rata 8.8 kali transaksi tiap bulan, dengan nilai minimum 1.5 dan nilai maksimum 26. Dan terakhir variabel *monetary* memiliki nilai rata-rata 13.26 juta dan nilai minimum Rp 971.389 serta nilai maksimum Rp 178.526.080.

Tabel 6.13 Karakteristik Segmen 2

Segmen		2
Peringkat Segmen		1
Jumlah Anggota		167
Karakteristik	**Length**	15 - 61 hari (rata-rata : 55.3 hari)
	Recency	1 - 20 hari (rata-rata : 2.75 hari)
	Frequency	1.5 - 26 kali (rata-rata : 8.8 kali)
	Monetary	Rp. 971.389, - Rp. 178.526.080,- (rata-rata : Rp. 13.265.716,-)

6.4.3 Perbandingan Karakteristik Segmen

Perbandingan karakteristik dari tiap segmen dapat dilihat pada tabel 6.14. dari tabel tersebut sebenarnya dapat dengan mudah disimpulkan bahwa segmen kedua selalu memiliki nilai variabel yang lebih baik dalam rata-ratanya. Sedangkan hal lain yang dapat diamati adalah ketimpangan atau perbedaan jumlah anggota antar segmen yang begitu signifikan. Hal ini bisa terjadi karena perbedaan kemiripan yang ada sangat tinggi sehingga skala perbedaan yang ada pun melonjak dan pelanggan terkelompokkan berdasarkan skala bisnisnya.

Tabel 6.14 Perbandingan Karakteristik Segmen

	Segmen 1	**Segmen 2**
Peringkat	2	1
Jumlah Anggota	995	167
Length	0 - 61 hari (rata-rata : 46.09 hari)	15 - 61 hari (rata-rata : 55.3 hari)
Recency	1 - 59 hari (rata-rata : 6 hari)	1 - 20 hari (rata-rata : 2.75 hari)
Frequency	0.5 - 7 kali (rata-rata : 3.3 kali)	1.5 - 26 kali (rata-rata : 8.8 kali)
Monetary	Rp. 4.620, - Rp. 24.764.092,- (rata-rata : Rp. 1.784.118,-)	Rp. 971.389, - Rp. 178.526.080,- (rata-rata : Rp. 13.265.716,-)

6.5 Analisis Visualisasi Clustering

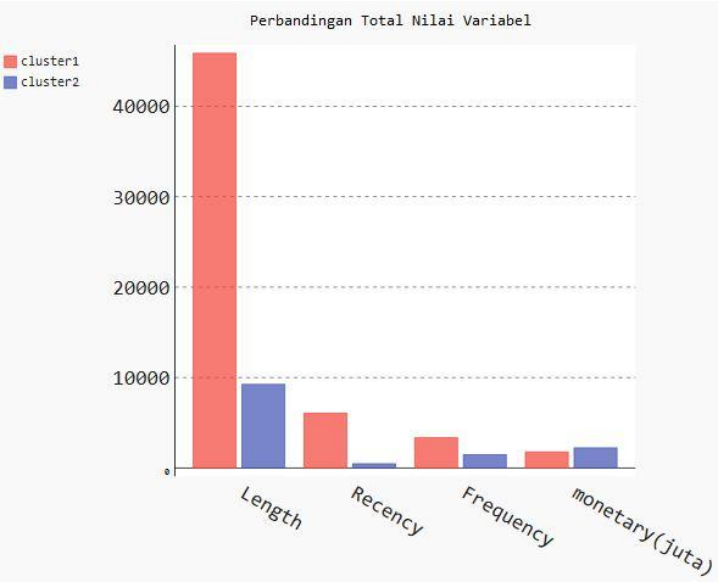
Hasil segmentasi pelanggan kemudian ditampilkan dalam bentuk grafik untuk mempermudah pengguna dalam menganalisa perilaku pelanggan pada tiap-tiap segmennya.

6.5.1 Grafik *Bar Chart*

Grafik *bar chart* ini ditampilkan untuk mengetahui perbandingan jumlah anggota di tiap segmen dan perbandingan nilai dari keempat variabel. Hasil grafik perbandingan anggota segmen dapat dilihat pada gambar 6.2.

Gambar 6.2 Bar Chart Anggota Segmen

Dari gambar 6.2 diketahui bahwa segmen 1 memiliki 995 anggota pelanggan dan segmen 2 memiliki 167 anggota pelanggan. Sedangkan *bar chart* untuk membandingkan nilai keempat variabel pada masing-masing segmen disajikan pada gambar 6.3.

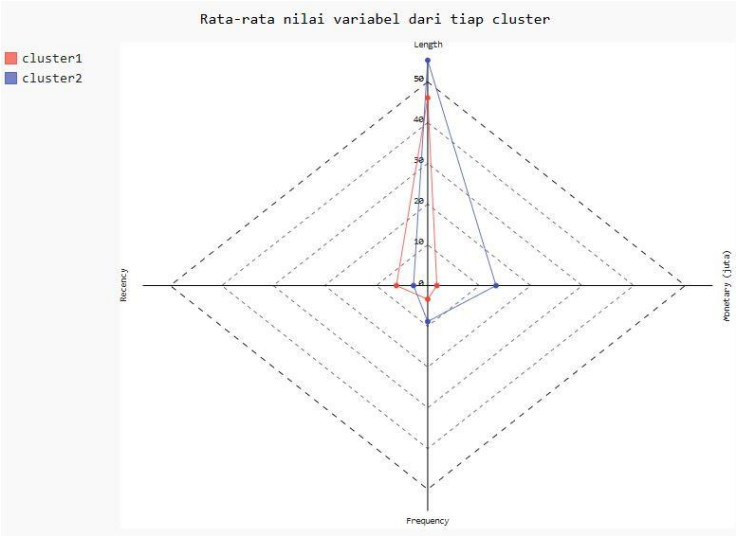


Gambar 6.3 Perbandingan Nilai 4 variabel

Perbandingan total nilai terhadap semua variabel untuk kedua segmen pada gambar 6.3 menunjukkan bahwa segmen 1 memiliki total nilai yang lebih besar, kecuali pada variabel *monetary*. Perbedaan paling signifikan terdapat pada variabel *length*, sedangkan pada variabel *monetary* segmen 2 memiliki total nilai lebih besar meskipun berjumlah jauh lebih sedikit.

6.5.2 Grafik *Radar*

Grafik *radar* menggambarkan perbandingan rata-rata dari nilai keempat variabel pada segmen 1 dengan segmen 2, agar dapat memahami karakteristik dari tiap segmen secara lebih baik dan detil. Hasil grafik radar dapat dilihat pada gambar 6.4.



Gambar 6.4 Grafik Radar

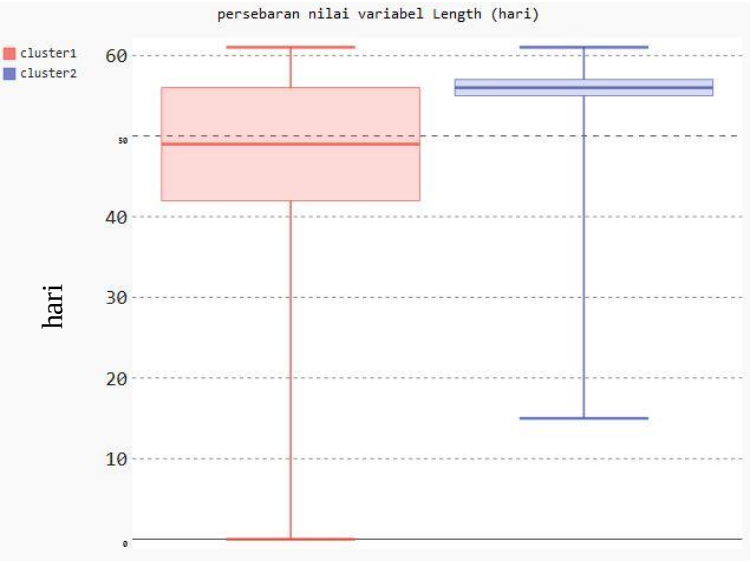
Grafik radar pada gambar 6.4 menunjukkan rata-rata nilai dari tiap variabel pada setiap segmennya. Saat diamati sangat terlihat perbedaan nilai dari kedua segmen, dimana segmen kedua selalu lebih unggul pada keempat dimensi tersebut. Satu-satunya variabel yang menunjukkan segmen 1 lebih besar adalah variabel *recency*, dimana variabel *recency* sendiri menunjukkan nilai yang lebih baik untuk nilai yang lebih kecil. Sedangkan nilai variabel yang paling signifikan perbedaannya adalah *monetary*.

6.5.3 Grafik *Box-plot*

Grafik *box-plot* menunjukkan persebaran nilai suatu variabel, dengan menunjukkan nilai minimum, kuartil 1, kuartil 2, kuartil 3, serta nilai maksimum. Grafik ini dibuat terhadap keempat nilai variabel pada setiap *cluster* guna menganalisa lebih lanjut tentang nilai atau performa dari pelanggan.

6.5.3.1 Grafik *Box-plot Length*

Gambar 6.3 adalah grafik *box-plot* untuk variabel *length.* Segmen 1 memiliki ukuran kotak yang lebih panjang dari pada segmen 2, dan lebih lanjut lagi rentang nilai pada segmen 1 juga lebih panjang dari pada segmen 2.

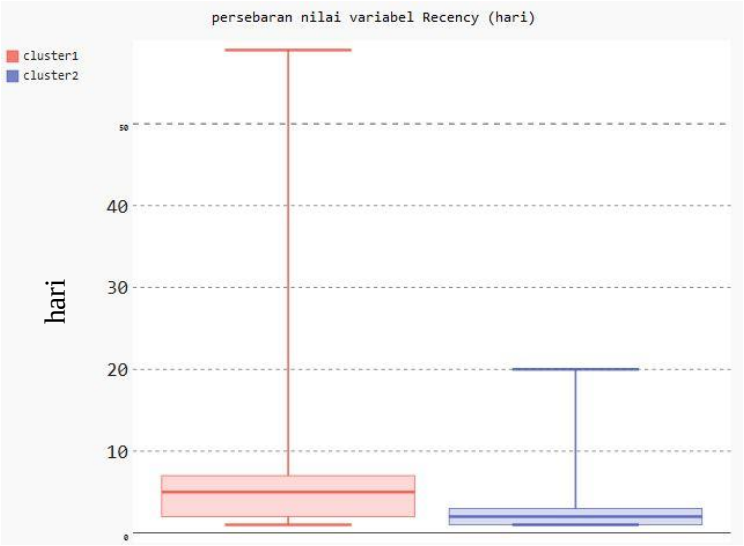


Gambar 6.5 Box-plot Length

Grafik pada gambar 6.6 menunjukkan bahwa sebaran data pada kedua segmen besar, yang artinya nilai tersebar dengan tidak merata. Sehingga dapat disimpulkan sebaran data pada segmen 1 lebih besar dan rentang nilai variabel *length* tersebut juga lebih besar.

6.5.3.2 Grafik *Box-plot Recency*

Grafik *box-plot recency* ditampilkan pada gambar 6.7. Grafik tersebut menggambarkan persebaran nilai variabel pada kedua segmen.

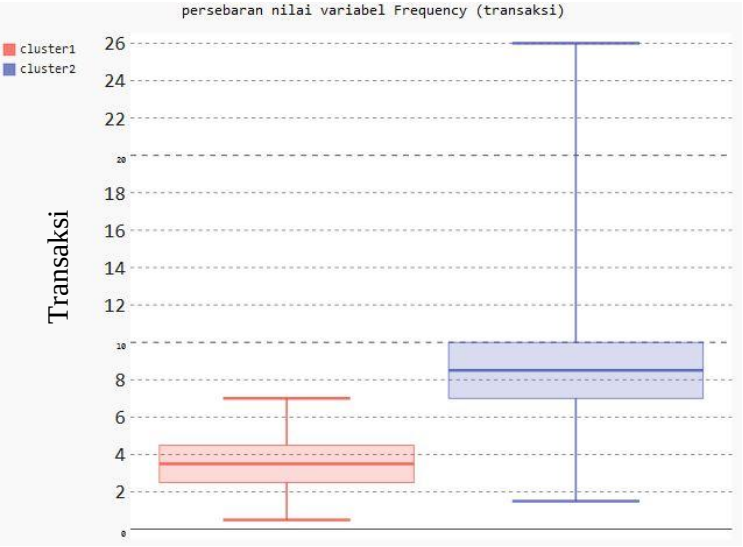


Gambar 6.6 Box-plot Recency

Pada gambar 6.7 dapat terlihat bahwa kotak segmen 1 lebih panjang dari pada segmen 2, hal ini menunjukkan sebaran data segmen 1 lebih besar. Selain sebaran data yang lebih besar rentang nilai yang dimiliki segmen 1 juga lebih lebar. Namun jika dilihat lebih detil garis median segmen 1 cenderung terletak diatas. Sehingga persebaran dari kuartil 1 hingga kuartil 2 lebih jauh dari pada kuartil 2 menuju kuartil 3.

6.5.3.3 Grafik *Box-plot Frequency*

Gambar 6.8 menunjukkan tampilan untuk grafik *box-plot* dengan nilai dari variabel *frequency*. Dapat diamati bahwa terdapat ketimpangan rentang nilai yang cukup jauh.

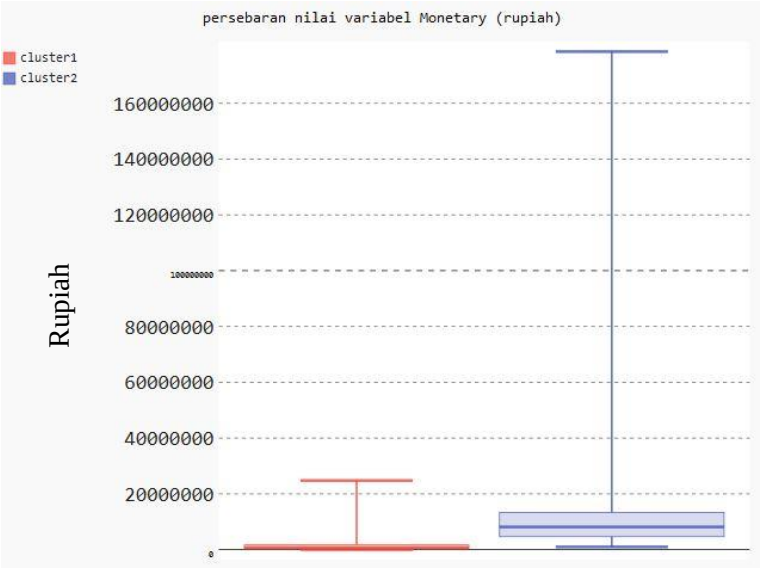


Gambar 6.7 Box-plot Frequency

Segmen 2 memiliki ukuran kotak yang lebih panjang dari pada segmen 1. Hal ini menunjukkan bahwa sebaran data pada segmen 2 lebih besar, dan jika melihat pada batas atas dan batas bawah nilai grafik maka segmen 2 juga memiliki rentang nilai yang jauh lebih besar dari pada segmen 1. Selain itu pusat sebaran nilai pada segmen 2 juga memiliki nilai yang lebih baik.

6.5.3.4 Grafik *Box-plot Monetary*

Gambar 6.9 adalah grafik *box-plot* untuk nilai variabel *monetary*. Jika diamati maka terdapat perbedaan nilai yang cukup jauh antara kedua *cluster*.



Gambar 6.8 Box-plot Monetary

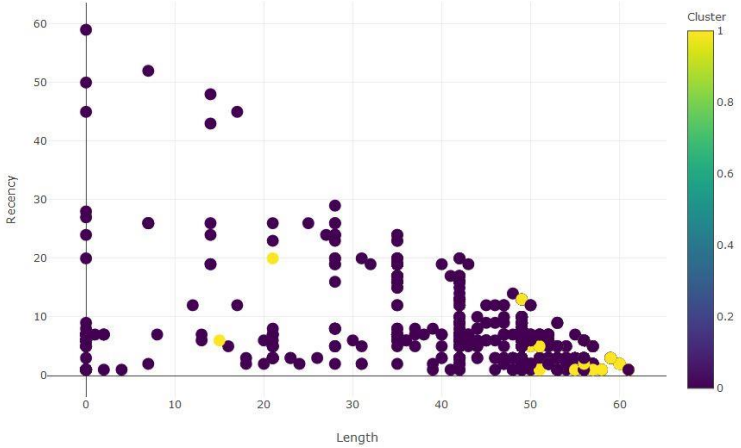
Seperti pada variabel *frequency* pada nilai variabel *monetary* pun ukuran kotak pada segmen 2 jauh lebih panjang dibandingkan segmen 1. Hal ini menunjukkan lebih baiknya segmen 2 dalam hal *monetary*, bahkan tidak hanya pusat sebaran nilai yang lebih baik namun rentang nilai yang dimiliki pun menunjukkan angka yang lebih baik pula.

6.5.4 Grafik *Scatter Plot* 2D

Grafik *scatter plot 2D* menunjukkan persebaran data konsumen dengan membandingkan nilai segmen berdasarkan dua variabel pada tiap grafiknya.

6.5.4.1 Scatter Plot 2D LR

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *length* dan *recency* disajikan pada gambar 6.9.

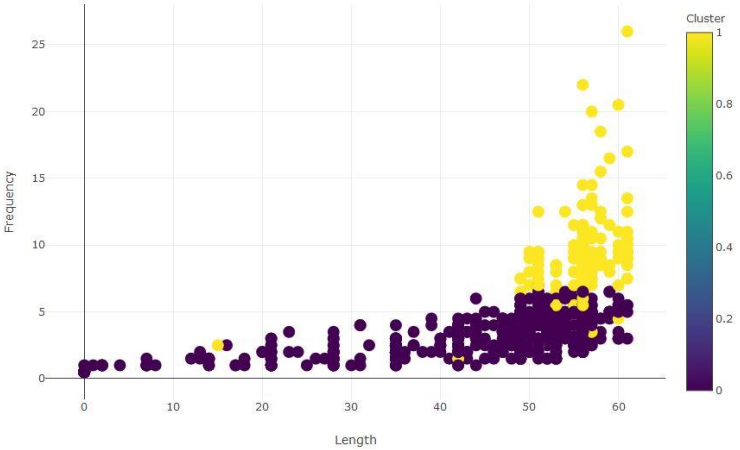


Gambar 6.9 Scatter 2D LR

Pada gambar 6.9 dapat diketahui bahwa segmen pertama berjumlah jauh lebih banyak, serta kecenderungannya untuk memiliki nilai *recency* yang lebih besar dan nilai yang lebih tersebar. Sedangkan segmen kedua cenderung berkumpul pada area kanan bawah, yang menunjukkan nilai *length* besar dan *recency* kecil.

6.5.4.2 Scatter Plot 2D LF

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *length* dan *frequency* disajikan pada gambar 6.10.

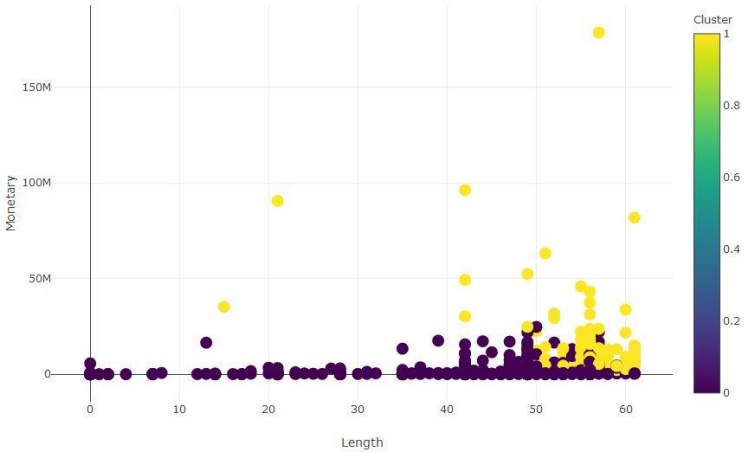


Gambar 6.10 Scatter Plot 2D LF

Gambar 6.10 menunjukkan persebaran data pada kedua segmen, secara umum segmen pertama cenderung memiliki nilai *frequency* yang lebih rendah sedangkan untuk *length* memiliki nilai merata dari kecil hingga besar. Sedangkan segmen kedua memiliki kecenderungan terkumpul pada nilai *length* yang tinggi dan nilai *frequency* yang lebih tinggi dari segmen pertama.

6.5.4.3 Scatter Plot 2D LM

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *length* dan *monetary* disajikan pada gambar 6.11

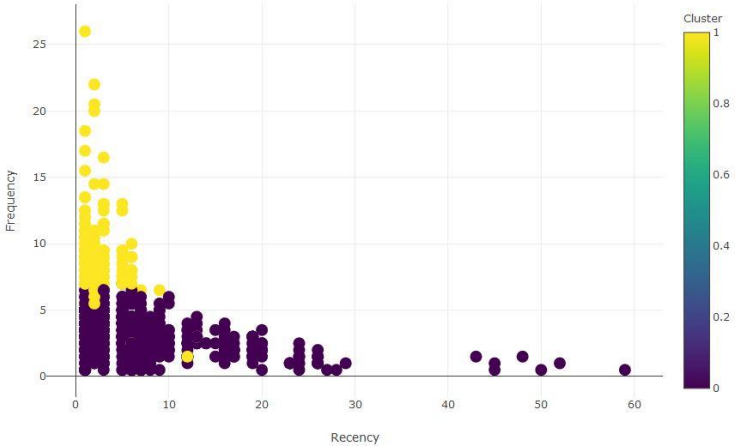


Gambar 6.11 Scatter Plot LM

Gambar 6.11 menunjukkan persebaran nilai variabel *length* dan *monetary*. Segmen pertama memiliki persebaran nilai yang rendah untuk *monetary* terbukti dengan posisi sumbu y yang rendah, sedangkan segmen kedua cenderung memiliki nilai *monetary* yang tinggi dan *length* yang besar.

6.5.4.4 Scatter Plot 2D RF

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *recency* dan *frequency* disajikan pada gambar 6.12.

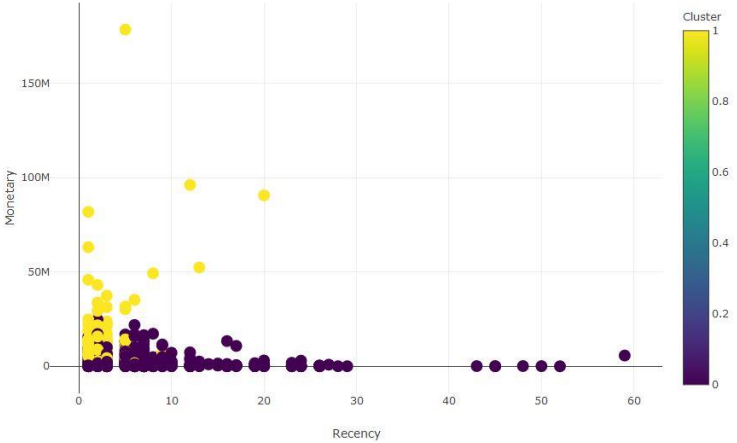


Gambar 6.12 Scatter Plot RF

Segmen pertama cenderung memiliki nilai *frequency* rendah dan tersebar pada variabel *recency*. Sedangkan segmen kedua cenderung lebih tinggi dalam hal *frequency* dan nilai *recency* yang rendah pula.

6.5.4.5 Scatter Plot 2D RM

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *recency* dan *monetary* disajikan pada gambar 6.13.

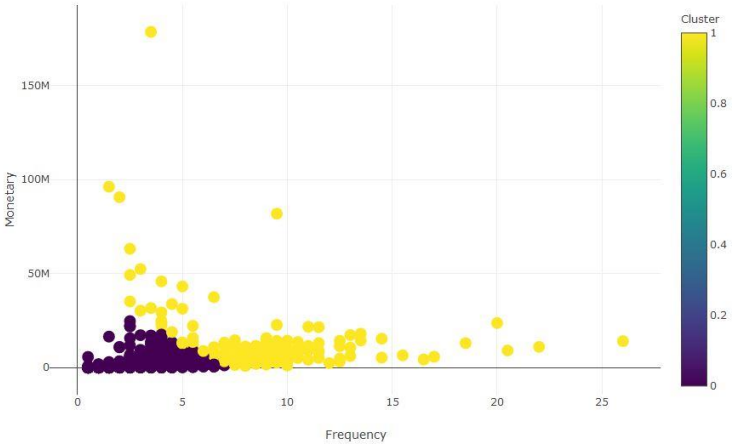


Gambar 6.13 Scatter Plot RM

Persebaran data untuk variabel *recency* dan *monetary* menunjukkan data yang cenderung terkumpul pada area kiri bawah diagram untuk kedua segmen. Yang menjadi pembeda adalah segmen kedua yang memiliki persebaran lebih besar dalam variabel *monetary* sehingga dapat disimpulkan nilai *monetary* milik segmen kedua lebih baik.

6.5.4.6 Scatter Plot 2D FM

Grafik *scatter plot 2D* yang menggambarkan persebaran data pelanggan berdasarkan variabel *frequency* dan *monetary* disajikan pada gambar 6.14



Gambar 6.14 Scatter Plot FM

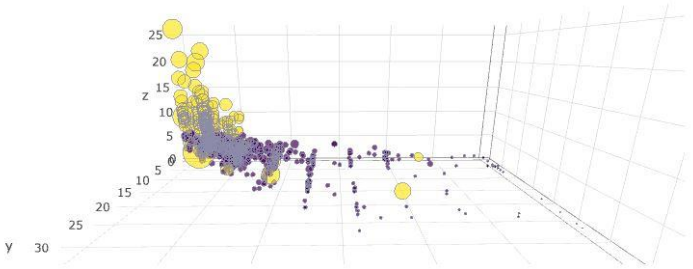
Pada gambar 6.14 ditampilkan persebaran nilai untuk kedua segmen. Segmen kedua cenderung lebih baik dalam kedua variabel yang dibandingkan, yaitu persebaran nilai *frequency* yang jauh lebih baik serta nilai *monetary* yang lebih tinggi.

6.5.5 Grafik *Scatter Plot* 3D

Grafik *scatter plot 3D* menunjukkan persebaran nilai objek pada ruang tiga dimensi dengan sumbu x,y, dan z merupakan kombinasi dari empat variabel LRFM, sehingga terbentuk empat grafik 3 dimensi. Keempat grafik yang ditampilkan meliputi grafik perbandingan antara variabel *length, recency, frequency* lalu *length, recency, monetary* lalu *length, frequency, monetary* dan *recency, frequency, monetary*. Lalu untuk lingkaran-lingkaran di dalam grafik merupakan representasi dari setiap pelanggan, dengan deskripsi bahwa ukuran lingkaran bergantung pada besar atau kecilnya nilai clv pelanggan tersebut. Sedangkan warna cerah menggambarkan segmen kedua dan warna yang lebih gelap menunjukkan segmen pertama.

6.5.5.1 Grafik 3D *Length, Recency, Frequency*

Gambar 6.10 menunjukkan tampilan grafik *scatter plot 3D* untuk variabel *length, recency, frequency*.

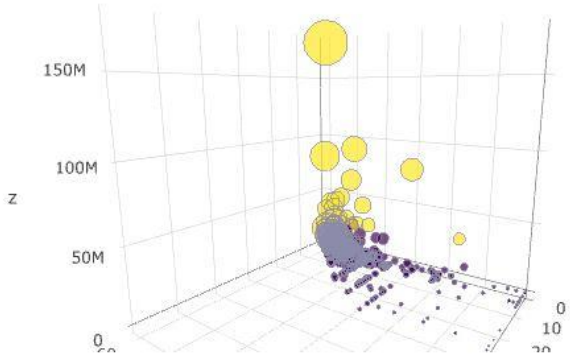


Gambar 6.15 Scatter Plot LRF

Pada gambar 6.10 dapat dilihat bahwa ukuran lingkaran segmen kedua selalu cenderung lebih besar namun berjumlah sedikit, dan segmen pertama berjumlah jauh lebih banyak namun berukuran kecil. Variabel yang paling menunjukkan kelebihan segmen kedua adalah sumbu z atau *frequency*. Sedangkan nilai objek sangat banyak terkumpul pada nilai sumbu x dan y yang baik dan nilai sumbu z rendah.

6.5.5.2 Grafik 3D *Length, Recency, Monetary*

Gambar 6.11 menunjukkan tampilan grafik *scatter plot 3D* untuk variabel *length, recency, monetary*.

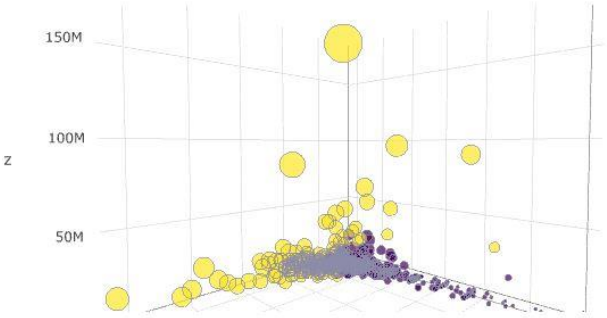


Gambar 6.16 Scatter Plot LRM

Seperti yang sudah diketahui sebelumnya dapat diamati bahwa ukuran lingkaran segmen kedua cenderung lebih besar dan memiliki nilai yang lebih baik. Dalam grafik ini sangat terlihat perbedaan nilai segmen pertama dan kedua dalam hal *monetary*, atau pada sumbu z.

6.5.5.3 Grafik 3D *Length, Frequency, Monetary*

Gambar 6.12 menunjukkan tampilan grafik *scatter plot 3D* untuk variabel *length, frequency, monetary*.

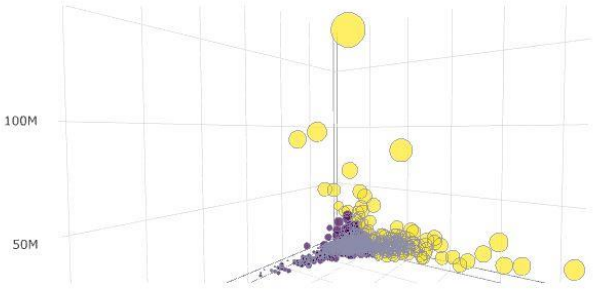


Gambar 6.17 Scatter Plot LFM

Gambar 6.12 menunjukkan persebaran data pada segmen pertama dan kedua. Jika diamati mungkin akan terlihat bahwa nilai objek pada segmen pertama cenderung lebih terkumpul pada sekitaran sumbu x, yang berarti memiliki nilai rendah pada sumbu y (*frequency*) dan z (*monetary*). Sedangkan segmen kedua cenderung lebih tersebar dan memiliki nilai yang lebih baik dalam sumbu y dan z.

6.5.5.4 Grafik 3D *Recency, Frequency, Monetary*

Gambar 6.13 menunjukkan tampilan grafik *scatter plot 3D* untuk variabel *recency*, *frequency*, *monetary*.



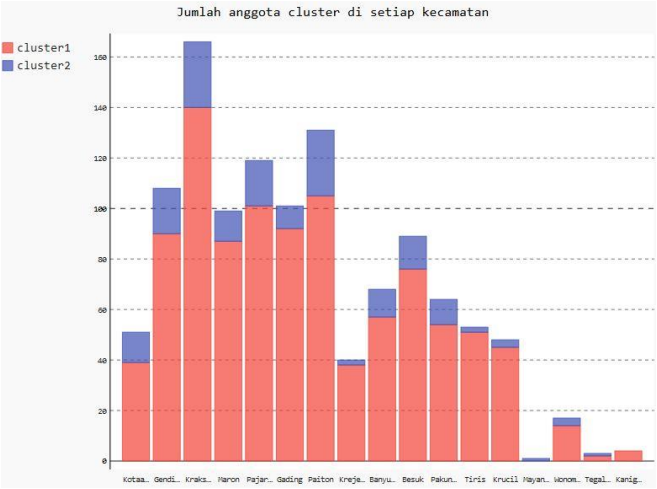
Gambar 6.18 Scatter Plot RFM

Gambar 6.13 menunjukkan persebaran data pada segmen pertama dan kedua. Dapat diamati bahwa nilai segmen pertama cenderung lebih tinggi dalam nilai sumbu x (*recency*) yang berarti memiliki nilai *recency* yang lebih buruk. Sedangkan segmen kedua cenderung lebih baik dalam sebaran nilai sumbu y (*frequency*) dan z (*monetary*).

6.5.6 Grafik *Stacked Bar*

Grafik *stacked bar* merupakan pengembangan dari *bar chart*. Pada penelitian ini grafik *stacked bar* digunakan untuk menganalisis hubungan antara *cluster* pelanggan dengan lokasi

pelanggan berada, dalam hal ini lokasi yang dimaksud adalah kecamatan. Nilai yang ditampilkan pada sumbu y merupakan akumulasi atau jumlah dari kedua segmen sedangkan sumbu x merupakan nama untuk tiap kecamatan tempat pelanggan berada. Untuk mengetahui hasil grafik *stacked bar* dapat dilihat pada gambar 6.14.

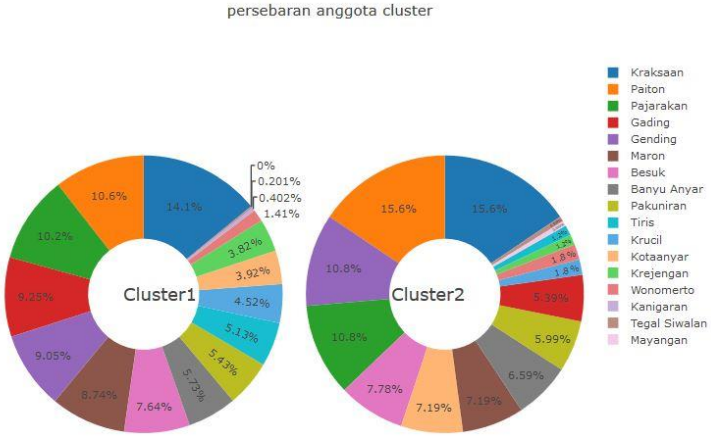


Gambar 6.19 Stacked Bar Lokasi Cluster

Dapat dilihat bahwadi setiap kecamatan segmen pertama selalu mendominasi jumlah segmen, kecuali pada satu kecamatan saja yaitu kecamatan Mayangan. Sedangkan kecamatan yang mendominasi jumlah total pelanggan adalah Kecamatan Kraksaan dengan jumlah lebih dari 160 pelanggan, lalu disusul Kecamatan Paiton dengan jumlah sekitar 130 pelanggan. Hal itu memang wajar terjadi mengingat Kraksaan adalah ibu kota dan pusat pemerintahan untuk wilayah Kabupaten Probolinggo. Sedangkan Paiton merupakan wilayah yang terkenal dengan industri pembangkit listriknya.

6.5.7 Grafik *Donut Chart*

Grafik *donut chart* pada dasarnya mirip dengan *pie chart*, hanya saja terdapat lubang ditengah grafik. Grafik *donut chart* ini digunakan untuk menunjukkan persentase jumlah anggota berdasarkan lokasi pada setiap segmen yang ada. Untuk mengetahui hasil grafik *stacked bar* dapat dilihat pada gambar 6.15.



Gambar 6.20 Donut Chart Lokasi Cluster

Pada gambar 6.15 dapat diamati bahwa anggota *cluster* selalu didominasi pelanggan yang berasal dari Kecamatan Kraksaan dan Paiton. Sedangkan daerah dengan penyumbang anggota paling sedikit adalah Kecamatan Mayangan.

6.6 Analisis Perbandingan

Pada tahap ini dilakukan perbandingan untuk berbagai variasi hasil *cluster*. Percobaan dilakukan dengan mengganti jumlah K yang awalnya ditentukan dua segmen menjadi tiga dan empat segmen. Selanjutnya dilakukan pula percobaan untuk menghilangkan bobot variabel yang kemungkinan dapat menjadi bias untuk proses *clustering*.

6.6.1 Perbandingan Jumlah Segmen Dengan Bobot

Percobaan pertama dilakukan dengan menggunakan jumlah segmen tiga dan empat serta menggunakan bobot. Dengan melakukan variasi jumlah *cluster* maka didapatkan persebaran anggota *cluster* yang dapat dilihat pada tabel 6.15.

Tabel 6.15 Perbandingan Jumlah Anggota

K	Persebaran anggota			
	segmen 1	**segmen 2**	**segmen 3**	**segmen 4**
2	995	167	-	-
3	1006	147	9	-
4	140	196	9	817

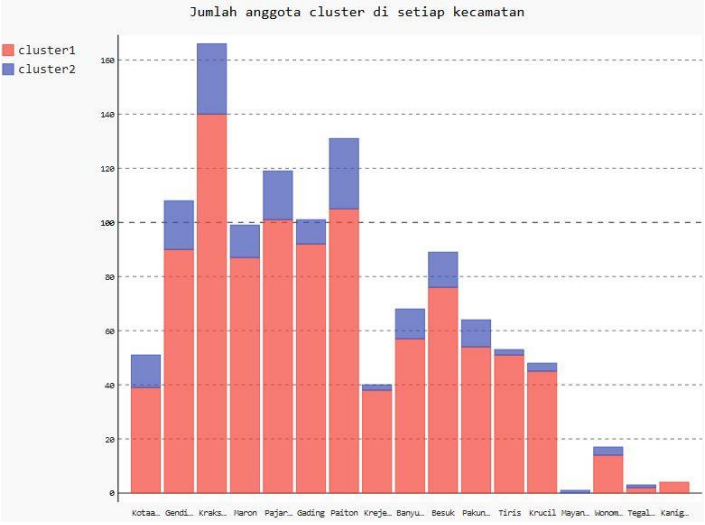
Dari tabel 6.15 dapat ditarik kesimpulan sederhana bahwa dengan menggunakan bobot akan selalu terdapat segmen yang memiliki anggota jauh lebih banyak dibanding yang lainnya. Selanjutnya dilakukan juga analisis karakteristik tiap segmen untuk setiap percobaan jumlah K yang berbeda. Hasil karakteristik segmen disajikan pada tabel 6.16.

Tabel 6.16 Karakteristik Segmen Setiap K

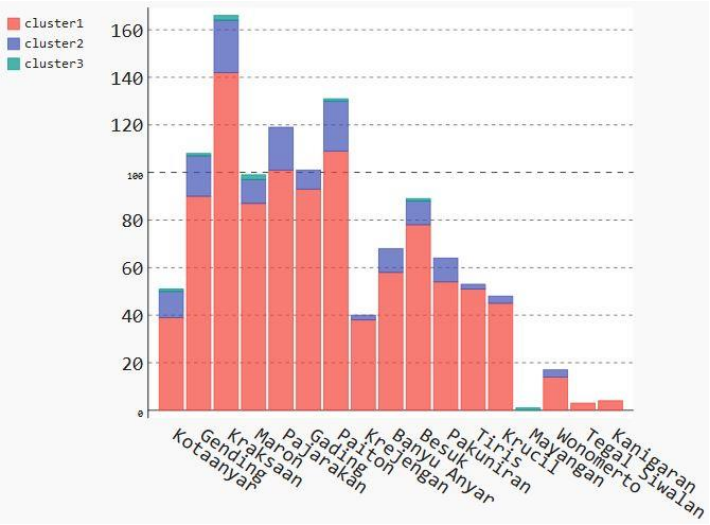
K	Karakteristik Segmen (rata-rata nilai)				
	Segmen	**L**	**R**	**F**	**M (juta)**
2	**1**	46.09	6	3.3	1.78
	2	55.3	2.75	8.8	**13.26**
3	**1**	46.12	6.04	3.35	2.04
	2	56.22	2.52	9.52	8.38
	3	48.22	7	3.72	**77.9**
4	**1**	56.33	2.42	9.68	8.43
	2	24.02	12.3	1.56	0.68
	3	48.22	7	3.72	**77.9**
	4	51.5	4.52	3.81	2.41

Dari percobaan tersebut, hal yang mudah disoroti adalah perbedaan yang mencolok pada variabel *monetary*. Dimana pada setiap percobaan jumlah K selalu terdapat perbedaan yang jauh pada salah satu segmen. Pada nilai k=2 hal itu terjadi pada segmen 2, sedangkan untuk k=3 pada segmen 3, dan untuk k=4 juga pada segmen 3. Lalu ketika diperhatikan lebih lanjut segmen 3 pada k=3 dan segmen 3 pada k=4 beranggotakan pelanggan yang sama.

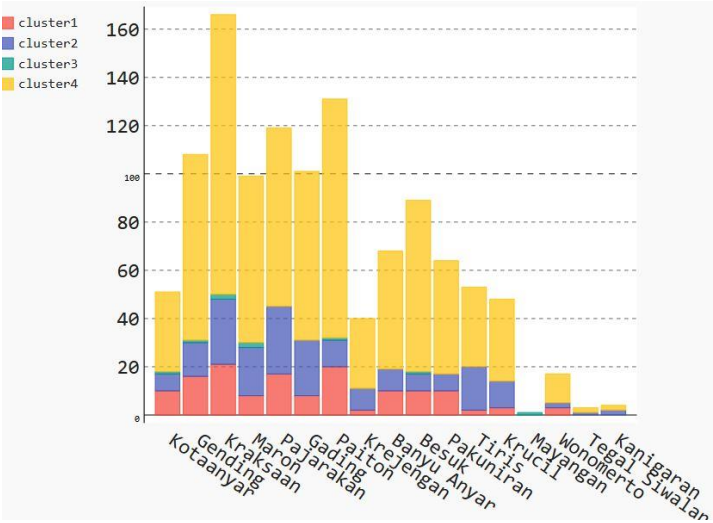
Selanjutnya dilakukan analisis tentang persebaran lokasi pelanggan pada setiap segmen. Persebaran lokasi pelanggan untuk k=2 ditampilkan pada gambar 6.21, sedangkan untuk k=3 pada gambar 6.22, dan untuk k=4 disajikan pada gambar 6.23.



Gambar 6.21 Persebaran Lokasi Pelanggan 2 segmen



Gambar 6.22 Persebaran Lokasi 3 Segmen



Gambar 6.23 Persebaran Lokasi 4 Segmen

Dapat terlihat bahwa persebaran segmen selalu dominan pada salah satu segmen saja.

6.6.2 Perbandingan Jumlah Segmen Tanpa Bobot

Selanjutnya dilakukan pula percobaan dengan menghilangkan faktor bobot dalam proses *clustering*. Percobaan tanpa menggunakan bobot ini dilakukan dengan jumlah K mulai dari tiga, empat, dan lima. Perbedaan percobaan jumlah K dilakukan karena tanpa menggunakan bobot rekomendasi jumlah segmen berdasarkan metode *elbow* adalah lima segmen. Perbandingan jumlah anggota segmen untuk setiap percobaan nilai K dapat dilihat pada tabel 6.17.

Tabel 6.17 Perbandingan Jumlah Anggota

K	Persebaran anggota				
	segmen 1	**segmen 2**	**segmen 3**	**segmen 4**	**segmen 5**
3	508	105	549	-	-
4	197	753	82	130	-
5	429	60	429	124	120

Berdasarkan tabel 6.17 dapat diamatai bahwa persebaran anggota tampak lebih merata, meskipun masih terdapat segmen yang memiliki anggota dalam jumlah sedikit. Selanjutnya dilakukan juga analisis karakteristik tiap segmen untuk setiap percobaan jumlah K yang berbeda. Hasil karakteristik segmen disajikan pada tabel 6.18.

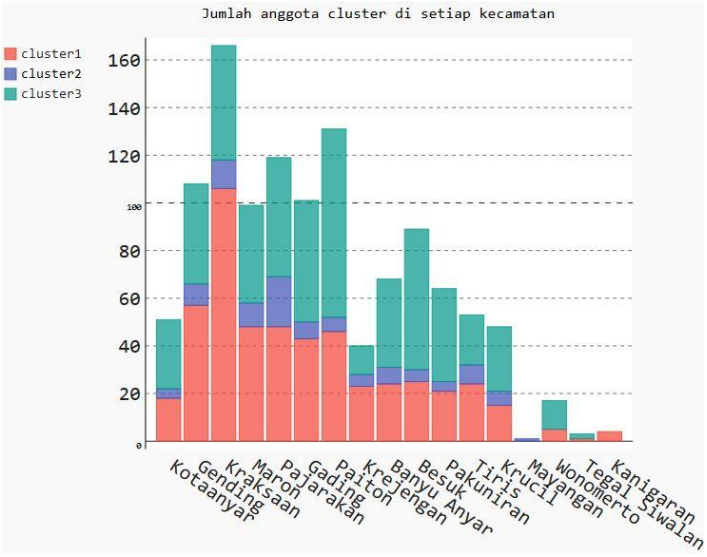
Tabel 6.18 Karakteristik Segmen tanpa Bobot

K	Karakteristik Segmen (rata-rata nilai)				
	Segmen	**L**	**R**	**F**	**M (juta)**
3	**1**	45.63	7.29	3.05	2.07
	2	12.6	13	1.21	1.74
	3	55.73	2.62	5.7	5.01
4	**1**	37.84	10.89	2.38	2.12

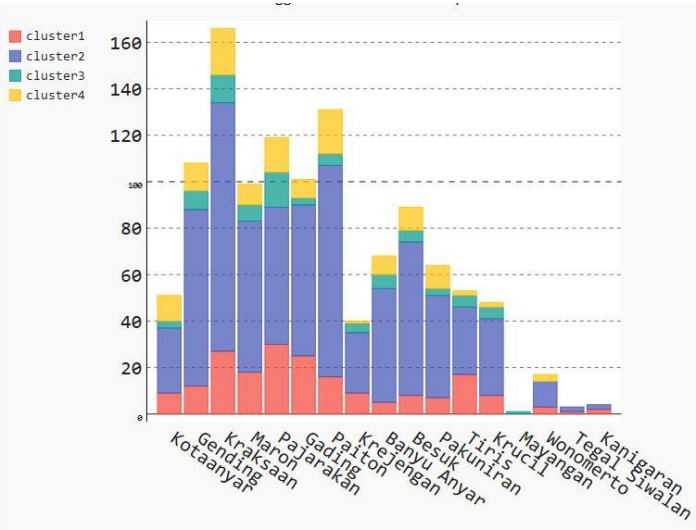
K	Karakteristik Segmen (rata-rata nilai)				
	Segmen	**L**	**R**	**F**	**M (juta)**
	2	52.56	4.09	3.93	2.79
	3	8.5	11.98	1.06	2.08
	4	56.71	2.3	9.92	9.99
5	**1**	47.41	6.23	4.71	2.39
	2	4.05	13.8	0.84	1.05
	3	55.5	2.69	4.39	3.92
	4	31.43	12.78	2.06	1.5
	5	56.73	2.26	10.19	8.58

Proses *clustering* tanpa menggunakan bobot menghasilkan segmen-segmen yang lebih merata, terutama dalam variabel *monetary*. Hal ini dibuktikan dengan data pada tabel 6.18 yang menunjukkan tidak adanya ketimpangan nilai yang besar, terutama pada nilai *monetary*.

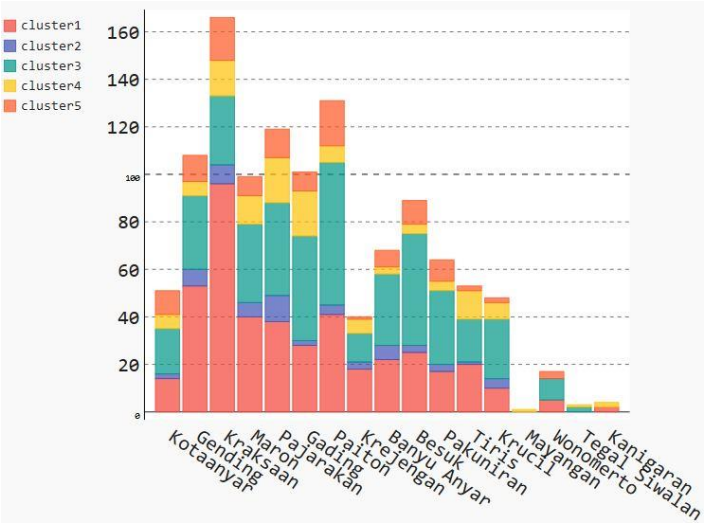
Selanjutnya dilakukan analisis tentang persebaran lokasi pelanggan pada setiap segmen. Persebaran lokasi pelanggan untuk k=3 ditampilkan pada gambar 6.24, sedangkan untuk k=4 pada gambar 6.25, dan untuk k=5 disajikan pada gambar 6.26.



Gambar 6.24 Persebaran 3 Segmen Tanpa Bobot



Gambar 6.25 Persebaran 4 Segmen Tanpa Bobot



Gambar 6.26 Persebaran 5 Segmen Tanpa Bobot

Dari gambar 6.24, 6.25, dan 6.26 dapat diamati bahwa secara umum persebaran segmen lebih merata untuk setiap daerah dan tidak terlalu didominasi oleh salah satu segmen saja. Lebih meratanya hasil *clustering* dalam persebaran dan nilai variabelnya dimungkinkan terjadi karena penghilangan bobot variabel. Bobot pada variabel dapat menjadi bias untuk mengelompokkan pelanggan, karena nilai bobot yang sangat besar pada *monetary* sehingga kecenderungan untuk membedakan pelanggan sangat bergantung pada nilai *monetary* pelanggan tersebut.

6.7 Kesimpulan Analisis

Dari proses *clustering* dan analisis visualisasi yang telah dilakukan, diperoleh rangkuman sebagai berikut :

1. Hasil AHP menunjukkan bahwa variabel yang paling penting adalah *monetary* dengan bobot tertinggi yaitu 0.598 atau lebih dari setengah total bobot. Dengan ini dapat disimpulkan bahwa variabel *monetary* akan selalu dijadikan pertimbangan paling penting dalam proses

clustering. Sedangkan ketiga variabel lainnya yaitu *length*, *recency*, dan *frequency* memiliki bobot berturut-turut 0.074, 0.050, dan 0.279.

2. Berdasarkan metode *elbow*, ditemukan bahwa nilai K terbaik adalah 2, sehingga proses *clustering* dilakukan dengan parameter K=2 dan menghasilkan 2 *cluster*.
3. Dari 1162 pelanggan yang diproses untuk *clustering*, 995 pelanggan tergabung pada *cluster* 1, dan 167 pelanggan tergabung pada *cluster* 2.
4. *Cluster* pertama memiliki rata-rata nilai CLV lebih rendah dari pada *cluster* kedua, yaitu 0.137935535. Bahkan rata-rata nilai keempat variabel *cluster* pertama selalu lebih buruk dari pada *cluster* kedua. *Cluster* pertama sendiri sebagian besar berada di kecamatan Kraksaan dan Paiton, hal ini disebabkan Kraksaan merupakan ibu kota wilayah Kabupaten Probolinggo. Segmen pertama secara umum berisi pelanggan dengan skala bisnis yang tidak besar, atau bisnis rumahan. Maka dengan ini untuk memicu peningkatan frekuensi dan jumlah pembelian pelanggan dapat memberi penawaran promosi berupa *up-selling* atau *cross-selling*.
5. *Cluster* kedua memiliki rataan CLV tertinggi yaitu berada di angka 0.250248127. Hal ini didukung oleh fakta bahwa rerata nilai untuk keempat variabel yang dibandingkan selalu bernilai lebih baik dari pada *cluster* pertama. Untuk *cluster* kedua komposisi anggota terbanyak ada pada kecamatan Kraksaan dan Paiton yang secara berimbang masing-masing menyumbangkan 15.6% dari total anggota. Dengan karakteristik tersebut maka kemungkinan besar segmen kedua terdiri dari pelanggan dengan skala bisnis yang lebih besar. Sehingga ketika perusahaan akan membagikan bonus di tiap periodenya anggota dari *cluster* kedua akan selalu menjadi prioritas untuk memperolehnya.

6. Analisis perbandingan dengan menggunakan variasi percobaan jumlah segmen dan penggunaan bobot menghasilkan kesimpulan sementara bahwa bobot yang terlalu besar pada salah satu variabel dapat mengubah hasil *clustering* secara signifikan dan dapat dimungkinkan untuk memunculkan bias.
7. Tugas akhir ini menunjukkan hasil segmentasi yang didominasi pelanggan dengan kriteria jumlah transaksi yang cenderung sedikit dan kurang loyal. Sedangkan pelanggan yang lebih loyal dan menghasilkan banyak profit hanya berjumlah 14,37% saja.

BAB VII
KESIMPULAN DAN SARAN

Bab ini memaparkan kesimpulan yang diperoleh dari serangkaian pengerjaan tugas akhir yang telah dilakukan, serta memberikan saran untuk pengembangan tugas akhir dengan topik yang serupa.

7.1. Kesimpulan

Dari penelitian ini, dapat diperoleh beberapa kesimpulan yaitu:

1. Penggalian data dengan *clustering* dapat membantu pihak perusahaan dalam mengetahui segmen pelanggan yang dimiliki. Dengan menggunakan aplikasi hasil penelitian perusahaan dapat melakukan segmentasi pelanggan secara otomatis. Aplikasi mampu menghasilkan 2 *cluster* pelanggan dengan menggunakan empat variabel sebagai pertimbangan. Nilai CLV untuk segmen pertama adalah 0.138, sedangkan untuk segmen kedua adalah 0.250.
2. Hasil *clustering* menghasilkan 2 *cluster* dengan akurasi kesalahan atau tingkat *error* 1.5468083. Acuan tingkat kesalahan yang digunakan adalah SSE (*Sum Squared Error*).
3. Aplikasi dapat menghasilkan visualisasi *cluster* dalam bentuk *dashboard* yang berbasis web. Harapannya adalah pengguna bisa melihat hasil proses *clustering* secara mudah dan nyaman. Visualisasi juga bermanfaat untuk menunjang analisis dan proses pengambilan keputusan.
4. Penelitian ini menghasilkan analisis segmentasi dari *clustering* yang telah dilakukan. Analisis dilakukan berdasarkan beberapa kriteria, kriteria pertama adalah nilai dari keempat variabel yang dijadikan acuan untuk proses *clustering* dan kriteria kedua merupakan lokasi atau tempat pelanggan tersebut berada. Analisis tersebut

menghasilkan *cluster* kedua yang superior dan kecamatan Kraksaan serta Paiton yang menyumbangkan anggota terbanyak.

7.2. Saran

Selama pengerjaan penelitian ini, tentunya terdapat kekurangan yang dapat diperbaiki dan dikembangkan pada penelitian selanjutnya. Saran yang dapat dipertimbangkan yaitu:

1. Penelitian dapat dikembangkan dengan memperpanjang periode data yang digunakan untuk pengolahan, misalkan data dalam periode satu tahun.
2. Mengembangkan penelitian dengan menambahkan variabel yang dijadikan sebagai acuan, misalkan dapat menggunakan variabel pertumbuhan (*growth*), lokasi pelanggan, ataupun *cost* (harga yang dikeluarkan).
3. Dapat menggunakan *tools* atau perangkat lunak lain yang bisa mendukung proses penelitian, seperti *spyder* milik *anaconda*.
4. Visualisasi *cluster* yang digunakan dalam penelitian dapat menggunakan bahasa pemrograman R yang lebih kaya akan *library* untuk menciptakan grafik-grafik yang lebih bervariasi.

DAFTAR PUSTAKA

[1] T. Bohling, D. Bowman, and S. Lavalle, "CRM Implementation : Effectiveness Issues and Insights," *J. Serv. Res.*, vol. 9, no. 2, 2015.

[2] D. L. Goodhue, B. H. Wixom, and H. J. Watson, "REALIZING BUSINESS BENEFITS THROUGH CRM : HITTING THE RIGHT TARGET IN THE RIGHT WAY," vol. 1, no. 2, 2002.

[3] K. F. Sandi, "Unilever Indonesia," 2017. [Online]. Available: https://ekbis.sindonews.com/read/1215306/32/unilever-indonesia-bagikan-dividen-rp64-triliun-1497962738. [Accessed: 05-Feb-2018].

[4] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar, "Introduction to Data Mining," 2nd ed., 2005.

[5] M. Diana, J. David, and R. Cote, "Exploring the benefits of a traveller clustering approach based on multimodality attitudes and behaviours," *Transp. Res. Procedia*, vol. 25, pp. 2556–2569, 2017.

[6] M. Alvandi, S. Fazli, and F. S. Abdoli, "K-Mean Clustering Method For Analysis Customer Lifetime Value With LRFM Relationship Model In Banking Services," vol. 3, no. 11, pp. 2294–2302, 2012.

[7] W. J. Reinartz and V. Kumar, "On the Profitability of Long-Life Customers in a Noncontractual Setting : An Empirical Investigation and Implications for Marketing," vol. 64, no. October, pp. 17–35, 2000.

[8] Z. Kakushadze and W. Yu, "Biomolecular Detection and Quanti fi cation * K-means and cluster models for cancer signatures," *Biomol. Detect. Quantif.*, vol. 13, no.

August, pp. 7–31, 2017.

[9] Z. Dan, "Integrating RFM model and Cluster for Students Loan Subsidy Valuation," pp. 461–464, 2008.

[10] C. Mati and L. Ilie, "Customer relationship management in the insurance industry," vol. 15, no. 14, pp. 1138–1145, 2014.

[11] M. Khajvand, K. Zolfaghar, S. Ashoori, and S. Alizadeh, "Estimating customer lifetime value based on RFM analysis of customer purchase behavior : case study," *Procedia Comput. Sci.*, vol. 3, pp. 57–63, 2011.

[12] Francis Buttle, *Customer Relationship Management, Concept and Technologies*, 2nd ed. A Butterworth-Heinemann Title, 2009.

[13] Z. Lin, "Research on Customer Segmentation Model by Clustering," pp. 316–318, 2005.

[14] B. Cooil, L. Aksoy, and L. Timothy, "Approaches to Customer Segmentation," no. November 2014, pp. 37–41.

[15] J. Han and M. Kamber, *Data Mining: Concepts and Technique 2nd edition*. 2009.

[16] K. Vadim, "Overview of different approaches to solving problems of Data Mining," *Procedia Comput. Sci.*, vol. 123, pp. 234–239, 2018.

[17] A. K. Jain, "Data clustering : 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010.

[18] V. Aggelis, "Customer Clustering using RFM analysis," pp. 1–5.

[19] R. Shanker, “Segmentation of Tumor and Edema Based on K-mean clustering and Hierarchical Centroid Shape Descriptor,” pp. 1105–1109, 2017.

[20] J. Wei, S. Lin, C. Weng, and H. Wu, “A case study of applying LRFM model in market segmentation of a children ’ s dental clinic,” *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5529–5533, 2012.

[21] Z. Liu, “A method of SVM with Normalization in Intrusion Detection,” *Procedia Environ. Sci.*, vol. 11, pp. 256–262, 2011.

[22] T. M. Kodinariya, “Review on determining number of Cluster in K-Means Clustering,” vol. 1, no. 6, pp. 90–95, 2013.

[23] M. Ivanco, G. Hou, and J. Michaeli, “Sensitivity analysis method to address user disparities in the analytic hierarchy process,” *Expert Syst. Appl.*, vol. 90, pp. 111–126, 2017.

[24] S. I. Gass and M. C. Fu, *Encyclopedia of Operations Research and Management*. Kluwer Academic.

[25] H. Li, F. Ni, Q. Dong, and Y. Zhu, “Application of analytic hierarchy process in network level pavement maintenance decision-making,” *Int. J. Pavement Res. Technol.*, 2017.

[26] L. Ashraf, “Introduction to the Analytic Hierarchy Process 3.1,” in *Learning From Failures*, 2014, pp. 33–44.

[27] B. Sohrabi and A. Khanlari, “Customer Lifetime Value (CLV) Measurement Based on RFM Model,” *Iran. Account. Audit. Rev.*, vol. 14, no. 47, pp. 7–20, 2007.

[28] R. J. Ã and T. Eltoft, “A new information theoretic analysis of sum-of-squared-error kernel clustering,” vol. 72, pp. 23–31, 2008.

BIODATA PENULIS

Penulis bernama lengkap Adam Ardiansyah, lahir di Jombang, 14 Agustus 1995. Penulis merupakan anak ketiga dari empat bersaudara. Penulis mengenyam pendidikan formal Sekolah Dasar di SD Negeri Jombatan III Jombang dan melanjutkan ke jenjang Sekolah Menengah Pertama di SMP Negeri 1 Jombang. Lulus tahun 2011, penulis melanjutkan di SMA Negeri 2 Jombang. Tiga tahun berselang atau setelah lulus SMA, penulis mendaftar kuliah di Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis aktif berorganisasi dan mengikuti kepanitiaan di tingkat fakultas, maupun institut. Penulis terdaftar sebagai staf *Sponsorship, Information System Expo 2015;* staf *Media dan Informasi* Kajian Islam Sistem Informasi 2015/2016; staf Dewan Perwakilan Mahasiswa 2016; dan Sekretaris Biro Kajian Strategis Himpunan Mahasiswa Sistem Informasi 2016/2017. Dalam pengerjaan Tugas Akhir, penulis memilih laboratorium Rekayasa Data dan Intelegensi Bisnis, dengan bimbingan Bapak Edwin Riksakomara, S.Kom., M.T., dengan topik mengenai penggalian data. Penulis dapat dihubungi melalui email admardnsyh@gmail.com untuk keperluan penelitian. Semoga penulisan Tugas Akhir ini dapat memberikan kontribusi positif bagi semua pihak terkait.

Halaman ini sengaja dikosongkan

LAMPIRAN A

KUESIONER SURVEY PEMERINGKATAN KEPENTINGAN KRITERIA PERILAKU PELANGGAN

Kuesioner ini bertujuan untuk mendapatkan masukan pendapat dari Narasumber terkait dengan pemeringkatan kriteria yang akan digunakan dalam pembobotan variabel segmentasi pelanggan. Kami menjamin kerahasiaan data/informasi yang diberikan serta tidak akan membawa dampak apapun bagi responden..

IDENTITAS RESPONDEN

Nama : _____

Jabatan : _____

PETUNJUK PENGISIAN KUESIONER

1. Berikan tanda centang (√) pada kriteria yang menurut persepsi bapak/ibu lebih penting dibanding dengan kriteria lainnya, dengan cara membandingkan krieria pada sebelah kiri dengan kriteria pada sebelah kanan. Kriteria yang dimaksud adalah perilaku pelanggan yang melakukan transaksi. Skala numerik akan menunjukkan suatu perbandingan dari tingkat kepentingan dua kriteria dengan penjelasan setiap skalanya yaitu:

Skala Nilai	Tingkat Preferensi	Penjelasan
1	Sama Pentingnya	Kedua kriteria memiliki pengaruh yang sama
3	Sedikit lebih penting	Penilaian dan pengalaman sedikit memihak pada salah satu kriteria tertentu dibanding kriteria pasangannya
5	Lebih penting	Penilaian dan pengalaman memihak pada salah satu kriteria tertentu disbanding kriteria pasangannya
7	Jelas lebih penting	Salah satu kriteria lebih diprioritaskan dan relatif lebih penting dibandingkan kriteria pasangannya.

A-2

Skala Nilai	Tingkat Preferensi	Penjelasan
9	Mutlak sangat penting	Salah satu kriteria sangat jelas lebih penting dibandingkan kriteria pasangannya.
2,4,6,8		Diberikan bila ada keraguan penilaian diantara dua tingkat kepentingan yang berdekatan.

2. Jika kriteria pada sebelah kiri lebih penting dibandingkan dengan kriteria sebelah kanan, maka pilih skala yang lebih dekat pada kriteria sebelah kiri, begitu pula sebaliknya.
3. Kriteria pelanggan yang dibandingkan yaitu :
   a. *Recency* : jarak dari waktu transaksi terakhir kali dilakukan dengan waktu saat ini.
   b. *Frequency :* total jumlah transaksi yang dilakukan selama periode tertentu
   c. *Monetary* : total nilai produk dalam bentuk uang yang diberikan pelanggan pada perusahaan
   d. *Length* : Jarak waktu (hari) dari transaksi pertama dengan transaksi terakhir dalam periode data yang telah ditentukan

CONTOH PENGISIAN KUESIONER

Kriteria																		Kriteria
Recency	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Frequency*
													√					

Artinya :
Pada pernyataan tersebut, tanda centang (√) di kolom jawaban sebelah kiri menunjukan bahwa: *recency* 1/5 kali lebih penting dibandingkan dengan f*requency* dalam hal peringkatan kriteria untuk retailer. Dengan kata lain *frequency* 5 kali lebih penting dibandingkan dengan r*ecency* dalam hal pemeringkatan kriteria untuk pelanggan.

DAFTAR PERTANYAAN

Berikan tanda centang (√) untuk menunjukkan persepsi atau penilaian bapak/ibu terhadap perbandingan kriteria pelanggan yang dipilih lebih penting dengan melihat perilaku pelanggan dalam melakukan transaksi dengan perusahaan. “Manakah kriteria yang lebih penting

antara keriteria pada kolom sebelah kiri dengan kriteria pada kolom sebelah kanan?"

Kriteria																		Kriteria
Recency	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Frequency*
Frequency	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Monetary*
Recency	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Monetary*
Length	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Frequency*
Monetery	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Length*
length	9	8	7	6	5	4	3	2	1	2	3	4	5	6	7	8	9	*Recency*

Terima kasih atas kesediaan bapak/ibu untuk mengisi kuesioner ini

Halaman ini sengaja dikosongkan

LAMPIRAN B

Tabel B.1 Use Case Melihat visualisasi grafik bar chart, radar chart, dan box plot

UC-2 Melihat visualisasi grafik *bar chart, radar chart,* dan *box plot*		
Purpose	Mempermudah analisis hasil *clustering* secara umum melalui media visual	
Overview	Dimulai ketika pengguna mengklik tombol visualisasi	
Actors	Pengguna aplikasi	
Pre Condition	Pengguna telah berada pada halaman awal aplikasi	
Post Condition	Keempat tipe grafik ditampilkan	
Typical Course Event	***Actor***	***System***
	2. Mengklik tombol visualisasi	Mengambil data hasil *clustering* pada direktori yang telah ditentukan Mengolah data sesuai kebutuhan grafik Menampilkan grafik dalam halaman visualisasi
Alternate Flow Event	-	
Exceptional Flow Event	Aplikasi tertutup karena terdapat *error*	

Tabel B.2 Use Case Melihat visualisasi grafik scatter plot 3D

UC-3 Melihat visualisasi grafik *scatter plot 3D*		
Purpose	Mempermudah analisis hasil *clustering*, terutama dalam nilai variabel melalui media visual	
Overview	Dimulai ketika pengguna mengklik tombol selanjutnya	
Actors	Pengguna aplikasi	
Pre Condition	Pengguna telah berada pada halaman pertama visualisasi	
Post Condition	Keempat grafik *scatter plot 3D* ditampilkan	
Typical Course Event	***Actor***	***System***
	9.1.1.1 Mengklik tombol selanjutnya	9.1.1.2 Mengolah data variabel sesuai kebutuhan grafik 9.1.1.3 Menampilkan grafik 3D dalam halaman visualisasi
Alternate Flow Event	-	
Exceptional Flow Event	Aplikasi tertutup karena terdapat *error*	

Tabel B.3 Use Case Melihat visualisasi data pelanggan berdasarkan lokasi (grafik stacked bar dan donut chart)

UC-4 Melihat visualisasi data pelanggan berdasarkan lokasi (grafik *stacked bar* dan *donut chart*)	
Purpose	Mempermudah analisis hasil *clustering*, terutama dalam manganalisis faktor lokasi pelanggan melalui media visual
Overview	Dimulai ketika pengguna mengklik tombol selanjutnya
Actors	Pengguna aplikasi
Pre Condition	Pengguna telah berada pada halaman kedua visualisasi
Post Condition	Dua tipe grafik ditampilkan (*stacked bar* dan *donut chart*)

Typical Course Event	***Actor***	***System***
	9.1.1.3.1.1.1 Menggklik tombol selanjutnya	9.1.1.3.1.1.2 Mengolah data hasil *clustering* terutama bagian lokasi sesuai kebutuhan grafik 9.1.1.3.1.1.3 Menampilkan grafik *stacked bar* dan *donut chart* dalam halaman visualisasi 3 dimensi
Alternate Flow Event	-	
Exceptional Flow Event	Aplikasi tertutup karena terdapat *error*	

Halaman ini sengaja dikosongkan

LAMPIRAN C

Tabel C.1 Pemodelan LRFM

Nama_outlet	Recency	Monetary	Frequency	Length
88	6	594029.48	4.5	56
371	6	2258207.32	5.5	56
2 PUTRI	2	1140490.14	3	52
27 WANDA, TK	8	922454.98	3.5	49
3F	7	976617.18	3.5	49
9, TK	6	207537.96	4	50
AA JAYA	5	763591.84	4	49
AA MAKMUR	1	1152982.69	4	56
AAN , BU	3	4377323.96	9	56
ABADI	5	425820.13	3.5	49
ABADI JAYA II	3	3606028.95	7.5	56
ABC	2	6997044.46	9	57
ABD HADI	1	6567066.63	15.5	58
ABD KAMAL	6	187919.92	3	49
ABD KARIM , HAJI	3	1652729.95	3.5	49
ABD RAHMAN	7	302563.17	4	49
ABD ROHIM/MU ADI	1	9254232.74	4	55
ABD. JALIL	6	946294.66	3.5	50
ABDUL HALIM	6	253234.74	4	56
ABDUL HANAN	1	3409430.93	5	56
ABDUL MUIS	1	5935940.62	9	50
ABDULAH / POJOK , HAJI	3	1403266.2	3	44
ABDULLAH , HAJI	2	277439.83	4.5	56
ABDUS SAHAM	5	2553388.74	3	42

C-2

Nama_outlet	Recency	Monetary	Frequency	Length
ABI	7	338364.94	2.5	47
ABI QODIR	5	385954.75	4.5	52
ABIL, TK	5	13163645.6	4.5	54
ABNA	9	2609192.77	2.5	42
ADA	7	460100.07	2	42
ADAM BP	6	354090.06	2	50
ADI	2	2580743.56	4.5	56
ADI MULYA	5	359750.01	2.5	47
AF	7	1959015.05	3	49
AFLAH	2	2953982.84	4.5	56
AGIL	2	999154.42	4	42
AGUN BP/HALIL BP	10	253187.79	3.5	49
AGUS	9	11540369.7	5	45
AGUS , BU	1	11559584.5	11	61
AGUS , HAJI	14	1193049.41	2.5	48
AGUSTIN	6	9880405.62	9	56
AHBAB	6	328329.85	1.5	36
AHMAD	3	3809369.57	8.5	59
AHMAD , BU	3	971389.36	8	55
AHMAD , HAJI	1	14264703.9	9.5	56
AHMAD HAJI	1	14039.96	0.5	0
AHMAD, BP	1	1438794.46	5	56
AHMADI	2	1483539.55	5	56
AIFA	3	1078128.03	3.5	23
AIR MANCUR	5	111369.97	0.5	0
AIR MANCUR / POJOK	2	43152374.2	5	56
AIR MANCUR II	3	1477650.26	2	46
AIR MANCUR, TK	29	37909.96	1	28

Nama_outlet	Recency	Monetary	Frequency	Length
AISYAH	1	18069771.2	13.5	57
AISYAH, BU	8	352049.92	3.5	42
AIZAH	3	706753.92	6.5	59
AJAIB	3	1704851.69	3.5	49
AKBAR	6	3279967.31	4	49
AL - AMIN	6	758733.97	3.5	56
AL - HIDAYAH	6	1014366.16	6	51
AL HAMDULILLAH, TK	2	3786911.68	4.5	56
AL HIDAYAH	3	144016.2	3	47
AL HIKMAH	20	602659.85	2.5	35
ALDI	1	1187771.11	5.5	56
ALEX	3	1142990.66	4.5	49
AL-FAN	3	5499832.65	2.5	56
ALI, HAJI	8	4854495.49	3.5	42
ALIF	1	44208.09	0.5	0
ALIF BAROKAH	2	1108548.55	5	57
ALIFIA	6	1099389.58	4.5	49
ALIMUDIN	1	140759.89	2	46
ALVIN	6	1556854.93	5.5	56
AM 89	15	457484.7	2.5	35
AMANAH	3	1952448.23	8.5	53
AMIN , BU	19	489843.36	3	35
AMIN BP	3	31335190.3	5	56
AMIN JAYA	3	27240	1.5	42
AMIN MADINAH	3	3689101.74	4	49
AMINAH	8	196299.95	1.5	28

Halaman ini sengaja dikosongkan

LAMPIRAN D

Tabel D.1 Tabel Normalisasi LRFM

nama_outlet	recency	length	monetary	frequency
88	0.0457	0.0670	0.0020	0.0438
371	0.0457	0.0670	0.0075	0.0547
2 PUTRI	0.0491	0.0622	0.0038	0.0274
27 WANDA, TK	0.0440	0.0586	0.0031	0.0328
3F	0.0448	0.0586	0.0033	0.0328
9, TK	0.0457	0.0598	0.0007	0.0383
AA JAYA	0.0466	0.0586	0.0025	0.0383
AA MAKMUR	0.0500	0.0670	0.0038	0.0383
AAN , BU	0.0483	0.0670	0.0146	0.0930
ABADI	0.0466	0.0586	0.0014	0.0328
ABADI JAYA II	0.0483	0.0670	0.0121	0.0766
ABC	0.0491	0.0682	0.0234	0.0930
ABD HADI	0.0500	0.0694	0.0220	0.1641
ABD KAMAL	0.0457	0.0586	0.0006	0.0274
ABD KARIM , HAJI	0.0483	0.0586	0.0055	0.0328
ABD RAHMAN	0.0448	0.0586	0.0010	0.0383
ABD ROHIM/MU ADI	0.0500	0.0658	0.0310	0.0383
ABD. JALIL	0.0457	0.0598	0.0032	0.0328
ABDUL HALIM	0.0457	0.0670	0.0008	0.0383
ABDUL HANAN	0.0500	0.0670	0.0114	0.0492
ABDUL MUIS	0.0500	0.0598	0.0199	0.0930
ABDULAH / POJOK , HAJI	0.0483	0.0527	0.0047	0.0274
ABDULLAH , HAJI	0.0491	0.0670	0.0009	0.0438
ABDUS SAHAM	0.0466	0.0503	0.0085	0.0274

D-2

nama_outlet	recency	length	monetary	frequency
ABI	0.0448	0.0562	0.0011	0.0219
ABI QODIR	0.0466	0.0622	0.0013	0.0438
ABIL, TK	0.0466	0.0646	0.0441	0.0438
ABNA	0.0431	0.0503	0.0087	0.0219
ADA	0.0448	0.0503	0.0015	0.0164
ADAM BP	0.0457	0.0598	0.0012	0.0164
ADI	0.0491	0.0670	0.0086	0.0438
ADI MULYA	0.0466	0.0562	0.0012	0.0219
AF	0.0448	0.0586	0.0065	0.0274
AFLAH	0.0491	0.0670	0.0099	0.0438
AGIL	0.0491	0.0503	0.0033	0.0383
AGUN BP/HALIL BP	0.0422	0.0586	0.0008	0.0328
AGUS	0.0431	0.0539	0.0386	0.0492
AGUS , BU	0.0500	0.0730	0.0387	0.1149
AGUS , HAJI	0.0388	0.0574	0.0040	0.0219
AGUSTIN	0.0457	0.0670	0.0331	0.0930
AHBAB	0.0457	0.0431	0.0011	0.0109
AHMAD	0.0483	0.0706	0.0127	0.0875
AHMAD , BU	0.0483	0.0658	0.0032	0.0821
AHMAD , HAJI	0.0500	0.0670	0.0478	0.0985
AHMAD HAJI	0.0500	0.0000	0.0000	0.0000
AHMAD, BP	0.0500	0.0670	0.0048	0.0492
AHMADI	0.0491	0.0670	0.0050	0.0492
AIFA	0.0483	0.0275	0.0036	0.0328
AIR MANCUR	0.0466	0.0000	0.0004	0.0000
AIR MANCUR / POJOK	0.0491	0.0670	0.1445	0.0492
AIR MANCUR II	0.0483	0.0550	0.0049	0.0164
AIR MANCUR, TK	0.0259	0.0335	0.0001	0.0055

nama_outlet	recency	length	monetary	frequency
AISYAH	0.0500	0.0682	0.0605	0.1422
AISYAH, BU	0.0440	0.0503	0.0012	0.0328
AIZAH	0.0483	0.0706	0.0024	0.0656
AJAIB	0.0483	0.0586	0.0057	0.0328
AKBAR	0.0457	0.0586	0.0110	0.0383
AL - AMIN	0.0457	0.0670	0.0025	0.0328
AL - HIDAYAH	0.0457	0.0610	0.0034	0.0602
AL HAMDULILLAH, TK	0.0491	0.0670	0.0127	0.0438
AL HIDAYAH	0.0483	0.0562	0.0005	0.0274
AL HIKMAH	0.0336	0.0419	0.0020	0.0219
ALDI	0.0500	0.0670	0.0040	0.0547
ALEX	0.0483	0.0586	0.0038	0.0438
AL-FAN	0.0483	0.0670	0.0184	0.0219
ALI, HAJI	0.0440	0.0503	0.0162	0.0328
ALIF	0.0500	0.0000	0.0001	0.0000
ALIF BAROKAH	0.0491	0.0682	0.0037	0.0492
ALIFIA	0.0457	0.0586	0.0037	0.0438
ALIMUDIN	0.0500	0.0550	0.0005	0.0164
ALVIN	0.0457	0.0670	0.0052	0.0547
AM 89	0.0379	0.0419	0.0015	0.0219
AMANAH	0.0483	0.0634	0.0065	0.0875
AMIN , BU	0.0345	0.0419	0.0016	0.0274
AMIN BP	0.0483	0.0670	0.1049	0.0492
AMIN JAYA	0.0483	0.0503	0.0001	0.0109
AMIN MADINAH	0.0483	0.0586	0.0123	0.0383
AMINAH	0.0440	0.0335	0.0006	0.0109

Halaman ini sengaja dikosongkan

LAMPIRAN E

Tabel E.1 Tabel Normalisasi

Nama_outlet	Recency	Monetary	Frequency	Length	CLV
88	6	594029.48	4.5	56	0.1584451
371	6	2258207.32	5.5	56	0.1749609
2 PUTRI	2	1140490.14	3	52	0.1425252
27 WANDA, TK	8	922454.98	3.5	49	0.1385029
3F	7	976617.18	3.5	49	0.1395464
9, TK	6	207537.96	4	50	0.1444996
AA JAYA	5	763591.84	4	49	0.1460275
AA MAKMUR	1	1152982.69	4	56	0.1591572
AAN , BU	3	4377323.96	9	56	0.2229397
ABADI	5	425820.13	3.5	49	0.1394255
ABADI JAYA II	3	3606028.95	7.5	56	0.2039443
ABC	2	6997044.46	9	57	0.2337738
ABD HADI	1	6567066.63	15.5	58	0.30551
ABD KAMAL	6	187919.92	3	49	0.1322959
ABD KARIM , HAJI	3	1652729.95	3.5	49	0.1452595
ABD RAHMAN	7	302563.17	4	49	0.1427591
ABD ROHIM/MU ADI	1	9254232.74	4	55	0.1850976
ABD. JALIL	6	946294.66	3.5	50	0.1415036
ABDUL HALIM	6	253234.74	4	56	0.151833
ABDUL HANAN	1	3409430.93	5	56	0.1776569
ABDUL MUIS	1	5935940.62	9	50	0.2227044
ABDULAH / POJOK , HAJI	3	1403266.2	3	44	0.1329696
ABDULLAH , HAJI	2	277439.83	4.5	56	0.1608329

E-2

Nama_outlet	Recency	Monetary	Frequency	Length	CLV
ABDUS SAHAM	5	2553388.74	3	42	0.1327047
ABI	7	338364.94	2.5	47	0.1240738
ABI QODIR	5	385954.75	4.5	52	0.1538233
ABIL, TK	5	13163645.6	4.5	54	0.1990187
ABNA	9	2609192.77	2.5	42	0.1239727
ADA	7	460100.07	2	42	0.1130274
ADAM BP	6	354090.06	2	50	0.1231081
ADI	2	2580743.56	4.5	56	0.1685484
ADI MULYA	5	359750.01	2.5	47	0.1258696
AF	7	1959015.05	3	49	0.1373666
AFLAH	2	2953982.84	4.5	56	0.1697986
AGIL	2	999154.42	4	42	0.1410258
AGUN BP/HALIL BP	10	253187.79	3.5	49	0.1345369
AGUS	9	11540369.7	5	45	0.1848329
AGUS , BU	1	11559584.5	11	61	0.2765884
AGUS , HAJI	14	1193049.41	2.5	48	0.122099
AGUSTIN	6	9880405.62	9	56	0.2387873
AHBAB	6	328329.85	1.5	36	0.1007971
AHMAD	3	3809369.57	8.5	59	0.2191567
AHMAD , BU	3	971389.36	8	55	0.1993928
AHMAD , HAJI	1	14264703.9	9.5	56	0.2632545
AHMAD HAJI	1	14039.96	0.5	0	0.0500316
AHMAD, BP	1	1438794.46	5	56	0.1710558
AHMADI	2	1483539.55	5	56	0.1703436
AIFA	3	1078128.03	3.5	23	0.11222
AIR MANCUR	5	111369.97	0.5	0	0.0469093
AIR MANCUR / POJOK	2	43152374.2	5	56	0.3099233

Nama_outlet	Recency	Monetary	Frequency	Length	CLV
AIR MANCUR II	3	1477650.26	2	46	0.1246711
AIR MANCUR, TK	29	37909.96	1	28	0.0649524
AISYAH	1	18069771.2	13.5	57	0.3209619
AISYAH, BU	8	352049.92	3.5	42	0.1282151
AIZAH	3	706753.92	6.5	59	0.1868814
AJAIB	3	1704851.69	3.5	49	0.1454341
AKBAR	6	3279967.31	4	49	0.1535947
AL - AMIN	6	758733.97	3.5	56	0.1480557
AL - HIDAYAH	6	1014366.16	6	51	0.1702813
AL HAMDULILLAH, TK	2	3786911.68	4.5	56	0.1725887
AL HIDAYAH	3	144016.2	3	47	0.1323416
AL HIKMAH	20	602659.85	2.5	35	0.0993916
ALDI	1	1187771.11	5.5	56	0.1756855
ALEX	3	1142990.66	4.5	49	0.1544932
AL-FAN	3	5499832.65	2.5	56	0.1555821
ALI, HAJI	8	4854495.49	3.5	42	0.1432972
ALIF	1	44208.09	0.5	0	0.0501326
ALIF BAROKAH	2	1108548.55	5	57	0.1702842
ALIFIA	6	1099389.58	4.5	49	0.1517609
ALIMUDIN	1	140759.89	2	46	0.121917
ALVIN	6	1556854.93	5.5	56	0.1726115
AM 89	15	457484.7	2.5	35	0.1032156
AMANAH	3	1952448.23	8.5	53	0.2057562
AMIN , BU	19	489843.36	3	35	0.1053463
AMIN BP	3	31335190.3	5	56	0.2694767
AMIN JAYA	3	27240	1.5	42	0.1095551

Halaman ini sengaja dikosongkan

LAMPIRAN F

script F.0.1 Script kode Keseluruhan

```
1.  from flask import Flask, render_template,
    request, flash, redirect, url_for, send_file,
    Markup
2.  import numpy as np
3.  import pandas as pd
4.  import matplotlib.pyplot as plt;
    plt.rcdefaults()
5.  from io import BytesIO
6.  import base64
7.  from werkzeug.utils import secure_filename
8.  import os
9.  import datetime as dt
10. import json
11.
12. UPLOAD_FOLDER = '..\coding template
    flask\\tempatUpload'
13. ALLOWED_EXTENSIONS = set(['xls', 'xlsx'])
14.
15. app = Flask(__name__)
16. app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
17. data=pd.DataFrame()
18. jumlahK=0
19.
20. def allowed_file(filename):
21.     return '.' in filename and \
22.            filename.rsplit('.', 1)[1].lower() in
    ALLOWED_EXTENSIONS
23.
24. @app.route('/upload', methods=['GET', 'POST'])
25. def upload_file():
26.     if request.method == 'POST':
27.         # check if the post request has the file
    part
28.         if 'file' not in request.files:
29.             flash('No file part')
30.             return redirect(request.url)
31.         file = request.files['file']
```

```
32.
33.         # if user does not select file, browser
    also
34.         # submit an empty part without filename
35.         if file.filename == '':
36.             flash('No selected file')
37.             return redirect(request.url)
38.         if file and allowed_file(file.filename):
39.             filename =
    secure_filename(file.filename)
40.
    file.save(os.path.join(app.config['UPLOAD_FOLDER
    '], filename))
41.             global data
42.
    data=pd.read_excel(app.config['UPLOAD_FOLDER']+'
    \\'+filename)
43.
44.             return LRFM_awal(data)
45.
46.     return '''
47.     <!doctype html>
48.     <style type="text/css">
49.     body {
50.     background:rgba(0,191,255,0.8);
51.     background-size: 100% 100%;
52.     background-repeat:no-repeat;
53.     background-position:top;
54.     background-attachment:scroll;
55.     }
56.     h1{
57.     font-family: courier new;
58.     color: white;
59.     }
60.     </style>
61.     <title>Upload new File</title>
62.     <h1>Upload new File</h1>
63.     <form method=post enctype=multipart/form-
    data>
64.       <input type=file name=file>
65.       <input type=submit value=Upload>
```

```
    </form>
    '''
```

```
@app.route('/analisa', methods=['GET', 'POST'])
def analisa(df=None):

    if request.method == 'POST' and 'jumlahK' in
    request.form:
        global jumlahK
        jumlahK = request.form['jumlahK']
        jumlahK = int(jumlahK)
        return cluster(jumlahK)

    nilai_error=np.empty(len(error_elbow)-2)

    for ab in np.arange(len(error_elbow)-2):
        nilai_error[ab]=(df[ab]-
    df[ab+1])/(df[ab+1]-df[ab+2])

    name2=np.argmax(nilai_error)+2
    plt.plot([2,3,4,5,6,7,8,9,10],df)

    figfile = BytesIO()
    plt.savefig(figfile, format='png')
    figfile.seek(0)
    figdata_png =
    base64.b64encode(figfile.getvalue())

    return render_template('index.html',
    jumlahK=jumlahK, name2=name2, title='Hasil
    Olahan Data', result=figdata_png.decode('utf8'))
```

```
import glob
hasil=pd.DataFrame()

@app.route('/', methods=['GET', 'POST'])
def index():
```

```
    list_of_files = glob.glob('..\coding
template flask\\data_hasil\\*.xlsx') # * means
all if need specific format then *.csv
    latest_file = max(list_of_files,
key=os.path.getctime)
    global hasil
    hasil=pd.read_excel(latest_file)
            tabel=hasil.to_html()
            return render_template('awal1.html',
tabel=tabel)
```

```
       panjang=1162
       #Hitung LRFM

       LRFM=pd.DataFrame(columns=[['nama_outlet'
,'recency','length','monetary','frequency']])
       norm_lrfm=pd.DataFrame(index=range(panjan
g),columns=[['nama_outlet','recency','length','m
onetary','frequency','clv','kecamatan']])

       def LRFM_awal(df):
           n=0
           nama_outlet=pd.Series(df['Outlet
Name'].unique())
           end_date = df['INVDate'].max() +
dt.timedelta(days=1)

           global LRFM
           for a in nama_outlet:
               ab=df[df['Outlet Name']==a]

jml_inv=len(ab['INVNumber'].unique())
               ab['recent'] =
pd.Series(end_date-ab['INVDate'])
               global LRFM


LRFM.loc[n,'recency']=ab['recent'].min()
               LRFM.loc[n,'nama_outlet']=a
```

```
122.
    LRFM.loc[n,'length']=ab['recent'].max()-
    ab['recent'].min()
123.
    LRFM.loc[n,'monetary']=ab['harga_net'].sum()
124.                LRFM.loc[n,'frequency']=jml_inv/2
125.
126.                n+=1
127.
128.            #normalisasi
129.            global panjang
130.            panjang=len(LRFM)
131.            global norm_lrfm
132.
133.            for b in np.arange(panjang):
134.
135.                #untuk R = 1-R, karena semakin
    kecil semakin baik
136.                norm_lrfm.loc[b,'recency']=1-
    (LRFM.loc[b,'recency']-
    LRFM['recency'].min())/(LRFM['recency'].max()-
    LRFM['recency'].min())
137.
    norm_lrfm.loc[b,'nama_outlet']=LRFM.loc[b,'nama_
    outlet']
138.
    norm_lrfm.loc[b,'length']=(LRFM.loc[b,'length']-
    LRFM['length'].min())/(LRFM['length'].max()-
    LRFM['length'].min())
139.
    norm_lrfm.loc[b,'monetary']=(LRFM.loc[b,'monetar
    y']-
    LRFM['monetary'].min())/(LRFM['monetary'].max()-
    LRFM['monetary'].min())
140.
    norm_lrfm.loc[b,'frequency']=(LRFM.loc[b,'freque
    ncy']-
    LRFM['frequency'].min())/(LRFM['frequency'].max(
    )-LRFM['frequency'].min())
141.
```

```
142.
    norm_lrfm['kecamatan']=pd.Series(kecm(df,
    nama_outlet))
143.
144.            return elbow()
```

```
145.        #mengisi kecamatan
146.        tempat=pd.read_excel('..\coding template
    flask\\data_kecamatan\\lokasi.xlsx')
147.
148.        def kecm(df, nama_outl):
149.            n=0
150.            o=0
151.            kcmtn=[len(nama_outl)]
152.            for ab in nama_outl:
153.                bc=(df.loc[:,'Outlet Name']==ab)
154.
    de=pd.Series(df.loc[bc,'OutletAddress'])
155.                ef=(tempat['alamat']==de[o])
156.
    kcmtn.append(tempat.loc[ef,'kecamatan'])
157.                o+=len(de)
158.                n+=1
159.            return(kcmtn[1:])
```

```
160.        #Elbow
161.        from sklearn.cluster import KMeans
162.
163.        error=np.empty([panjang,4])
164.        label=np.empty(panjang)
165.        ttl_error=np.empty(panjang)
166.        error_elbow=np.empty([9]) #nyoba 10
167.        centroid=np.empty([10,4])
168.
169.        bobotR = 0.05
170.        bobotF = 0.279
171.        bobotM = 0.598
172.        bobotL = 0.073
173.
174.        def elbow():
```

```
175.
    norm_lrfm['recency']=norm_lrfm['recency']*bobotR
176.
    norm_lrfm['frequency']=norm_lrfm['frequency']*bo
    botF
177.
    norm_lrfm['monetary']=norm_lrfm['monetary']*bobo
    tM
178.
    norm_lrfm['length']=norm_lrfm['length']*bobotL
179.
180.            global error_elbow
181.            for tiap in np.arange(9):
182.
    error_elbow[tiap]=hitung_error(norm_lrfm,tiap+2)
183.
184.            return analisa(error_elbow)
185.
186.        def hitung_error(df, k):
187.            #kmeans clustering
188.            kmeans =
    KMeans(n_clusters=k).fit(norm_lrfm.drop(['nama_o
    utlet','clv','kecamatan'],axis=1, level=0))
189.
190.            #mengambil centroid tiap cluster
191.            global centroid
192.            centroid=kmeans.cluster_centers_
193.            global label
194.            label=kmeans.labels_
195.
196.            #∑_(k=1)^K▒∑_(Xi∈Ck)▒〖||X_i-μ_k ||〗
    ^2
197.            for ab in
    np.arange(len(norm_lrfm['recency'])):
198.
199.                error[ab][0] =
    np.square(np.absolute(df.loc[ab,'recency']-
    centroid[label[ab]][0]))
```

```
                error[ab][1] =
    np.square(np.absolute(df.loc[ab,'length']-
    centroid[label[ab]][1]))
                error[ab][2] =
    np.square(np.absolute(df.loc[ab,'monetary']-
    centroid[label[ab]][2]))
                error[ab][3] =
    np.square(np.absolute(df.loc[ab,'frequency']-
    centroid[label[ab]][3]))

                global ttl_error
                ttl_error[ab] =
    error[ab][0]+error[ab][1]+error[ab][2]+error[ab]
    [3]

            return np.sum(ttl_error)
```

```
        hasil_akhir=pd.DataFrame(index=range(panj
    ang),columns=['nama_outlet','recency','monetary'
    ,'frequency','length','clv','cluster','kecamatan
    '])

        @app.route('/kmeans', methods=['GET',
    'POST'])
        def cluster(jumlah):

    norm_lrfm['clv']=pd.Series(clv(norm_lrfm),
    index=norm_lrfm.index)

            global hasil_akhir

            kmeans =
    KMeans(n_clusters=jumlah).fit(norm_lrfm.drop(['n
    ama_outlet','clv','kecamatan'],axis=1, level=0))
            hasil_cluster=kmeans.labels_


    hasil_akhir['nama_outlet']=norm_lrfm['nama_outle
    t']

```

```
221.
    cb=LRFM['recency'].values.astype('timedelta64[D]
    ')
222.
    hasil_akhir['recency']=cb.astype(float)
223.
224.
    hasil_akhir['monetary']=LRFM['monetary']
225.
    hasil_akhir['frequency']=LRFM['frequency']
226.
227.
    cv=LRFM['length'].values.astype('timedelta64[D]'
    )
228.
    hasil_akhir['length']=cv.astype(float)
229.
230.            hasil_akhir['clv']=norm_lrfm['clv']
231.
    hasil_akhir['cluster']=pd.Series(hasil_cluster,
    index=hasil_akhir.index)
232.
233.
    hasil_akhir['kecamatan']=norm_lrfm['kecamatan']
234.            zlst =
    list(zip(*hasil_akhir['kecamatan']))
235.
    hasil_akhir['kecamatan']=pd.Series(zlst[0])
236.
237.            return
    render_template('hasil_clust.html',jumlah=jumlah
    )
```

```
238.        #Hitung Customer Lifetime Value
239.        nilai_clv=np.empty(panjang)
240.
241.        def clv(df):
242.            for ab in np.arange(panjang):
243.
    nilai_clv[ab]=float(df.loc[ab,'recency'])+float(
```

```
    df.loc[ab,'frequency'])+float(df.loc[ab,'monetar
    y'])+float(df.loc[ab,'length'])
244.            return nilai_clv
```

```
245.        #export file
246.        @app.route('/download', methods=['GET',
    'POST'])
247.        def export():
248.            output = BytesIO()
249.            writer = pd.ExcelWriter(output,
    engine='xlsxwriter')
250.
251.            hasil_akhir.to_excel(writer, startrow
    = 0, merge_cells = False, sheet_name =
    "Sheet_1")
252.            workbook = writer.book
253.            worksheet = writer.sheets["Sheet_1"]
254.
255.            #the writer has done its job
256.            writer.close()
257.
258.            #go back to the beginning of the
    stream
259.            output.seek(0)
260.
261.            #finally return the file
262.            return send_file(output,
    attachment_filename="hasil.xlsx",
    as_attachment=True)
```

```
263.        #Visualisasi Dashboard
264.        import pygal
265.        from pygal.style import DefaultStyle
266.
267.        @app.route('/dashboard', methods=['GET',
    'POST'])
268.        def visual(k=None):
269.            k=len(hasil['cluster'].unique())
270.
271.            d=np.empty([k, 5])
272.            e=np.empty([k, 4])
```

```
273.
274.           for a in range(k):
275.               b=(hasil['cluster']==a)
276.               d[a, 0]=len(hasil_akhir[b])
277.               d[a,
   1]=hasil.loc[b,'length'].sum()
278.               d[a,
   2]=hasil.loc[b,'recency'].sum()
279.               d[a,
   3]=hasil.loc[b,'frequency'].sum()
280.               d[a,
   4]=hasil.loc[b,'monetary'].sum()
281.
282.               e[a,
   0]=hasil.loc[b,'length'].mean()
283.               e[a,
   1]=hasil.loc[b,'recency'].mean()
284.               e[a,
   2]=hasil.loc[b,'frequency'].mean()
285.               e[a,
   3]=hasil.loc[b,'monetary'].mean()/1000000
286.
287.           try:
288.               #graph.range = [0,
   np.amax(np.array(hasil_akhir['length']))]
289.
290.               bar=pygal.Bar()
291.               bar.title = 'jumlah anggota
   cluster'
292.               for ab in range(k):
293.
   bar.add('cluster'+str(ab+1),d[ab, 0])
294.               bar_data=bar.render_data_uri()
295.
296.               #Bar chart untuk total nilai tiap
   variabel (nilai awal)
297.               bar_chart =
   pygal.Bar(x_label_rotation=30,
   style=custom_style)
```

```
298.                bar_chart.title = 'Perbandingan
    Total Nilai Variabel'
299.                bar_chart.x_labels = ['Length',
    'Recency', 'Frequency','monetary(juta)']
300.                for ab in range(k):
301.
    bar_chart.add('cluster'+str(ab+1), d[ab, 1:5])
302.
    dot_data=bar_chart.render_data_uri()
303.
304.                #box plot (persebaran nilai tiap
    variabel)
305.                box_plot = pygal.Box()
306.                box_plot.title = 'persebaran
    nilai variabel Length'
307.                for ab in range(k):
308.                    b=hasil[hasil['cluster']==ab]
309.
    box_plot.add('cluster'+str(ab+1),
    np.array(b['length']))
310.
    box_data=box_plot.render_data_uri()
311.
312.                box2_plot = pygal.Box()
313.                box2_plot.title = 'persebaran
    nilai variabel Recency'
314.                for ab in range(k):
315.                    b=hasil[hasil['cluster']==ab]
316.
    box2_plot.add('cluster'+str(ab+1),
    np.array(b['recency']))
317.
    box2_data=box2_plot.render_data_uri()
318.
319.                box3_plot = pygal.Box()
320.                box3_plot.title = 'persebaran
    nilai variabel Frequency'
321.                for ab in range(k):
322.                    b=hasil[hasil['cluster']==ab]
```

```
box3_plot.add('cluster'+str(ab+1), np.array(b['frequency']))
box3_data=box3_plot.render_data_uri()
                box4_plot = pygal.Box()
                box4_plot.title = 'persebaran nilai variabel Monetary'
                for ab in range(k):
                    b=hasil[hasil['cluster']==ab]
box4_plot.add('cluster'+str(ab+1), np.array(b['monetary']))
box4_data=box4_plot.render_data_uri()
                #radar (melihat karakteristik cluster (average nilai))
                radar = pygal.Radar(dynamic_print_values=True, style=DefaultStyle(
value_font_family='googlefont:Raleway',
value_font_size=30,
value_colors=('white',)))
                radar.title = 'Rata-rata nilai variabel dari tiap cluster'
                radar.x_labels = ['Length', 'Recency', 'Frequency', 'Monetary (juta)']
                for ab in range(k):
radar.add('cluster'+str(ab+1), e[ab, 0:4])
radar_data=radar.render_data_uri()
                return render_template('graph.html', box_data=box_data,
```

```
    box2_data=box2_data,
    box3_data=box3_data,box4_data=box4_data,
    bar_data=bar_data, dot_data=dot_data,
    mon_data=mon_data, radar_data=radar_data)
345.
346.           except ValueError:
347.               print("Oops!  That was no valid
    number.")
```

```
348.       from plotly.offline import
    download_plotlyjs, init_notebook_mode, plot,
    iplot
349.       import plotly.graph_objs as go
350.
351.       init_notebook_mode(connected=True)
352.
353.       @app.route('/dashboard2', methods=['GET',
    'POST'])
354.       def visual2(k=None):
355.           dfs=hasil['clv']*60000
356.           trace1 = go.Scatter3d(
357.               x=hasil['length'][:],
358.               y=hasil['recency'][:],
359.               z=hasil['frequency'][:],
360.               text=hasil['nama_outlet'][:],
361.               mode='markers',
362.               marker=dict(
363.                   sizemode='diameter',
364.                   sizeref=panjang,
365.                   size=dfs[:],
366.                   color = hasil['cluster'][:],
367.                   colorscale = 'Viridis',
368.                   colorbar = dict(title =
    'Cluster'),
369.                   line=dict(color='rgb(140,
    140, 170)'))
370.           )
371.
372.           trace2 = go.Scatter3d(
373.               x=hasil['length'][:],
374.               y=hasil['recency'][:],
```

```
                z=hasil['monetary'][:],
                text=hasil['nama_outlet'][:],
                mode='markers',
                marker=dict(
                    sizemode='diameter',
                    sizeref=panjang,
                    size=dfs[:],
                    color = hasil['cluster'][:],
                    colorscale = 'Viridis',
                    colorbar = dict(title =
    'Cluster'),
                    line=dict(color='rgb(140,
    140, 170)')
                )
            )

            trace3 = go.Scatter3d(
                x=hasil['length'][:],
                y=hasil['frequency'][:],
                z=hasil['monetary'][:],
                text=hasil['nama_outlet'][:],
                mode='markers',
                marker=dict(
                    sizemode='diameter',
                    sizeref=panjang,
                    size=dfs[:],
                    color = hasil['cluster'][:],
                    colorscale = 'Viridis',
                    colorbar = dict(title =
    'Cluster'),
                    line=dict(color='rgb(140,
    140, 170)')
                )
            )

            trace4 = go.Scatter3d(
                x=hasil['recency'][:],
                y=hasil['frequency'][:],
                z=hasil['monetary'][:],
                text=hasil['nama_outlet'][:],
```

```
                mode='markers',
                marker=dict(
                    sizemode='diameter',
                    sizeref=panjang,
                    size=dfs[:],
                    color = hasil['cluster'][:],
                    colorscale = 'Viridis',
                    colorbar = dict(title =
    'Cluster'),
                    line=dict(color='rgb(140,
    140, 170)')
                )
            )

            graf=plot([trace1], show_link=False,
    output_type='div')
            graf2=plot([trace2], show_link=False,
    output_type='div')
            graf3=plot([trace3], show_link=False,
    output_type='div')
            graf4=plot([trace4], show_link=False,
    output_type='div')

            return render_template('graph2.html',
    grafik=Markup(graf), grafik2=Markup(graf2),
    grafik3=Markup(graf3), grafik4=Markup(graf4))
```

```
        @app.route('/dashboard3', methods=['GET',
    'POST'])
        def visual3():
            k=len(hasil['cluster'].unique())

    kcmtn=pd.Series(hasil['kecamatan'].unique())
            lokasi=np.empty([k, len(kcmtn)])


            for i in range(k):
                n=0
                jml=hasil[hasil['cluster']==i]

                for daerah in kcmtn:
```

```
441.
    ab=(jml.loc[:,'kecamatan']==daerah)
442.
    lokasi[i,n]=len(jml.loc[ab,'kecamatan'])
443.                    n+=1
444.
445.            #stacked Bar
446.            stacked = pygal.StackedBar()
447.            stacked.title = 'Jumlah anggota
    cluster di setiap kecamatan'
448.            stacked.x_labels = kcmtn
449.            for i in range(k):
450.                stacked.add('cluster'+str(i+1),
    lokasi[i,:])
451.
    stacked_data=stacked.render_data_uri()
452.
453.            #donut chart
454.            fig = {
455.              "data": [
456.                {
457.                  "values": lokasi[0,:],
458.                  "labels": kcmtn[:],
459.                  "domain": {"x": [0, .48]},
460.                  "name": "Cluster1",
461.
    "hoverinfo":"label+percent+name",
462.                  "hole": .4,
463.                  "type": "pie"
464.                },
465.                {
466.                  "values": lokasi[1,:],
467.                  "labels": kcmtn[:],
468.                  "text":"Cluster2",
469.                  "textposition":"inside",
470.                  "domain": {"x": [.52, 1]},
471.                  "name": "Cluster2",
472.
    "hoverinfo":"label+percent+name",
473.                  "hole": .4,
```

```
                "type": "pie"
            }],
        "layout": {
                "title":"persebaran anggota cluster",
                "annotations": [
                    {
                        "font": {
                            "size": 20
                        },
                    "showarrow": False,
                    "text": "Cluster1",
                    "x": 0.20,
                    "y": 0.5
                    },
                    {
                    "font": {
                        "size": 20
                    },
                    "showarrow": False,
                    "text": "Cluster2",
                    "x": 0.8,
                    "y": 0.5
                    }
                ]
            }
        }
        donut=plot(fig, show_link=False, output_type='div')

        return render_template('graph3.html', stacked_data=stacked_data, donut=Markup(donut))
```

```
    from flask_session import Session

    sess = Session()
    app.secret_key = 'the mega super secret key'

    if __name__ == '__main__':
```

```
509.            app.config['SESSION_TYPE'] =
    'filesystem'
510.
511.            sess.init_app(app)
512.
513.            app.run()
```