**SKRIPSI**

# ANALISIS KLASTER HIERARKI UNTUK DATA KATEGORIK DENGAN ALGORITMA *DIVISIVE HIEARCHICAL CLUSTERING OF CATEGORICAL DATA* (DHCC) (STUDI KASUS: PENGELOMPOKKAN ANGGOTA UKM MAPAGAMA)

***HIERARCHICAL CLUSTERING ANALYSIS FOR CATEGORICAL DATA USING  DIVISIVE HIERARCHICAL CLUSTERING OF CATEGORICAL DATA (DHCC) ALGORITHM (CASE STUDY: GROUPING MEMBERS OF UKM MAPAGAMA)***


**Oleh:**

**Muhammad Burhan Aziz**

**17/412744/PA/18063**

**PROGRAM STUDI STATISTIKA**

**JURUSAN MATEMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS GADJAH MADA**



**YOGYAKARTA**

**2021**

**SKRIPSI**

# ANALISIS KLASTER HIERARKI UNTUK DATA KATEGORIK DENGAN ALGORITMA *DIVISIVE HIERARCHICAL CLUSTERING OF CATEGORICAL DATA* (DHCC) (STUDI KASUS: PENGELOMPOKKAN ANGGOTA UKM MAPAGAMA)

***HIERARCHICAL CLUSTERING ANALYSIS FOR CATEGORICAL DATA USING DIVISIVE HIERARCHICAL CLUSTERING OF CATEGORICAL DATA (DHCC) ALGORITHM (CASE STUDY: GROUPING MEMBERS OF UKM MAPAGAMA)***

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Sarjana Program Studi Statistika Departemen Matematika

**Oleh:**

**Muhammad Burhan Aziz**

**17/412744/PA/18063**

**PROGRAM STUDI STATISTIKA**

**JURUSAN MATEMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS GADJAH MADA**

**YOGYAKARTA**

**2021**

**HALAMAN PENGESAHAN**

**SKRIPSI**

**ANALISIS KLASTER HIERARKI UNTUK DATA KATEGORIK DENGAN ALGORITMA *DIVISIVE HIERARCHICAL CLUSTERING OF CATEGORICAL DATA* (DHCC) (STUDI KASUS: PENGELOMPOKKAN ANGGOTA UKM MAPAGAMA)**

Telah dipersiapkan dan disusun oleh:

Muhammad Burhan Aziz

17/412744/PA/18063

Telah dipertahankan di depan Tim Penguji

Pada tanggal 11 Juni 2021

Susunan Tim Penguji

Dr. Abdurakhman, M.Si.
Pembimbing

Dr. Adhitya Ronnie Effendie, M.Sc.
Ketua Tim Penguji

Dr. Herni Utami, M.Si.
Penguji

Vemmie Nastiti Lestari, S.Si., M.Sc.
Penguji

## PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan di bawah ini:

| | |
|---|---|
| Nama | : Muhammad Burhan Aziz |
| NIM | : 17/412744/PA/18063 |
| Tahun terdaftar | : 2017 |
| Program Studi | : Statistika |
| Fakultas | : Matematika dan Ilmu Pengetahuan Alam |

Menyatakan bahwa dalam dokumen ilmiah skripsi ini tidak terdapat bagian dari karya ilmiah lain yang telah diajukan untuk memperoleh gelar akademik di suatu lembaga Pendidikan Tinggi, dan juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang/lembaga lain, kecuali yang secara tertulis disitasi dalam dokumen ini dan disebutkan sumbernya secara lengkap dalam daftar pustaka.

Dengan demikian saya menyatakan bahwa dokumen ilmiah ini bebas dari unsur-unsur plagiasi dan apabila dokumen ilmiah skripsi ini di kemudian hari terbukti merupakan plagiasi dari hasil karya penulis dan/atau dengan sengaja merupakan karya atau pendapat yang merupakan hasil karya penulis lain, maka penulis bersedia menerima sanksi akademik dan/atau saksi hukum yang berlaku.

Yogyakarta, 10 Juli 2021


Muhammad Burhan Aziz

Untuk kedua orang tua,
saudara, atas semua
dukungan moral dan
material.

Tak lupa juga untuk
orang-orang di sekitar
yang memicu saya untuk
menjadi manusia yang
lebih baik dan berguna.

# PRAKATA

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa atas limpahan rahmat-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Analisis Klaster Hierarki untuk Data Kategorik dengan Algoritma *Divisive Hierarchical Clustering of Categorical Data* (DHCC) (Studi Kasus: Pengelompokkan Anggota UKM Mapagama)" sebagai sarana untuk memperoleh gelar S1 pada Program Studi Statistika Jurusan Matematika FMIPA Universitas Gadjah Mada.

Pada kesempatan ini penulis juga mengucapkan terima kasih kepada pihak yang telah membantu dalam penulisan skripsi ini, baik secara langsung maupun tidak langsung. Ucapan terima kasih dari penulis disampaikan kepada :

1. Bapak Prof. Dr. Triyono, S.U., selaku Dekan FMIPA Universitas Gadjah Mada.
2. Bapak Dr. Fajar Adi Kusomo, S.Si., M.Si., selaku Ketua Departemen Matematika FMIPA Universitas Gadjah Mada.
3. Ibu Dr. Herni Utami, S.Si., M.Si., selaku Ketua Program Studi Statistika Jurusan Matematika FMIPA Universitas Gadjah Mada.
4. Bapak Dr. Abdurakhman, S.Si., M.Si., selaku Dosen Pembimbing yang senantiasa memberikan motivasi dan juga masukan kepada penulis dalam menyelesaikan tugas akhir.
5. Ibu Dr. Herni Utami, S.Si., M.Si., selaku Dosen Pembimbing Akademik yang telah memberikan saran, bimbingan, dan ilmu kepada penulis.
6. Dosen Program Studi Statistika UGM yang telah memberikan ilmu selama perkuliahan.
7. Bapak Idrap Selaku Staf Akademik Program Studi Statistika FMIPA UGM yang telah memberikan bantuan untuk kelancaran tugas akhir.
8. Bapak Gunawan dan Ibu Yuni selaku orang tua yang selalu memberikan doa dan dukungan.

9. UKM Mapagama yang telah memberikan wadah untuk mengembangkan pribadi dan juga memberikan inspirasi untuk tugas akhir.
10. Indo dan Devan yang mengisi hari-hari penulis di kos Wakidi dan tak lupa juga penghuni lainnya Gufron, Angkie, dan Jorkaef.
11. Gita, Areta, dan Enggal yang memberikan dukungan kepada penulis dalam memulai pengerjaan tugas akhir.
12. Orang-orang terdekat penulis yang selalu memberikan dukungan dalam bentuk apapun.
13. Teman-teman Program Studi Statistika UGM 2017 yang telah menjadi keluarga selama masa perkuliahan.
14. Penghuni kontrakan Soropadan beriman yang menemani penulis selama di Yogyakarta.
15. Semua pihak yang telah membantu penulis dalam proses penyelesaian tugas akhir ini yang tidak dapat disebutkan satu per satu.

Penulis menyadari bahwa penulisan tugas akhir ini masih banyak terdapat kekurangan mengingat keterbatasan penulis sebagai mahasiswa yang masih harus banyak belajar. Oleh karena itu, penulis memohon maaf, kritik, dan saran yang berkaitan dengan tugas akhir ini. Semoga tugas akhir ini bermanfaat bagi banyak orang yang membacanya.

Yogyakarta, April 2021

Penulis

# DAFTAR ISI

# DAFTAR GAMBAR

# DAFTAR TABEL

# DAFTAR LAMPIRAN

# INTISARI

**Analisis Klaster Hierarki untuk Data Kategorik dengan Algoritma *Divisive Hierarchical Clustering of Categorical Data* (DHCC) (Studi Kasus: Pengelompokkan Anggota UKM Mapagama)**

Oleh

Muhammad Burhan Aziz
17/412744/PA/18063

Analisis klaster adalah analisis yang bertujuan untuk mengelompokkan objek-objek yang memiliki tingkat kemiripan tinggi ke dalam suatu kelompok. Analisis klaster pada data kategorik menjadi lebih kompleks karena tidak ada ukuran kemiripan yang memiliki makna berarti diantara objek kategorik. Pada tugas akhir ini, akan dipaparkan suatu metode analisis klaster untuk data kategorik, yaitu algoritma DHCC. Algoritma ini berjalan dengan pendekatan atas-bawah. Pemisahan klaster berdasarkan analisis multikorespondensi pada matriks indikator yang menunjukkan kemunculan nilai kategorik pada suatu objek. Total inersia atau rata-rata jarak *Chi-square* pada perhitungan analisis multikorespondensi yang berupa jumlah kuadrat dari elemen matriks residual standar dapat dipandang sebagai ukuran heterogenitas suatu klaster dari prespektif analisis klaster. Dimensi pertama dari ruang yang ditransformasi berdasarkan MCA menjelaskan variansi terbesar dari rata-rata jarak *Chi-square* sehingga pemisahan klaster berdasarkan dimensi pertama akan menurunkan rata-rata jarak *Chi-Square*. Unsur optimisasi pada algoritma DHCC berjalan dengan memperbaiki pemisahan hasil dari analisis multikorespondensi. Metode ini diaplikasikan untuk melakukan segmentasi anggota UKM Mapagama periode 2020/2021. Diperoleh 8 segmen dari anggota UKM Mapagama dengan masing-masing karakteristiknya. Optimisasi yang terdapat pada algoritma DHCC terbukti meningkatkan kualitas hasil analisis klaster yang lebih baik.

Kata kunci : Analisis klaster hierarki, Data kategorik, Analisis multikorespondensi, Algoritma DHCC, Segmentasi.

# ABSTRACT

**Hierarchical Clustering Analysis for Categorical Data Using Divisive Hierarchical Clustering of Categorical Data (DHCC) Algorithm (Case Study: Grouping Members of UKM Mapagama)**

By

Muhammad Burhan Aziz
17/412744/PA/18063

Cluster analysis is an analysis that aims to group objects that have a high level of similarity into a group. Cluster analysis on categorical data becomes more complex because there is no significant similarity measure between categorical objects. In this final project, a cluster analysis method for categorical data will be presented, namely the DHCC algorithm. This algorithm runs with a top-down approach. Cluster separation is based on multiple correspondence analysis on an indicator matrix that shows the occurrence of categorical values in an object. The total inertia or the average Chi-square distance in the calculation of multiple correspondence analysis is the sum squares of elements from the standard residual matrix elements can be viewed as a measure of the heterogeneity of a cluster from the perspective of cluster analysis. The first dimension of the transformed space based on MCA explains the largest variance of the average Chi-square distance so that the cluster separation based on the first dimension will decrease the average Chi-Square distance. The optimization element in the DHCC algorithm works by improving the separation of results from multiple correspondence analysis. This method was applied to segment Mapagama UKM members for the 2020/2021 period. Obtained 8 segments from members of UKM Mapagama with their respective characteristics. The optimization contained in the DHCC algorithm is proven to improve the quality of the results of better cluster analysis.

Keywords : Hierarchical cluster analysis, Categorical data, Multiple correspondence analysis, DHCC algorithm, Segmentation.

# BAB I
# PENDAHULUAN

## 1. 1 Latar Belakang

Analisis klaster adalah salah satu teknik *data mining* yang penting dan berguna (Tan, et al., 2005). *Clustering* adalah proses *unsupervised learning* dengan tujuan mengelompokkan data yang tidak memiliki label ke dalam klaster sehingga objek dalam klaster yang sama menunjukkan kemiripan yang besar dan objek dari klaster yang berbeda menunjukkan ketidakmiripan yang besar. Terdapat banyak aplikasi dari analisis klaster misalnya mengekstraksi dari pola, mencari informasi dari data, peringkasan, dan lainnya. Mengelompokkan data kategorik menjadi salah satu bagian yang penting karena data kategorik sering digunakan oleh berbagai bidang studi misalnya ilmu sosial, psikologi, statistik, dan lainnnya. Banyaknya jumlah data kategorik yang terdapat pada kehidupan sehari-hari menyebabkan analisis klaster untuk data kategorik mendapatkan perhatian bagi para peneliti dalam beberapa kurun waktu terakhir (Xiong, et al., 2009).

Pengelompokkan data kategorik menjadi lebih kompleks karena tidak ada ukuran kemiripan yang memiliki makna berarti diantara objek kategori. Masalah lain dalam pengelompokkan data kategorik adalah bagaimana secara efektif menangani klaster yang memiliki kecenderungan lebih besar untuk dipisahkan dalam *subspace* yang berbeda. Algoritma pengelompokkan biasa yang mencari klaster berdasarkan seluruh dimensi atau variabel akan mengalami kesulitan dalam menemukan klaster di dalam *subspace* terutama ketika dimensi *subspace* kecil.

Beberapa algoritma didesain para peneliti untuk analisis klaster data kategorik. Namun algoritma tersebut memiliki beberapa kekurangan. Pertama adalah penentuan parameter misalnya asumsi jumlah klaster. Algoritma dengan parameter tertentu terkadang sulit untuk dicari parameternya, bahkan menjadi masalah jika perubahan sedikit pada parameter mengakibatkan hasil klaster yang jauh berbeda. Kedua adalah ketergantungan pada urutan data. Data dengan urutan yang berbeda dapat menghasilkan klaster yang berbeda. Tentunya algoritma yang baik adalah algoritma yang tidak bergantung pada urutan data. Contoh dari masalah ini adalah algoritma COOLCAT (Barbara, et al., 2002). Ketiga adalah tidak bisa diterapkan untuk data yang besar. Misalnya algoritma ROCK (Guha, et al., 1999) memiliki kompleksitas waktu secara kuadratik mengikuti banyaknya objek pada data. Tentunya algoritma yang baik dapat diterapkan untuk data yang besar.

Analisis klaster hierarki menghasilkan dendrogram yang menyebabkan lebih mudah dianalisis secara eksploratif, dan beberapa studi menyebutkan bahwa pendekatan secara hierarki bisa menghasilkan klaster dengan kualitas lebih baik (Tan, et al., 2005). Klaster hierarki dengan pendekatan *divisive* lebih baik dalam hal kompleksitas komputasional dan kualitas klaster jika dibandingkan dengan pendekatan *agglomerative* (Zhao & Karypis, 2005). Informasi dari distribusi data akan dilibatkan dalam pembentukan klaster dengan pendekatan *divisive*.

Algoritma DHCC didesain untuk analisis klaster hierarki data kategorik dengan tidak ada kekurangan seperti yang disebutkan sebelumnya (Xiong, et al., 2012). Karakteristik algoritma ini tidak dibutuhkan parameter, tidak terpengaruh oleh urutan dari data, dan dapat diterapkan untuk skala data yang besar.

Universitas Gadjah Mada (UGM) memiliki beberapa unit kegiatan mahasiswa (UKM) yang berguna mengembangkan pribadi dan kemampuan mahasiswa. Mahasiswa Pencinta Alam Universitas Gadjah Mada (Mapagama) sebagai salah satu UKM yang dimiliki UGM bergerak dalam kegiatan alam bebas. Mapagama juga mendorong anggotanya untuk turut melaksanakan Tri Dharma perguruan tinggi (pendidikan, penelitian, dan pengabdian masyarakat) dalam setiap kegiatan alam bebas sebagaimana anggotanya adalah mahasiswa.

Mapagama sebagai organisasi yang sifat kegiatannya sukarela merupakan jenis organisasi yang bergantung pada kesediaan anggotanya untuk berkegiatan. Hal ini disebabkan oleh tidak adanya faktor eksternal seperti kontrak yang mengikat ataupun kompensasi material yang bisa menjadi daya tarik untuk terus berkegiatan di dalamnya. Oleh karena itu, organisasi jenis tersebut rawan mengalami hilangnya anggota atau enggannya anggota untuk berkegiatan apabila mereka kehilangan faktor internal di dalam diri mereka untuk terus menjalankan organisasi.

Kondisi dari anggota Mapagama yang dinilai cukup heterogen karena terdiri dari berbagai latar belakang bisa menjadi pemicu hilangnya anggota atau enggannya anggota untuk berkegiatan. Contohnya bagaimana persepsi mereka terhadap Mapagama, hal yang ingin didapatkan mereka di Mapagama dan mungkin terdapat faktor lainnya yang menjadikan anggota dari Mapagama heterogen.

Hilangnya anggota atau keengganan anggota untuk berkegiatan dapat diatasi salah satunya dengan melihat karakteristik dari anggota yang dimiliki berupa atribut kategorik. Tentunya terdapat beberapa tipe karakteristik yang dimiliki oleh anggota Mapagama. Dengan diketahuinya karakteristik yang dimiliki oleh anggota, pengurus Mapagama dapat menentukan kebijakan atau mengambil langkah yang sesuai dan relevan dengan karakteristik yang dimiliki anggotanya.

Dalam tugas akhir ini, akan dibahas salah satu algoritma *clustering* untuk data kategorik untuk melihat karakteristik yang dimiliki oleh anggota Mapagama. Algoritma ini memiliki pendekatan *divisive* dengan pemisahan berdasarkan *multiple correspondence analysis* (MCA) yang disebut dengan *divisive hierarchical clustering of categorical data* (DHCC). Pembahasan teoritis algoritma ini akan dibahas menyeluruh  dan contoh aplikasinya dalam melihat karakteristik anggota Mapagama.

## 1. 2 Batasan Masalah

Pembatasan masalah dilakukan agar tidak terjadi penyimpangan dari tujuan awal dan pemecahan masalah. Tugas akhir ini difokuskan pada penerapan algoritma *divisive hierarchical clustering of categorical data* (DHCC) dengan berdasarkan *multiple correspondence analysis* pada matriks indikator. Klaster terbaik yang terbentuk adalah saat algoritma DHCC berhenti dan ukuran kualitas dari klaster yang dihasilkan diukur dengan indeks utilitas kategori ($CU$).

## 1. 3 Tujuan Penelitian

Tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar S1 pada Program Studi Statistika, Departemen Matematika, Fakultas MIPA, Universitas Gadjah Mada. Tujuan penulisan tugas akhir adalah sebagai berikut :

1. Mengaplikasikan algoritma DHCC untuk melakukan segmentasi dari anggota Mapagama.
2. Melihat karakteristik segmen dari anggota Mapagama yang dominan.
3. Menilai performa dari optimisasi yang terdapat pada algoritma DHCC.

## 1. 4 Manfaat Penelitian

Berdasarkan uraian latar belakang, tugas akhir ini membahas tentang algoritma DHCC untuk analisis klaster hierarki data kategorik. Penulisan ini diharapkan memberikan manfaat sebagai berikut :

1. Dengan menggunakan algoritma DHCC dapat memberikan hasil klaster yang baik serta memberikan informasi yang terdapat dari data yang dianalisis.
2. Memberikan kontribusi bagi ilmu pengetahuan terkait teknik untuk analisis klaster.
3. Sebagai bahan referensi bagi peneliti lain untuk mengembangkan bidang yang sejenis.

## 1. 5 Tinjauan Pustaka

Sukarela didefiniskan setiap aktivitas yang waktunya diberikan secara bebas untuk memberi manfaat bagi orang lain, kelompok, atau organisasi (Wilson, 2000). Dalam definisi tersebut, perilaku relawan dapat dikotomiskan lebih lanjut sebagai relawan dalam peran yang kontribusinya ditujukan pada aktivitas yang diharapkan secara formal seperti tugas terjadwal pembantu di acara penggalangan dana atau relawan peran ekstra yang mana perilaku menolong tersebut tidak secara formal diperlukan atau dihargai oleh penerima,(Cornelis, et al., 2013).

Faktor motivasi untuk menjadi sukarelawan telah mendapat perhatian yang signifikan dalam literatur yang ada. Umumnya didasarkan pada nilai, yaitu perilaku sukarela dimotivasi oleh nilai yang diharapkan individu untuk diperoleh dengan terlibat dalam perilaku tersebut (Clary & Snyder, 1999). Di antara faktor-faktor motivasi ini, yang paling sering dikutip adalah nilai altruistik atau manfaat yang didapatkan dari aktivitas tersebut (Cornelis, et al., 2013). Motivasi altruistik mencerminkan imbalan tak berwujud yang diperoleh secara intrinsik untuk tindakan sukarela itu sendiri seperti membantu orang lain (Wang, 2004). Motivator lain termasuk faktor sosial, perkembangan pribadi, faktor solidaritas, perhatian komunitas, dan faktor pelindung. Faktor sosial mencerminkan penghargaan yang diperoleh dari interaksi antar pribadi (Clary & Snyder, 1999), sedangkan pengembangan pribadi mencerminkan perolehan nilai yang diperoleh melalui pertumbuhan atau perkembangan pribadi seperti pengembangan keterampilan baru atau pengalaman kerja (Wang, 2004). Faktor solidaritas mencerminkan motivator sosial mencakup nilai yang diperoleh dari rasa memiliki dan manfaat dari jaringan (Caldwell & Andereck, 1994), sedangkan kepedulian komunitas mencerminkan pengurangan rasa kewajiban untuk memberi kembali kepada komunitas sukarela (Omoto & Snyder, 1995). Terakhir, faktor pelindung mencerminkan motivasi relawan untuk melindungi objek yang dianggap penting bagi relawan (Clary, et al., 1998).

## 1. 6 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam tugas akhir ini adalah studi literatur. Studi literatur dilakukan dengan sumber dari perpustakaan, jurnal-jurnal ilmiah, dan sumber-sumber lain yang relevan. Penulis menyelesaikan studi kasus pada tugas akhir ini dengan menggunakan bantuan *software R Studio* (*R*.4.0.3), *Microsoft Excel* 365, *Microsoft Word* 365. Data yang digunakan adalah data yang bersumber dari pengurus harian UKM Mapagama periode 2020/2021.

## 1. 7 Sistematika Penulisan

Sistematika penulisan tugas akhir ini disusun sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, batasan masalah, tujuan penelitian, manfaat penelitian, tinjauan pustaka, metodologi penelitian, dan sistematika penulisan.

BAB II : LANDASAN TEORI

Bab ini berisi tentang teori-teori dasar yang menunjang pembahasan mengenai analisis klaster. Teori tersebut diantaranya adalah skala pengukuran data, vektor, aljabar matriks.

BAB III : ANALISIS KLASTER HIERARKI DENGAN ALGORITMA DHCC

Bab ini berisi mengenai pembahasan secara teoritis algoritma DHCC dengan unsur optimisasinya.

BAB IV : STUDI KASUS

Bab ini berisi penerapan algoritma DHCC dalam analisis klaster, deskripsi dari data serta hasil dari analisis.

BAB V : PENUTUP

Bab ini berisi mengenai kesimpulan yang diperoleh dari studi kasus maupun pembahasan pada bab sebelumnya dan berisi saran dari kekurangan tugas akhir yang dilakukan.

# BAB II
# LANDASAN TEORI

Dalam bab ini dibahas teori-teori yang menjadi dasar dari metode yang menjadi topik utama pada tugas akhir ini. Dasar teori yang terkait dibahas sebagai berikut.

## 2. 1 Skala Pengukuran

Secara umum, data dapat terdiri dari dua tipe yaitu data numerik dan data kategorik. Berdasarkan skala pengukurannya data numerik dibagi menjadi dua jenis yaitu interval dan rasio, sedangkan data kategorik dibagi menjadi dua jenis yaitu nominal dan ordinal. Berikut penjelasan mengenai skala pengukuran interval dan rasio.

1. Interval

   Skala pengukuran interval memiliki karakteristik berupa adanya interval atau jarak antar tingkatan yang tetap atau jelas. Kemudian, skala pengukuran interval tidak memiliki titik 0 (nol) mutlak. Contoh dari skala pengukuran interval adalah suhu. Diketahui terdapat skala *celcius* dan *fahrenheit* dalam pengukuran suhu yang mana 0 pada skala *celcius* sama dengan 32 pada skala *fahrenheit*.
2. Rasio

   Skala pengukuran rasio memiliki karakteristik berupa adanya interval atau jarak antar tingkatan yang tetap atau jelas. Kemudian, skala pengukuran interval memiliki titik 0 (nol) mutlak. Contoh dari skala pengukuran rasio adalah tinggi. Misalnya jika sebuah benda ternyata diukur dengan tinggi 0 *centimeter*, maka jika benda tersebut diukur dengan skala *meter* memiliki tinggi 0 juga yang menunjukkan bahwa benda tersebut tidak memiliki tinggi.

Data kategorik juga memiliki dua jenis skala pengukuran yaitu nominal dan ordinal. Berikut penjelasan mengenai skala pengukuran nominal dan ordinal.

1. Nominal

   Skala pengukuran nominal digunakan untuk data kategorik yang nilainya tidak mempunyai tingkatan atau strata. Contoh dari skala pengukuran ini diantaranya adalah jenis kelamin, golongan darah, dan lain-lain. Misalnya jenis kelamin terdiri dari laki-laki dan perempuan, tidak ada urutan atau strata yang menunjukkan nilai mana yang lebih tinggi.

2. Ordinal

   Skala pengukuran ordinal digunakan untuk data kategorik yang nilainya mempunyai tingkatan atau strata. Contoh dari skala pengukuran ini adalah tingkat pendidikan dan skala likert. Misalnya skala likert memiliki rentang dari sangat tidak setuju, tidak setuju, netral, setuju, sampai sangat setuju. Urutan tersebut menandakan urutan terkait tingkat persetujuan terhadap sesuatu. Contoh lainnya adalah variabel tingkat pendidikan. Misalkan tingkat pendidikan SD stratanya lebih rendah dibandingkan dengan tingkat pendidikan SMA.

## 2. 2 Vektor

Sekumpulan data multivariat secara praktis disajikan dalam bentuk matriks. Sebuah matriks dapat dikatakan sebagai suatu kumpulan vektor. Pada data multivariat, suatu vektor umumnya dianggap sebagai satu variabel atau satu observasi. Berikut merupakan definisi dari vektor.

**Definisi 2.1 (Johnson & Wichern, 2018)** *Sebuah vektor adalah sebuah array (kumpulan nilai dengan tipe sama)* $x$ *dari* $n$ *bilangan riil* $x_1, x_2, \dots, x_n$ *yang dapat dituliskan sebagai :*

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} atau\ x^T = [x_1 \quad x_2 \quad \cdots \quad x_n]. \tag{2.1}$$

Suatu vektor $x$ secara geometrik dapat direpresentasikan sebagai suatu garis yang terhubung pada $n$ dimensi dengan komponen $x_1$ pada sumbu pertama, $x_2$ pada sumbu kedua, dan seterusnya.

## 2. 3 Aljabar Matriks

Penggunaan matriks untuk menyajikan data multivariat dan analisisnya dapat dibantu dengan operasi aljabar pada matriks. Pada sub bab ini akan dijelaskan dasar-dasar dari konsep aljabar matriks.

### 2.3.1 Pengertian Matriks

Matriks adalah sekumpulan dari vektor. Sekumpulan vektor dengan jumlah baris sebanyak $m$ dan jumlah kolom sebanyak $n$ dapat disebut dengan matriks orde $m \times n$. Berikut adalah definisi dari matriks.

**Definisi 2.2 (Anton & Rorres, 2004)** *Sebuah matriks adalah suatu segi empat siku-siku dari bilangan-bilangan. Bilangan-bilangan dalam susunan tersebut dinamakan elemen/entri dari matriks.*

Secara umum, matriks $\boldsymbol{A}$ dengan ukuran atau orde $m \times n$ adalah susunan segi empat dari bilangan dengan jumlah baris sebanyak $m$ dan jumlah kolom sebanyak $n$ yang dituliskan sebagai :

$$\boldsymbol{A} = \left[a_{ij}\right]_{m\times n} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \tag{2.2}$$

dengan $a_{ij}$ adalah elemen/entri pada baris ke-$i$ dan kolom ke-$j$ untuk $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$.

### 2.3.2 Operasi Matriks

Sebuah matriks dapat dioperasikan secara matematis seperti penjumlahan, pengurangan, perkalian, dan *transpose*. Aturan operasi pada matriks akan dijelaskan sebagai berikut.

**Definisi 2.3 (Anton & Rorres, 2004)** *Jika* $\boldsymbol{A}$ *dan* $\boldsymbol{B}$ *adalah matriks-matriks berukuran sama, maka jumlahan* $\boldsymbol{A}+\boldsymbol{B}$ *adalah matriks yang diperoleh dengan menambahkan entri-entri matriks* $\boldsymbol{A}$ *yang bersesuaian dengan entri-entri matriks* $\boldsymbol{B}$*, dan selisih* $\boldsymbol{A}-\boldsymbol{B}$ *adalah matriks yang diperoleh dengan mengurangkan entri-*

*entri matriks $\boldsymbol{A}$ dengan entri-entri matriks $\boldsymbol{B}$ yang saling bersesuaian. Matriks-matriks yang berukuran tidak sama tidak dapat dijumlahkan atau dikurangkan.*

Dalam notasi matriks, jika $\boldsymbol{A} = [a_{ij}]$ dan $\boldsymbol{B} = [b_{ij}]$ memiliki ukuran yang sama maka penjumlahan dan pengurangan matriks dinotasikan dengan :

$$(\boldsymbol{A}+\boldsymbol{B})_{ij} = (\boldsymbol{A})_{ij} + (\boldsymbol{B})_{ij} = a_{ij} + b_{ij}, \tag{2.3}$$

$$(\boldsymbol{A}-\boldsymbol{B})_{ij} = (\boldsymbol{A})_{ij} - (\boldsymbol{B})_{ij} = a_{ij} - b_{ij}. \tag{2.4}$$

**Definisi 2.4 (Anton & Rorres, 2004)** *Jika $\boldsymbol{A}$ adalah sembarang matriks dan $c$ adalah sembarang skalar, maka perkalian $c\boldsymbol{A}$ adalah matriks yang diperoleh dengan mengalikan setiap entri matriks $\boldsymbol{A}$ dengan $c$.*

Dalam notasi matriks, jika $\boldsymbol{A} = \left[a_{ij}\right]$ maka :

$$(c\boldsymbol{A})_{ij} = c(\boldsymbol{A})_{ij} = c[a_{ij}]. \tag{2.5}$$

**Definisi 2.5 (Anton & Rorres, 2004)** *Jika $\boldsymbol{A}$ adalah matriks berukuran $m \times r$ dan $\boldsymbol{B}$ adalah matriks berukuran $r \times n$, maka perkalian $\boldsymbol{AB}$ adalah matriks berukuran $m \times n$ dengan entri-entri yang didefinisikan sebagai berikut. Untuk mencari entri pada baris i kolom j pada matriks $\boldsymbol{AB}$, pilih baris i dari matriks $\boldsymbol{A}$ dan kolom j dari matriks $\boldsymbol{B}$. Kalikan entri-entri yang bersesuaian dari baris dan kolom secara bersama-sama dan kemudian jumlahkan hasil perkaliannya.*

Perkalian antara matriks $\boldsymbol{A}$ dan matriks $\boldsymbol{B}$ dengan $\boldsymbol{A} = [a_{ij}]_{m\times r}$ dan $\boldsymbol{B} = [b_{ij}]_{r\times n}$ dinotasikan dengan :

$$\boldsymbol{AB} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1r} \\ a_{21} & a_{22} & \dots & a_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mr} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r1} & b_{r2} & \dots & b_{rn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}b_{11} + \cdots + a_{1r}b_{r1} & a_{11}b_{12} + \cdots + a_{1r}b_{r2} & \dots & a_{11}b_{1n} + \cdots + a_{1r}b_{rn} \\ a_{21}b_{11} + \cdots + a_{2r}b_{r1} & a_{21}b_{12} + \cdots + a_{2r}b_{r2} & \dots & a_{21}b_{1n} + \cdots + a_{2r}b_{rn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}b_{11} + \cdots + a_{mr}b_{r1} & a_{m1}b_{12} + \cdots + a_{mr}b_{r2} & \dots & a_{m1}b_{1n} + \cdots + a_{mr}b_{rn} \end{bmatrix}. \tag{2.6}$$

**Definisi 2.6 (Anton & Rorres, 2004)** *Jika $\boldsymbol{A}$ adalah matriks sembarang berukuran $m \times n$, maka transpose dari $\boldsymbol{A}$, dinotasikan dengan $\boldsymbol{A}^T$, didefinisikan sebagai* *matriks berukuran* $n \times m$ *yang merupakan hasil dari menukar baris-baris dan kolom-kolom pada* $\boldsymbol{A}$*; artinya kolom pertama pada* $\boldsymbol{A}^T$ *adalah baris pertama pada* $\boldsymbol{A}$*, kolom kedua pada* $\boldsymbol{A}^T$ *adalah baris kedua pada* $\boldsymbol{A}$*, dan seterusnya.*

Dalam notasi matriks, matriks $\boldsymbol{A}$ dengan ukuran $m \times n$ maka *transpose* dari matriks $\boldsymbol{A}$ adalah :

$$\boldsymbol{A}^T = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{m1} \\ a_{12} & a_{22} & \dots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \dots & a_{mn} \end{bmatrix}. \tag{2.7}$$

Dapat dilihat bahwa kolom dari $\boldsymbol{A}^T$ merupakan baris dari $\boldsymbol{A}$ dan baris dari $\boldsymbol{A}^T$ merupakan kolom dari $\boldsymbol{A}$, sehingga entri pada baris $i$ dan kolom $j$ pada $\boldsymbol{A}^T$ merupakan entri pada baris $j$ dan kolom $i$ pada $\boldsymbol{A}$.

### 2.3.3 Jenis Matriks

**Matriks Bujur Sangkar**

Matriks bujur sangkar adalah matriks dengan jumlah baris dan jumlah kolom sama. Matriks $\boldsymbol{A}$ disebut matriks bujur sangkar orde $n$ jika jumlah baris dan jumlah kolom sebanyak $n$.

$$\boldsymbol{A} = [a_{ij}]_{n \times n} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}. \tag{2.8}$$

Elemen $a_{11}, a_{22}, \dots, a_{nn}$ merupakan diagonal utama dari matriks bujur sangkar $\boldsymbol{A}$.

**Matriks Diagonal**

Matriks diagonal adalah umumnya matriks bujur sangkar yang selain diagonal utama entrinya nol.

$$\boldsymbol{A} = [a_{ij}]_{n \times n} = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix}. \tag{2.9}$$

Elemen $a_{11}, a_{22}, \ldots, a_{nn}$ merupakan diagonal utama dari matriks bujur sangkar $\boldsymbol{A}$ yang entrinya mungkin terdapat nol.

**Matriks Simetris**

Matriks simetris adalah matriks bujur sangkar yang elemennya simetris pada diagonal utamanya atau *transpose* dari matriks tersebut menghasilkan matriks yang sama. Maka $a_{ij} = a_{ji}$.

**Matriks Identitas**

Matriks bujur sangkar dengan entri 1 pada diagonal utama dan entri 0 pada selain diagonal utama disebut matriks identitas yang dilambangkan dengan $\boldsymbol{I}$. $\boldsymbol{I}_n$ melambangkan matriks identitas berukuran $n \times n$.

Matriks identitas $\boldsymbol{I}_n$ jika dituliskan dalam bentuk matriks menjadi :

$$\boldsymbol{I}_n = diag(1)_{n\times n} = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}_{n\times n}. \tag{2.10}$$

Matriks identitas berperan dalam aritmatika matriks seperti bilangan 1 pada hubungan numerikal. Peran tersebut mengakibatkan $\boldsymbol{A}_{m\times n}\boldsymbol{I}_n = \boldsymbol{A}_{m\times n}$ dan $\boldsymbol{I}_m\boldsymbol{A}_{m\times n} = \boldsymbol{A}_{m\times n}$.

**Matriks Orthogonal**

Suatu sembarang matriks $\boldsymbol{A}$ merupakan matriks orthogonal jika memenuhi :

$$\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}. \tag{2.11}$$

Sifat dari matriks orthogonal sebagai berikut :

1. Invers dari matriks *orthogonal* juga merupakan matriks orthogonal.
2. Hasil kali matriks-matriks orthogonal juga merupakan matriks orthogonal.
3. Jika *A* merupakan matriks orthogonal maka $\det(\boldsymbol{A}) = 1$ atau $\det(\boldsymbol{A}) = -1$.

**Matriks Definit Positif**

Jika $\boldsymbol{A} = a_{ij}$ adalah matriks definit positif $n$, maka elemen diagonal $\boldsymbol{A}$ bernilai positif. Namun sifat tersebut tidak berlaku sebaliknya. Artinya, jika ada matriks yang elemen diagonalnya bernilai positif belum tentu matriks tersebut adalah matriks definit positif. Matriks definit positif adalah matriks non-singular, yaitu matriks yang memiliki determinan bukan nol. Untuk sembarang matriks kolom $\boldsymbol{X}$ berukuran $n \times 1$ untuk setiap $\boldsymbol{X} \neq 0$ artinya entri dari $\boldsymbol{X}$ tidak semuanya 0, maka $\boldsymbol{A}$ merupakan matriks definit positif jika memenuhi :

$$\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{X} > 0. \tag{2.12}$$

**Matriks Semidefinit Positif**

Untuk sembarang matriks kolom $\boldsymbol{X}$ berukuran $n \times 1$ untuk setiap $\boldsymbol{X} \neq 0$ artinya entri dari $\boldsymbol{X}$ tidak semuanya 0, maka $\boldsymbol{A}$ merupakan matriks semidefinit positif jika memenuhi :

$$\boldsymbol{X}^{T}\boldsymbol{A}\boldsymbol{X} \geq 0. \tag{2.13}$$

### 2.3.4 Determinan Matriks

Determinan matriks hanya terdapat pada matriks bujur sangkar. Umumnya determinan matriks digunakan untuk mencari invers matriks dan menyelesaikan sistem persamaan linear dalam matriks.

**Definisi 2.7** *Diberikan matriks bujur sangkar berukuran $n \times n$, yaitu $\boldsymbol{A} = [a_{ij}]$. Kofaktor dari matriks $\boldsymbol{A}$ dinotasikan dengan $C_{ij}$ dan determinan matriks $\boldsymbol{A}$ dinotasikan dengan $det\ (\boldsymbol{A})$ yang didefiniskan sebagai berikut :*

$$C_{ij} = (-1)^{i+j}M_{ij}. \tag{2.14}$$

$M_{ij}$ adalah minor elemen ke-$ij$ dari matriks $\boldsymbol{A}$.

Perluasan kofaktor menghasilkan determinan matriks $\boldsymbol{A}$ dengan ukuran $n \times n$ dapat dihitung dengan jumlahan dari perkalian elemen-elemen pada baris atau kolom matriks tersebut dengan kofaktornya.

### 2.3.5 *Adjoint* Matriks

**Definisi 2.8 (Anton & Rorres, 2004)** *Jika $\boldsymbol{A}$ adalah matriks berukuran $n \times n$ dan $C_{ij}$ adalah kofaktor dari $a_{ij}$ maka matriks :*

$$\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}$$

*disebut matriks kofaktor dari $\boldsymbol{A}$. Transpose dari matriks ini disebut adjoint dari matriks $\boldsymbol{A}$ dan dilambangkan dengan $adj(\boldsymbol{A})$.*

### 2.3.6 Invers Matriks

**Definisi 2.9 (Anton & Rorres, 2004)** *Jika $\boldsymbol{A}_{n\times n}$ adalah matriks bujur sangkar dan matriks $\boldsymbol{B}_{n\times n}$ dengan ukuran yang sama dapat dicari sedemikian hingga $\boldsymbol{AB} = \boldsymbol{BA} = \boldsymbol{I}$, maka $\boldsymbol{A}$ dikatakan invertibel (dapat dibalik) dan $\boldsymbol{B}$ disebut invers dari matriks $\boldsymbol{A}$, dinotasikan dengan $\boldsymbol{A}^{-1}$.*

Invers dari matriks $\boldsymbol{A}$ dapat dihitung dengan persamaan :

$$\boldsymbol{A}^{-1} = \frac{1}{\det(\boldsymbol{A})} adj(\boldsymbol{A}). \tag{2.15}$$

Persamaan (2.15) menunjukkan bahwa jika $\det(\boldsymbol{A})$ bernilai 0, maka matriks tersebut tidak memiliki invers.

Invers matriks memiliki sifat-sifat :

1. Jika $\boldsymbol{A}$ dan $\boldsymbol{B}$ adalah matriks-matriks yang invertibel dengan ukuran sama, maka $\boldsymbol{AB}$ invertibel dan $(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$.
2. Jika $\boldsymbol{A}$ matriks yang invertibel, maka :
   a. $\boldsymbol{A}^{-1}$ invertibel dan $(\boldsymbol{A}^{-1})^{-1} = \boldsymbol{A}$.
   b. $\boldsymbol{A}^{-1}$ invertibel dan $(\boldsymbol{A}^{n})^{-1} = (\boldsymbol{A}^{-1})^{n}$ untuk $n \geq 0$.
   c. Untuk skalar $k \neq 0$, matriks $k\boldsymbol{A}$ invertibel dan $(k\boldsymbol{A})^{-1} = \frac{1}{k}\boldsymbol{A}^{-1}$.

### 2.3.7 *Eigenvalue* dan *Eigenvector*

**Definisi 2.10 (Anton & Rorres, 2004)** *Jika A matriks berukuran $m \times m$, maka setiap skalar $\lambda$ memenuhi persamaan :*

$$\boldsymbol{A}x = \lambda x \tag{2.16}$$

*dengan $\lambda$ adalah vektor berukuran $m \times 1$ dan tidak entri 0 disebut eigenvalues dari $\boldsymbol{A}$. Vektor $x$ disebut eigenvectors dari $\boldsymbol{A}$ yang berhubungan dengan eigenvalues $\lambda$.*

Terkadang *eigenvalues* dan *eigenvectors* juga disebut sebagai *latents root and vectors* atau karakteristik *roots* dan vektor. Persamaan (2.16) dapat dituliskan sebagai :

$$(\boldsymbol{A} - \lambda \boldsymbol{I})x = 0 \tag{2.17}$$

maka setiap nilai *eigenvalue* $\lambda$ harus memenuhi persamaan determinan :

$$|\boldsymbol{A} - \lambda \boldsymbol{I}| = 0 \tag{2.18}$$

yang dikenal sebagai persamaan karakteristik $\boldsymbol{A}$.

### 2.3.8 *Rank* Matriks

**Definisi 2.11 (Anton & Rorres, 2004)** *Rank dari sebuah matriks adalah banyaknya baris atau kolom yang bebas linear dalam matriks tersebut dinotasikan dengan $r$.*

Jika $r = k$ maka terdapat $k$ baris yang bebas linear dan $k$ kolom yang bebas linear dalam sebuah matriks.

Berikut akan dijelaskan sifat-sifat dari *rank* matriks :

1. $r(\boldsymbol{A})$ adalah bilangan bulat positif, kecuali untuk matriks yang entrinya semua nol.
2. $r(\boldsymbol{A}_{p\times q}) \leq \min(p, q)$.
3. Jika $r(\boldsymbol{A}) \neq 0$ maka minimal terdapat satu sub matriks bujur sangkar orde $r$ dari $\boldsymbol{A}$ yang non-singular.

4. Jika $r(\boldsymbol{A}_{n\times n}) = n$ maka $\boldsymbol{A}$ non-singular, yaitu memiliki invers dan dikatakan memiliki *rank* penuh.
5. Jika $r(\boldsymbol{A}_{n\times n}) < n$ maka $\boldsymbol{A}$ singular.
6. Jika $r(\boldsymbol{A}_{p\times q}) = p, p < q$ maka $\boldsymbol{A}$ dikatakan memiliki *rank* baris penuh.
7. Jika $r(\boldsymbol{A}_{p\times q}) = q, q < p$ maka $\boldsymbol{A}$ dikatakan memiliki *rank* kolom penuh.

### 2.3.9 Nilai Singular

**Definisi 2.12** (Goldberg, 1991) *Eigenvalues positif dari* $(\boldsymbol{A}^T\boldsymbol{A})^{1/2}$ *disebut nilai singular dari* $\boldsymbol{A}$. *Jika* $\sigma$ *adalah nilai singular dari* $\boldsymbol{A}$ *maka* $\sigma^2$*adalah eigenvalues dari* $\boldsymbol{A}^T\boldsymbol{A}$.

Dengan kata lain, untuk matriks $\boldsymbol{A}$ dengan *rank* $r$ dan *eigenvalues* dari matriks $\boldsymbol{A}^T\boldsymbol{A}$ adalah $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r = \cdots = \lambda_n = 0$, maka $\sigma_i = \sqrt{\lambda_i}$ dengan $i = 1, 2, \dots, r, r+1, \dots, n$ disebut nilai singular dari matriks $\boldsymbol{A}$.

Berikut diberikan definisi nilai singular yang dihubungkan dengan vektor singular kiri dan vektor singular kanan.

**Definisi 2.13** (Goldberg, 1991) *Diberikan matriks* $\boldsymbol{A}$ *riil berukuran* $m \times n$. *Bilangan riil positif* $\sigma$ *disebut nilai singular dari matriks* $\boldsymbol{A}$ *jika ada vektor tak nol* $u \in R^m$ *dan* $v \in R^n$ *sehingga :*

$$\boldsymbol{A}v = \sigma u \; dan \; \boldsymbol{A}^T u = \sigma v.$$

*Vektor* $u$ *disebut vektor singular kiri dan* $v$ *disebut vektor singular kanan. Selanjutnya,* $(\sigma v)$ *disebut pasangan singular kanan dari* $\boldsymbol{A}$ *dan* $(\sigma u)$ *disebut pasangan singular kiri dari* $\boldsymbol{A}$.

Hubungan antara suatu matriks dengan *rank* tertentu dan nilai singular tak nol dari matriks tersebut diberikan dalam teorema berikut ini.

**Teorema 2.1** *Diberikan matriks* $\boldsymbol{A}$ *dengan rank* $r$. *Maka terdapat tepat sejumlah r nilai singular tak nol dari matriks* $\boldsymbol{A}$.

Bukti :

Misalkan *eigenvalues* dari matriks $\boldsymbol{A}$ adalah $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Hal ini berarti terdapat sejumlah $n$ *eigenvectors* $x_1, x_2, \ldots, x_n$ yang bersesuaian dengan *eigenvectors* tersebut.

Himpunan *eigenvectors* $\{x_1, x_2, \ldots, x_n\}$ membentuk basis orthogonal untuk $\boldsymbol{R}$. Dengan menormalisasikan basis orthogonal tersebut, diperoleh basis orthonormal.

Perhatikan nilai $\langle P_i, P_j \rangle$. Untuk $i \neq j$, nilai $\langle P_i, P_j \rangle = 0$, dan untuk $i = j$, nilai $\langle P_i, P_j \rangle = 1$. Akibatnya $\langle \boldsymbol{A}P_i, \boldsymbol{A}P_i \rangle = (\boldsymbol{A}P_i)^T(\boldsymbol{A}P_i) = P_i \boldsymbol{A}\boldsymbol{A}^T P_i = \lambda_i \|P_i\|^2$, akibatnya $\lambda_i > 0$.

Menurut definisi nilai singular, berlaku $\sigma_i^2 = \lambda_i = \|\boldsymbol{A}P_i\|^2$ untuk setiap $i$. *Rank* matriks $\boldsymbol{A}$ sama dengan dimensi ruang kolomnya yaitu dim $\{\boldsymbol{A}x | x \in R^m\}$. Karena diketahui *rank* $(\boldsymbol{A}) = r$, maka :

$$\boldsymbol{A}P_1 = \boldsymbol{A}P_2 = \cdots = \boldsymbol{A}P_r \neq 0 \; dan \; \boldsymbol{A}P_{r+1} = \boldsymbol{A}P_{r+2} = \cdots = \boldsymbol{A}P_n = 0.$$

Jadi diperoleh $\sigma_i \neq 0$ untuk $i = 1, 2, \ldots, r$.

### 2.3.10 *Singular Value Decomposition* (SVD)

*Singular value decomposition* (SVD) dapat dilihat dari tiga sudut pandang yang saling berhubungan. Pertama dapat dilihat sebagai sebuah metode yang mengubah variabel saling berkorelasi menjadi sekumpulan variabel yang tidak berkorelasi sebagai cara yang lebih baik untuk melihat berbagai hubungan diantara data asli. Kedua dapat dilihat sebagai metode untuk mengidentifikasi dan mengurutkan dimensi dimana variabel menunjukan variansi paling banyak. Hal tersebut terkait dengan sudut pandang SVD yang ketiga yaitu menemukan perkiraan terbaik dari data asli dengan dimensi yang lebih sedikit. Oleh karena itu, SVD dapat juga dilihat sebagai metode reduksi dimensi data (Baker, 2005).

Suatu proses dekomposisi akan memfaktorkan sebuah matriks menjadi lebih dari satu matriks. Demikian halnya dengan *singular value decomposition* atau yang lebih dikenal sebagai SVD adalah salah satu teknik dekomposisi berkaitan dengan nilai singular suatu matriks yang merupakan salah satu karakteristik matriks tersebut.

**Definisi 2.14** *Dekomposisi nilai singular matriks riil* $\boldsymbol{A}$ *berukuran* $m \times n$ *adalah faktorisasi :*

$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^T \tag{2.19}$$

*dengan* $\boldsymbol{U}$ *matriks orthogonal* $m \times m$*,* $\boldsymbol{V}$ *matriks orthogonal* $n \times n$ *dan* $\boldsymbol{\Sigma}$ *matriks diagonal* $m \times n$ *bernilai riil tak negatif yang disebut nilai singular.*

Dengan kata lain $\boldsymbol{\Sigma} = diag(\sigma_1, \sigma_2, \dots, \sigma_n)$ terurut sehingga $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$. Jika $\boldsymbol{U} = (u_1, u_2, \dots, u_n)$ dan $\boldsymbol{V} = (v_1, v_2, \dots, v_n)$ maka :

$$\boldsymbol{A} = \sum_{i=1}^{n} \sigma_i u_i v_i^T \,. \tag{2.20}$$

Menentukan SVD meliputi langkah-langkah menentukan *eigenvalues* dan *eigenvectors* dari matriks $\boldsymbol{AA}^T$ atau $\boldsymbol{A}^T\boldsymbol{A}$. *Eigenvectors* dari $\boldsymbol{A}^T\boldsymbol{A}$ membentuk kolom $\boldsymbol{V}$ sedangkan *eigenvectors* dari $\boldsymbol{AA}^T$ membentuk kolom $\boldsymbol{U}$. Nilai singular dalam $\boldsymbol{\Sigma}$ adalah angkar pangkat dua dari *eigenvalues* matriks $\boldsymbol{AA}^T$ atau $\boldsymbol{A}^T\boldsymbol{A}$ yang merupakan elemen-elemen diagonal dari $\boldsymbol{\Sigma}$ dan disusun dengan urutan menurun.

## 2. 4 Variabel Random

Pada statistika dan ilmu peluang, variabel random dideskripsikan sebagai variabel yang nilainya bergantung pada fenomena atau kejadian random. Secara formal, variabel random merupakan suatu fungsi yang memetakan setiap *outcome* pada suatu keadaan dengan bilangan riil. Pada sub bab ini akan dijelaskan konsep dasar tentang variabel random.

### 2.4.1 Pengertian Variabel Random

**Definisi 2.15 (Bain & Engelhardt, 1992)** *Suatu variabel random,* $X$*, merupakan suatu fungsi yang terdefinisi terhadap suatu ruang sampel,* $S$*, yang memetakan suatu bilangan riil,* $X(e) = x$*, dengan setiap kemungkinan hasil* $e$ *pada* $S$*.*

Huruf besar $X$ digunakan untuk menotasikan variabel random, sedangkan huruf kecil $x$ digunakan untuk menotasikan nilai-nilai yang mungkin dari variabel random. Variabel random memiliki suatu distribusi probabilitas yang menentukan probabilitas dari setiap nilai yang mungkin. Terdapat dua jenis variabel random yaitu variabel random diskrit dan variabel random kontinu.

**Definisi 2.16 (Bain & Engelhardt, 1992)** *Variabel random X dikatakan diskrit jika nilai-nilai yang mungkin dari variabel ini berhingga (finite) atau tak berhingga (infinite) tetapi tetap terhitung (countable).*

Fungsi kepadatan probabilitas suatu variabel random diskrit (*pdf*) $X$ didefinisikan dengan :

$$f(x) = P(X = x), x = x_1, x_2, \dots. \tag{2.21}$$

Fungsi distribusi kumulatif (*cdf*) dari suatu variabel random diskrit $X$ didefinisikan dengan :

$$F(x) = P(X \leq x) = \sum_{t \leq x} f(t). \tag{2.22}$$

Suatu fungsi $f(x)$ disebut sebagai distribusi kepadatan probabilitas dari variabel random diskrit jika dan hanya jika memenuhi syarat :

1. $f(x_i) \geq 0$ untuk semua $x_i$.
2. $\sum_{\forall x_i} f(x_i) = 1$.

**Definisi 2.17 (Bain & Engelhardt, 1992)** *Suatu variabel random, $X$, dikatakan kontinu jika terdapat suatu fungsi $f(x)$, yakni fungsi kepadatan probabilitas (pdf) dari $X$, sedemikian hingga CDF dapat dituliskan sebagai berikut :*

$$F(x) = \int_{-\infty}^{x} f(t)dt. \tag{2.23}$$

Suatu fungsi $f(x)$ disebut sebagai distribusi kepadatan probabilitas dari variabel random kontinu jika dan hanya jika memenuhi syarat :

1. $f(x_i) \geq 0$ untuk semua $x_i$.

2. $\int_{-\infty}^{\infty} f(x)dx = 1$.

Jika $X$ variabel random kontinu, maka probabilitas suatu kejadian dapat dihitung dengan :

$$P(a < x \leq b) = P(a \leq x \leq b) = P(a \leq x < b) = P(a < x < b)$$
$$= F_x(b) - F_x(a) = \int_a^b f(x)dx\,. \qquad (2.24)$$

### 2.4.2 Ekspektasi dan Variansi

Ekspektasi dari suatu variabel random sering juga disebut dengan harga harapan, *mean*, atau momen pertama.

**Definisi 2.18 (Bain & Engelhardt, 1992)** *Jika* $X$ *adalah suatu variabel random dengan fungsi kepadatan probabilitas* $f(x)$, *maka nilai ekspektasi dari* $X$ *didefinisikan sebagai berikut* :

$$E(X) = \begin{cases} \sum_x xf(x)\,, X\ diskrit \\ \int_{-\infty}^{\infty} xf(x)dx\,, X\ kontinu \end{cases}. \qquad (2.25)$$

Misalkan $Y = u(x)$ merupakan suatu fungsi dari variabel random $X$ yang menyebabkan $Y$ juga variabel random. Harga harapannya dapat dihitung dengan mencari distribusi $Y$. Teorema berikut menyatakan bahwa kita bisa menentukan harga harapan $Y$ melalui distribusi $X$.

**Teorema 2.2** *Jika X adalah variabel random dengan fungsi kepadatan probabilitas* $f(x)$ *dan* $u(x)$ *merupakan suatu fungsi nilai riil dengan domain nilai-nilai yang mungkin dari* $X$, *maka :*

$$E[u(X)] = \begin{cases} \sum_x u(x)f(x)\,, X\ diskrit \\ \int_{-\infty}^{\infty} u(x)f(x)dx\,, X\ kontinu \end{cases}. \qquad (2.26)$$

Bukti :

Berikut akan diberikan bukti untuk kasus kontinu dengan menggunakan transformasi satu-satu. Misalkan $Y = u(X)$ dan $x = w(y)$ maka akan diperoleh bahwa $f_Y(y) = f_X[w(y)]\frac{d}{dy}w(y)$.

$$\begin{aligned} E[u(X)] &= E[Y] \\ &= \int_{-\infty}^{\infty} y f_Y(y) dy \\ &= \int_{-\infty}^{\infty} u[w(y)] f_X[w(y)] \frac{d}{dy} w(y) dy \\ &= \int_{-\infty}^{\infty} u(x) f_X dx. \end{aligned}$$

Selanjutnya, teorema berikut akan menunjukkan sifat linearitas dari ekspektasi.

**Teorema 2.3** *Jika $X$ adalah variabel random dengan fungsi kepadatan probabilitas $f(x)$, a dan b adalah konstan, $g(x)$ dan $h(x)$ adalah fungsi nilai riil dengan domain yaitu nilai-nilai yang mungkin dari $X$, maka :*

$$E[ag(X) + bh(X)] = aE[g(x)] + bE[h(X)]. \tag{2.27}$$

Bukti :

Berikut akan diberikan bukti untuk kasus kontinu.

$$\begin{aligned} E[ag(X) + bh(X)] &= \int_{-\infty}^{\infty} [ag(X) + bh(X)] f(X) dx \\ &= a\int_{-\infty}^{\infty} g(x) f(x) dx + b\int_{-\infty}^{\infty} h(x) f(x) dx \\ &= aE[g(x)] + bE[h(X)]. \end{aligned}$$

Akibatnya, dari Teorema 2.3 diperoleh bahwa :

$$E(aX + b) = aE(X) + b.$$

Berdasarkan Teorema 2.2, apabila fungsi $u(x)$ didefinisikan $u(x) = (x-\mu)^2$ maka akan diperoleh suatu nilai ekspektasi khusus yang penting disebut dengan variansi.

**Definisi 2.19 (Bain & Engelhardt, 1992)** *Variansi dari suatu variabel random $X$ diberikan sebagai berikut :*

$$Var(X) = E[(X-\mu)^2]. \tag{2.28}$$

Variansi merupakan suatu ukuran variabilitas suatu data atau suatu nilai yang menunjukkan tingkat persebaran dalam distribusi dari suatu variabel random. Variansi biasa dinotasikan dengan $\sigma^2$ atau $V(X)$. Akar kuadrat positif dari variansi disebut dengan deviasi standar, yakni $\sigma = \sqrt{V(X)}$. Dengan menjelaskan Persamaan (2.28), akan didapatkan :

$$\begin{aligned} Var(X) &= E[X^2 - 2\mu X + \mu^2] \\ &= E(X^2) - 2\mu E(X) + \mu^2 \\ &= E(X^2) - 2\mu^2 + \mu^2 \\ &= E(X^2) - \mu^2 \,. \end{aligned} \tag{2.29}$$

Persamaan (2.29) memberikan cara yang lebih mudah untuk menghitung variansi dari $X$. Variansi terpengaruh pada perubahan skala, namun tidak dengan translasi. Teorema di bawah ini akan menjelaskan sifat tersebut.

**Teorema 2.4** *Jika $X$ adalah variabel random, $a$ dan $b$ adalah konstan, maka :*

$$Var(aX+b) = a^2 Var(X). \tag{2.30}$$

Bukti :

$$\begin{aligned} Var(aX+b) &= E[(aX + b - a\mu_x - b)^2] \\ &= E[a^2(X-\mu)^2] \\ &= a^2 Var(X). \end{aligned}$$

## 2. 5 Analisis Statistika Multivariat

Analisis statistika multivariat adalah analisis statistika yang dalam prosesnya melibatkan data multivariat. Data multivariat dapat diartikan sebagai data yang didapatkan dengan mengukur lebih dari satu variabel pada setiap anggota sampel. Dalam pengertian lain, analisis statistika multivariat dapat diartikan sebagai metode statistika yang secara simultan menganalisis banyak variabel.

Tujuan dari analisis statistika multivariat adalah sebagai berikut (Johnson & Wichern, 2018) :

1. Reduksi data.

   Pereduksian data dilakukan untuk merepresentasikan suatu keadaan secara sederhana tanpa mengorbankan informasi yang penting.
2. Pengelompokkan.

   Kelompok-kelompok yang dibentuk terdiri dari objek atau individu yang memiliki kesamaan tinggi untuk setiap kelompok yang terbentuk.
3. Melihat ketergantungan antar variabel.

   Hubungan antar variabel menjadi fokus utama dalam analisis multivariat. Apakah variabel yang ada saling dependen atau independen.
4. Prediksi.

   Memprediksi nilai satu atau lebih variabel berdasarkan variabel lainnya perlu untuk melihat apakah terdapat hubungan diantara variabel tersebut.
5. Konstruksi dan pengujian hipotesis.

   Pengujian hipotesis tertentu yang dirumuskan dalam parameter populasi multivariat. Hal ini dilakukan untuk validasi asumsi atau memperkuat kepercayaan.

### 2.5.1 Matriks Data Multivariat

Sampel data multivariat berukuran $n$ dengan $p$ variabel dapat digambarkan dalam bentuk tabel sebagai berikut (Johnson & Wichern, 2018) :

**Tabel 2.1 Penyajian data multivariat**

| | Variabel 1 | Variabel 2 | ... | Variabel $j$ | ... | Variabel $p$ |
|---|---|---|---|---|---|---|
| Objek 1 | $x_{11}$ | $x_{12}$ | ... | $x_{1j}$ | ... | $x_{1p}$ |
| Objek 2 | $x_{21}$ | $x_{22}$ | ... | $x_{2j}$ | ... | $x_{2p}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| Objek $i$ | $x_{i1}$ | $x_{i2}$ | ... | $x_{ij}$ | ... | $x_{ip}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| Objek $n$ | $x_{n1}$ | $x_{n2}$ | ... | $x_{nj}$ | ... | $x_{np}$ |

Tabel (2.1) dapat ditulis dalam bentuk matriks $X$ berukuran $n \times p$ dengan $x_{ij}$ adalah data sampel ke-$i$ pada variabel ke-$j$.

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{ij} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nj} & \dots & x_{np} \end{bmatrix}. \tag{2.31}$$

### 2.5.2 *Mean* Sampel

*Mean* sampel dari variabel ke-$j$ adalah :

$$\bar{x}_j = \sum_{i=1}^{n} x_{ij}\,, j = 1, 2, \dots, p. \tag{2.32}$$

Vektor *mean* sampel untuk $p$ variabel adalah :

$$\bar{x}^T = [\bar{x}_1 \quad \bar{x}_2 \quad \dots \quad \bar{x}_p]. \tag{2.33}$$

### 2.5.3 Variansi dan Kovariansi Sampel

Variansi sampel dari variabel ke-$j$ adalah :

$$s_j^2 = \frac{1}{n-1} \sum_{i=1}^{n} \left(x_{ij} - \bar{x}_j\right)^2, j = 1, 2, \dots, p. \tag{2.34}$$

Akar dari variansi sampel, $\sqrt{s_j^2}$, disebut sebagai deviasi standar sampel.

Kovariansi sampel variabel ke-$j$ dan variabel ke-$k$ adalah :

$$s_{jk}^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_{ij} - \bar{x}_j\right)\left(x_{ik} - \bar{x}_k\right), j\ dan\ p = 1, 2, \dots, p. \qquad (2.35)$$

Kovariansi mengukur tingkat hubungan linear antara dua variabel. Matriks variansi-kovariansi sampel dapat dituliskan sebagai berikut :

$$\boldsymbol{s}_n = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22} & \cdots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_{pp} \end{bmatrix} \qquad (2.36)$$

dengan $s_{jk} = s_{kj}$, sehingga matriks variansi-kovariansi $\boldsymbol{s}_n$ merupakan matriks simetris. Pada matriks $\boldsymbol{s}_n$, variansi sampel berada pada diagonal utama matriks tersebut dan semua kovariansi sampel dari pasangan dua variabel berada di luar diagonal utamanya.

## 2. 6 Analisis Klaster

Salah satu teknik analisis statistika multivariat adalah teknik analisis klaster. Tujuan dari teknik ini adalah mengelompokkan objek yang memiliki kemiripan berdasarkan karakteristik yang diukur. Dalam *machine learning*, analisis klaster termasuk *unsupervised learning* yaitu suatu pendekatan pengelompokkan dimana data tidak memiliki label sebelumnya. Berbeda dengan *supervised* learning yang datanya memiliki label sehingga pengelompokkan berdasarkan informasi atau *training* dari data yang ada. Contoh penggunaan analisis klaster adalah seperti segmentasi pasar, pengelompokkan produk, segmentasi citra, dan sebagainya.

### 2.6.1 Pengertian Analisis Klaster

Analisis klaster atau *clustering* adalah suatu teknik analisis yang digunakan untuk mengelompokkan objek ke dalam beberapa klaster atau kelompok sehingga objek yang berada pada klaster yang sama memiliki tingkat kemiripan yang tinggi dan objek yang berada pada klaster yang berbeda memiliki tingkat kemiripan yang rendah. Pengelompokkan objek yang berdasarkan dari data yang tidak memliki label sebelumnya atau dikenal sebagai *unsupervised learning*.

### 2.6.2 Metode Analisis Klaster

Metode analisis klaster secara praktis dibagi menjadi *partitioning clustering* dan *hierarchical clustering* (Han, et al., 2011). Penjelesan metode tersebut akan dijabarkan sebagai berikut.

1. *Partitioning clustering.*

   *Partitioning clustering* mengelompokkan titik-titik data menjadi *k* klaster yang tidak *overlap* sedemikian hingga setiap titik data berada dalam tepat satu klaster. Penentuan jumlah klaster perlu dilakukan sebagai bagian dari prosedur atau algoritma dari analisis klaster tersebut. Teknik ini akan meminimalkan suatu kriteria tertentu dengan merelokasi titik-titik data diantara klaster secara berulang sampai didapatkan partisi yang optimal. Algoritma *partitioning clustering* yang paling terkenal adalah *k-means*.

2. *Hierarchical clustering.*

   *Hierarchical clustering* membentuk klaster secara hierarki atau alami berdasarkan kedekatan atau kemiripan antar titik-titik data. Teknik analisis klaster ini memungkinkan suatu klaster memiliki sub klaster yang disusun dengan suatu struktur pohon, yang disebut sebagai dendrogram. Terdapat dua pendekatan dalam metode ini yaitu *agglomerative* (*bottom-up*) dan *divisive* (*top-down*). Pada pendekatan *agglomerative*, setiap objek data dianggap sebagai satu klaster, kemudian secara bertahap menggabungkan setiap klaster dengan nilai kemiripan terbesar, hingga setiap objek atau klaster tergabung dalam kelompok yang sama. Pendekatan *divisive* memiliki proses yang berkebalikan dengan pendekatan *agglomerative*, yakni dengan menganggap semua objek tergabung pada satu klaster yang sama, kemudian dipisah berdasarkan nilai ketidaksamaannya hingga setiap objek merepresentasikan suatu klaster yang berbeda.

### 2.6.3 Evaluasi Hasil Analisis Klaster

Mengevaluasi kualitas hasil pengelompokan merupakan tantangan besar karena tidak ada cara yang konsisten dan unik untuk melakukannya seperti pengukuran presisi dalam klasifikasi (Wu, et al., 2009). Terdapat salah satu ukuran kualitas *unsupervised* yang biasanya digunakan untuk mengevaluasi kualitas pengelompokkan data kategorik disebut dengan indeks utilitas kategori ($CU$) (Do & Kim, 2008).

Indeks utilitas kategori ($CU$) dapat dihitung dengan persamaan :

$$CU(C) = \sum_{C_i \in C} \frac{|C_i|}{n} \sum_{j=1}^{m} \sum_{v \in D_j} \left( p\left(x_j = v \middle| C_i\right)^2 - p\left(x_j = v \middle| T\right)^2 \right). \qquad (2.37)$$

Nilai $p\left(x_j = v \middle| C_i\right)$ akan selalu positif karena mengukur kemungkinan munculnya nilai kategori dalam klaster. Indeks utilitas kategori mengukur perbedaan kemungkinan munculnya nilai kategori dalam klaster dengan kemungkinan munculnya nilai kategori dalam keseluruhan data. Nilai indeks utilitas kategori yang lebih tinggi mengindikasikan klaster yang dihasilkan lebih baik sejalan dengan mengelompokkan data menjadi klaster harus meningkatkan homogenitas dalam klaster yang dilihat dari kemungkinan munculnya nilai kategori dalam klaster harus lebih tinggi dari kemungkinan munculnya nilai kategori dalam keseluruhan data. Nilai dari $CU$ akan terus meningkat seiring meningkatnya jumlah klaser yang dihasilkan. Alasan tersebut mengakibatkan dalam perbandingan hasil klaster dengan indeks utilitas kategori harus pada jumlah klaster yang sama.

## 2. 7 Analisis Interval Skala Likert

Skala likert adalah skala pengukuran yang dikembangkan oleh Likert (Sugiyono, 2011). Skala likert mempunyai empat atau lebih butir-butir pertanyaan yang dikombinasikan sehingga membentuk sebuah skor/nilai yang merepresentasikan sifat individu, misalkan sikap, pengetahuan, dan perilaku. Dalam proses analisis data, biasanya digunakan jumlahan atau rataan dari skor hasil pertanyaan yang digunakan.

Saat menanggapi pertanyaan dalam skala likert, responden menentukan tingkat persetujuan mereka terhadap suatu pernyataan dengan memilih salah satu dari pilihan yang tersedia. Biasanya disediakan dengan lima pilihan skala dengan format sebagai berikut :

1. Sangat setuju.
2. Setuju.
3. Netral.
4. Tidak setuju.
5. Sangat tidak setuju.

Skala likert kerap digunakan sebagai skala penilaian karena memberikan nilai terhadap sesuatu. Untuk keperluan analisis kualitatif, skala jawaban diberi skor misalnya :

1. Sangat setuju diberi skor 5.
2. Setuju diberi skor 4.
3. Netral diberi skor 3.
4. Tidak setuju diberi skor 2.
5. Sangat tidak setuju diberi skor 1.

Pemberian skor dapat digunakan untuk melihat kecenderungan sikap dari responden secara umum dengan menggunakan interval skala likert. Penentuan interval dapat dicari dengan mencari skor maksimal yang mungkin dari suatu pernyataan, dengan persamaan :

$$skor\ maksimal = nilai\ skala\ jawaban\ tertinggi\ \times jumlah\ responden. \quad (2.38)$$

Kemudian skor/nilai sikap dari suatu pernyataan menjumlahkan semua skor jawaban dari semua responden. Kecenderungan sikap secara umum dapat ditentukan dengan menghitung interval yaitu :

$$interval = \frac{jumlah\ skor}{skor\ maksimal} \times 100\%. \quad (2.39)$$

Interval tersebut dicocokkan dengan batas interval dari jawaban yang mungkin untuk menentukan kecenderungan secara umum. Misalkan dengan lima skala likert maka batas interval tersebut disajikan pada tabel berikut.

**Tabel 2.2 Interval skala likert**

| **Interval** | **Sikap** |
|---|---|
| 81-100 | Sangat setuju |
| 61-80 | Setuju |
| 41-60 | Netral |
| 21-40 | Tidak setuju |
| 0-20 | Sangat tidak setuju |

# BAB III
# ANALISIS KLASTER HIERARKI DENGAN ALGORITMA DHCC

Pada bab ini akan dibahas mengenai metode utama dalam tugas akhir ini yaitu algoritma DHCC untuk analisis klaster data kategorik. Metode ini adalah salah satu teknik analisis klaster dengan pendekatan *divisive* yang tidak membutuhkan parameter, tidak terpengaruh oleh urutan data, dan dapat diterapkan pada data berdimensi tinggi.

## 3. 1 Notasi dan Definisi

Diberikan $T = \{X_1, X_2, \dots, X_n\}$ adalah kumpulan data dari $n$ objek dengan setiap objek adalah vektor multidimensi dari $m$ atribut kategorik dengan masing-masing domain $D_1, D_2, \dots, D_m$. *Clustering* data $T$ adalah membagi $n$ objek menjadi beberapa kelompok yaitu $C = \{C_1, C_2, \dots, C_K\}$, dimana setiap $C_i \neq \emptyset$ $(i = 1, \dots, K)$ adalah klaster yang memenuhi $C_1 \cup \dots \cup C_K = T$, $C_i \cap C_j = \emptyset$ untuk semua $i, j = 1, \dots, K$ $(i \neq j)$. Untuk setiap kategori $v \in D_j, p(x_j = v|C_i)$ adalah probabilitas $x_j = v$ pada klaster $C_i$. Probabilitas tersebut diestimasi dengan kemunculan kategori $v$ pada klaster $C_i$ atau sama perbandingan dengan frekuensi dari $v$ dengan banyaknya anggota dari klaster $C_i$.

Notasi $\boldsymbol{I}$ menunjukkan matriks identitas. Kemudian, $\mathbf{1}$ menunjukkan vektor kolom yang dengan entri 1 semuanya. Transpose matriks dari $\boldsymbol{A}$ adalah $\boldsymbol{A}^T$. *Trace* dari matriks $\boldsymbol{A}$ adalah jumlahan elemen diagonal utama dinotasikan dengan $trace(\boldsymbol{A})$. *Row mass* vektor dari matriks $\boldsymbol{A}$ dilambangkan dengan $\boldsymbol{r}$, dimana setiap komponen dari $r_i$ adalah jumlahan dari elemen di baris $i$. *Coloumn mass* vektor dilambangkan dengan $\boldsymbol{c}$, mirip dengan *row mass* vektor. Terdapat juga *row mass* matriks yang didefinisikan sebagai matriks diagonal dengan elemen $\boldsymbol{r}$**,** ekivalen dengan *coloumn mass* matriks dengan elemen $\boldsymbol{c}$. Jika $\boldsymbol{A}$ adalah matriks biner, *row and coloumn mass* matriks dapat ditulis dengan $diag(\boldsymbol{A}\boldsymbol{A}^T)$ dan $diag(\boldsymbol{A}^T\boldsymbol{A})$. Ringkasan dari lambang yang akan digunakan disajikan pada tabel di bawah ini.

**Tabel 3.1 Ringkasan lambang**

| **Simbol** | **Deskripsi** |
|---|---|
| $n$ | Jumlah objek untuk dikelompokkan |
| $m$ | Jumlah atribut |
| $K$ | Jumlah klaster |
| $\boldsymbol{Z}$ | Matriks indikator |
| $\boldsymbol{Z}^{(p)}$ | Matriks indikator dari klaster $C_p$ |
| $D_t$ | Domain dari atribut ke-$t$ |
| $v$ | Nilai kategori |
| $J$ | Total jumlah nilai kategori |
| $\lvert C_p \rvert$ | Jumlah objek dari klaster $C_p$ |
| $C_P^L, C_P^R$ | Subklaster kiri dan kanan dari klaster $C_p$ |

## 3. 2 Perhitungan MCA pada Matriks Indikator

Matriks indikator $\boldsymbol{Z}$ dengan orde $n \times J$ setiap elemennya didefiniskan sebagai berikut :

$$z_{ij} = \begin{cases} 1 & jika\ objek\ i\ memiliki\ atribut\ ke-j \\ 0 & lainnya \end{cases}.$$

$Z_i$ melambangkan objek $X_i$ sesuai dengan representasi data matriks indikator.

Atribut kategorik dari kumpulan data diubah menjadi matriks indikator, dilambangkan dengan $\boldsymbol{Z}$, yang merupakan tabel disjungtif boolean. Dengan sekumpulan data $(T)$, banyak nilai kategorik dari domain atribut ke-$t$ dilambangkan dengan $|D_t|$. Untuk setiap atribut $(A_t)$ terdapat kolom yang sesuai sebanyak $|D_t|$. Karena itu, terdapat $J = \sum_{t=1}^{m} |D_t|$ kolom pada matriks indikator yang merepresentasikan dari semua atribut asli. Disini nilai yang identik dari atribut yang berbeda dianggap sebagai berbeda. Matriks indikator adalah matriks yang menunjukkan hubungan objek dengan nilai biner yang mengindikasikan objek memiliki kategori tertentu.

Pendekatan standar untuk *multiple correspondence analysis* (MCA) yaitu dengan menerapkan *correspondence analysis* (CA) ke matriks indikator $\boldsymbol{Z}$ (Greenacre & Blasius, 2006). MCA pada matriks indikator dapat menganalisis pola diantara objek dan nilai kategori sehingga dalam prespektif analisis klaster dapat mengelompokkan objek kategorik.

Matriks indikator ($\boldsymbol{Z}$) memiliki total jumlahan dari semua elemennya sebesar $nm$ yang menunjukkan jumlah total kemunculan semua nilai kategori, matriks korespondensi ($\boldsymbol{P}$) adalah :

$$\boldsymbol{P} = \frac{\boldsymbol{Z}}{nm} \ , \tag{3.1}$$

sehingga *row mass* vektor dari matriks korespondensi adalah :

$$\boldsymbol{r} = \left(\frac{1}{n}\right)\boldsymbol{1} \tag{3.2}$$

dan *row mass* matriksnya adalah :

$$\boldsymbol{D_r} = \left(\frac{1}{n}\right)\boldsymbol{I}. \tag{3.3}$$

Kemudian *coloumn mass* matriks dari matriks korespondensi adalah :

$$\boldsymbol{D_c} = \left(\frac{1}{nm}\right) diag(\boldsymbol{Z}^T\boldsymbol{Z}) \tag{3.4}$$

sehingga *column mass* vektor dari matriks korespondensi dapat ditulis :

$$\boldsymbol{c} = \boldsymbol{D_c} \times \boldsymbol{1}. \tag{3.5}$$

Dalam asumsi independensi antara baris dan kolom, $E(p_{ij}) = r_i c_j$ (Greenacre & Blasius, 2006). Selisih antara observasi dengan nilai ekspektasi disebut residual yaitu $p_{ij} - r_i c_j$. Normalisasi dari residual diperoleh dengan membagi residual dengan akar kuadrat dari $r_i c_j$, $s_{ij} = (p_{ij} - r_i c_j)/\sqrt{r_i c_j}$ . Dalam matriks, residual standar dinotasikan sebagai berikut :

$$\boldsymbol{S} = \boldsymbol{D_r}^{-\frac{1}{2}}(\boldsymbol{P} - \boldsymbol{r}\boldsymbol{c}^{\boldsymbol{T}})\boldsymbol{D_c}^{-\frac{1}{2}}. \tag{3.6}$$

Untuk melihat keterkaitan diantara objek, *singular value decomposition* (SVD) digunakan pada matriks $\boldsymbol{S}$, sehingga matriks $\boldsymbol{S}$ dapat ditulis :

$$\boldsymbol{S} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3.7}$$

memenuhi $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}$, dan $\sum$ adalah matriks diagonal dengan nilai singular dengan urutan semakin mengecil ($\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_s \geq 0$), dimana $s$ adalah *rank* dari matriks residual. Kolom dari $\boldsymbol{U}$ disebut vektor singular kiri dan kolom dari $\boldsymbol{V}$ disebut vektor singular kanan. Notasi $\boldsymbol{U}$ menunjukkan nilai skala untuk $n$ objek, sedangkan $\boldsymbol{V}$ menunjukkan nilai skala untuk nilai kategori $J$. Hubungan antara SVD dengan dekomposisi *eigenvalues* ditunjukkan sebagai berikut :

$$\boldsymbol{S}^T\boldsymbol{S} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^T,$$

$$\boldsymbol{S}\boldsymbol{S}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T = \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T.$$

Dari hubungan diatas, $\boldsymbol{V}$ merupakan *eigenvectors* dari $\boldsymbol{S}^T\boldsymbol{S}$ dan $\boldsymbol{U}$ merupakan *eigenvectors* dari $\boldsymbol{S}\boldsymbol{S}^T$. Kuadrat dari nilai singular yang terdapat pada $\boldsymbol{\Sigma}^2$ berkorespondensi dengan *eigenvalues*.

Hasil dari SVD memberikan komponen untuk melakukan MCA berupa koordinat utama baris dan koordinat utama kolom. Koordinat utama baris ($\boldsymbol{F}$) dan koordinat utama kolom ($\boldsymbol{G}$) dapat dihitung dengan persamaan :

$$\boldsymbol{F} = \boldsymbol{D_r}^{-\frac{1}{2}}\boldsymbol{U}\boldsymbol{\Sigma}, \tag{3.8}$$

$$\boldsymbol{G} = \boldsymbol{D_c}^{-\frac{1}{2}}\boldsymbol{V}\boldsymbol{\Sigma}. \tag{3.9}$$

Koordinat tersebut merepresentasikan hubungan antara baris dan kolom jika dipetakan pada sistem koordinat.

Dalam MCA, jumlah kuadrat dari elemen matriks residual standar disebut total inersia yang dinotasikan sebagai berikut :

$$\sum_i \sum_j s_{ij}^2 = \sum_i \sum_j \left(\frac{p_{ij} - r_i c_j}{\sqrt{r_i c_j}}\right)^2$$

$$= \sum_i \sum_j \left(\frac{\frac{z_{ij}}{nm} - \frac{z_i}{nm}\frac{z_j}{nm}}{\sqrt{\frac{z_i}{nm}\frac{z_j}{nm}}}\right)^2$$

$$= \sum_i \sum_j \frac{\left(\frac{z_{ij}}{nm} - \frac{z_i z_j}{(nm)^2}\right)^2}{\frac{z_i}{nm}\frac{z_j}{nm}}$$

$$= \sum_i \sum_j \frac{\frac{1}{(nm)^2}\left(z_{ij} - \frac{z_i z_j}{nm}\right)^2}{\frac{1}{(nm)^2}\left(z_i z_j\right)}$$

$$= \sum_i \sum_j \frac{\left(z_{ij} - \frac{z_j}{n}\right)^2}{m z_j}$$

$$= \frac{1}{mn} \sum_i \sum_j \frac{\left(z_{ij} - \frac{z_j}{n}\right)^2}{\frac{z_j}{n}} \qquad (3.10)$$

Dari prespektif *clustering*, $\sum_j \frac{\left(z_{ij} - \frac{z_j}{n}\right)^2}{\frac{z_j}{n}}$ adalah ukuran jarak diantara objek ($Z_i$) dan sekumpulan objek ($T$) yaitu jarak *Chi-square* yang dilambangkan dengan $d_{Chi}(Z_i, T)$ (Xiong, et al., 2012). Persamaan (3.10) dapat ditulis sebagai :

$$\sum_i \sum_j s_{ij}^2 = \frac{1}{mn} \sum_i d_{Chi}(Z_i, T)\,. \qquad (3.11)$$

Jadi, total inersia dapat dijelaskan sebagai rata-rata jarak *Chi-square*. Total inersia juga dapat ditulis sebagai :

$$\sum_i \sum_j s_{ij}^2 = trace(\boldsymbol{S}\boldsymbol{S}^T) = trace(\boldsymbol{\Sigma}^2) = \sum_i^s \sigma_i^2\,. \qquad (3.12)$$

Dari Persamaan (3.11) dan Persamaan (3.12) dapat terlihat bahwa rata-rata jarak *Chi-square* ekivalen dengan penjumlahan *eigenvalues* dari perhitungan MCA.

Dalam MCA, *eigenvalues* ($\sigma^2$) mewakili jumlah inersia yang mencerminkan kepentingan relatif dari dimesi yang ditransformasi. Dimensi pertama selalu menjelaskan bagian terbesar dari variansi, yang kedua menjelaskan bagian terbesar dari variansi tersisa yang tidak dapat dijelaskan, dan seterusnya. Dalam analisis klaster hierarki *divisive*, tujuan dari pemisahan klaster adalah untuk menurunkan variansi dalam sub klaster yang dihasilkan yang terdiri dari menurunkan rata-rata jarak *Chi-square* atau ekivalen dengan menurunkan jumlah *eigenvalues* dari perhitungan MCA.

### 3. 3 Optimisasi Klastering Data Kategorikal

Jarak *Chi-square* antara satu objek dengan sekumpulan objek dimanfaatkan untuk menentukan fungsi objektif. Kesamaan yang didefinisikan antara satu objek dan satu sekumpulan objek lebih bermakna daripada kesamaan pasangan objek, terutama ketika tidak ada perbandingan yang signifikan yang dapat digunakan untuk mendefinisikan kesamaan pasangan dalam analisis klaster (Xiong, et al., 2012). Perhitungan MCA pada matriks indikator melibatkan ukuran ketidaksamaan *Chi-square* antara satu objek dengan sekumpulan objek. Jadi tujuan dari optimisasi adalah meminimalkan fungsi objektif yaitu *sum Chi-square error* (SCE) :

$$SCE = \sum_{k=1}^{K} \sum_{Z_i \in C_k} d_{Chi}(Z_i, C_k) \tag{3.13}$$

dengan $d_{Chi}(Z_i, C_k)$ adalah jarak *Chi-square* antara objek ($Z_i$) dan klaster ($C_k$). Berdasarkan Persamaan (3.10), jarak *Chi-square* antara objek ($Z_i$) dan klaster ($C_k$) dituliskan dengan :

$$d_{Chi}(Z_i, C_k) = \sum_{j} \frac{\left(z_{ij} - \bar{c}_{kj}\right)^2}{\bar{c}_{kj}} \tag{3.14}$$

dengan $\bar{c}_{kj}(1 \leq j \leq J)$ adalah elemen ke-$j$ dari pusat klaster $C_k$ dan objek $Z_i$ berada pada klaster $C_k$.

Pusat klaster dapat ditentukan dengan mengoptimalkan fungsi objektif. Diferensiasi pertama pada fungsi objektif terhadap $\bar{c}_{pt}$ $(1 \leq t \leq J)$, elemen ke-$t$

dari pusat klaster dari klaster ke-*p* tetap mengoptimalkan fungsi objektif karena pada dasarnya jika setiap klaster menghasilkan *Chi-square error* yang optimal maka fungsi objektif akan optimal karena jumlahan dari *Chi-square error* yang optimal. Fungsi objektif ditulis kembali dengan bentuk :

$$SCE = \sum_{k=1}^{K} \sum_{Z_i \in C_k} \sum_{j}^{J} \frac{\left(z_{ij} - \bar{c}_{kj}\right)^2}{\bar{c}_{kj}}, \tag{3.15}$$

sehingga diperoleh :

$$\begin{aligned} \frac{\partial SCE}{\partial \bar{c}_{pt}} &= \frac{\partial}{\partial \bar{c}_{pt}} \sum_{k=1}^{K} \sum_{Z_i \in C_k} \sum_{j}^{J} \frac{\left(z_{ij} - \bar{c}_{kj}\right)^2}{\bar{c}_{kj}} \\ &= \sum_{k=1}^{K} \sum_{Z_i \in C_k} \sum_{j}^{J} \frac{\partial}{\partial \bar{c}_{pt}} \frac{\left(z_{ij} - \bar{c}_{kj}\right)^2}{\bar{c}_{kj}} \\ &= \sum_{Z_i \in C_p} \sum_{j}^{J} \frac{\partial}{\partial \bar{c}_{pt}} \frac{\left(z_{ij} - \bar{c}_{pj}\right)^2}{\bar{c}_{pj}}, \end{aligned}$$

dibawah asumsi bahwa dimensi saling independen, seperti pada MCA, maka diperoleh :

$$\begin{aligned} \frac{\partial SCE}{\partial \bar{c}_{pt}} &= \sum_{Z_i \in C_p} \frac{\partial}{\partial \bar{c}_{pt}} \frac{\left(z_{it} - \bar{c}_{pt}\right)^2}{\bar{c}_{pt}} \\ &= \sum_{Z_i \in C_p} \frac{2\left(\bar{c}_{pt} - z_{it}\right)\bar{c}_{pt} - \left(\bar{c}_{pt} - z_{it}\right)^2}{\bar{c}_{pt}^{\ 2}} \\ &= \sum_{Z_i \in C_p} \left(1 - \frac{z_{it}^{\ 2}}{\bar{c}_{pt}^{\ 2}}\right). \end{aligned}$$

Diperoleh $\bar{c}_{pt}$ yang mengoptimalkan fungsi objektif :

$$\sum_{Z_i \in C_p} \left(1 - \frac{z_{it}^{\ 2}}{\bar{c}_{pt}^{\ 2}}\right) = 0$$

$$|C_p| \sum_{Z_i \in C_p} -\frac{z_{it}^2}{\bar{c}_{pt}^2} = 0$$

$$|C_p| = \sum_{Z_i \in C_p} \frac{z_{it}^2}{\bar{c}_{pt}^2}$$

$$\bar{c}_{pt}^2 = \frac{1}{|C_p|} \sum_{Z_i \in C_p} z_{it}^2$$

$$\bar{c}_{pt} = \sqrt{\frac{1}{|C_p|} \sum_{Z_i \in C_p} z_{it}^2},$$

karena $z_{it}$ bernilai 1 atau 0, sehingga $z_{it} = z_{it}^2$ maka :

$$\bar{c}_{pt} = \sqrt{\frac{1}{|C_p|} \sum_{Z_i \in C_p} z_{it}}.$$

Pusat klaster dari setiap klaster $(C_p)$ berdasarkan optimisasi fungsi objektif adalah akar kuadrat dari kemungkinan munculnya kategori dalam klaster, dinotasikan dengan :

$$\bar{c}_{pt} = \sqrt{p(v_t|C_p)} = \sqrt{\frac{z_t}{|C_p|}}. \tag{3.16}$$

Diferensiasi kedua pada fungsi objektif terhadap $\bar{c}_{pt}$ $(1 \leq t \leq J)$, elemen ke-$t$ dari pusat klaster dari klaster ke-$p$, untuk mengetahui apakah optimisasi yang dilakukan memaksimalkan SCE atau meminimalkan SCE yaitu :

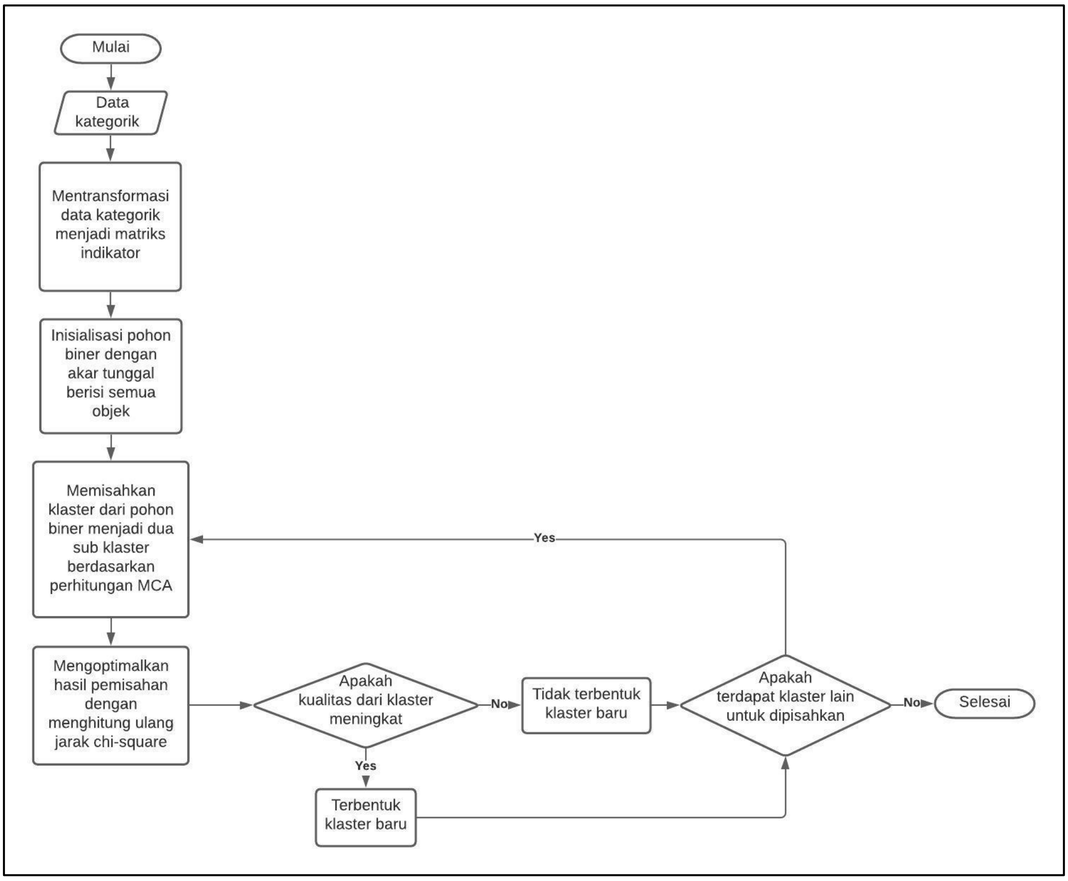
$$\begin{aligned}\frac{\partial^2 SCE}{\partial \bar{c}_{pt}^2} &= \frac{\partial}{\partial \bar{c}_{pt}} \frac{\partial SCE}{\partial \bar{c}_{pt}} \\ &= \frac{\partial}{\partial \bar{c}_{pt}} \sum_{Z_i \in C_p} \left(1 - \frac{z_{it}^2}{\bar{c}_{pt}^2}\right) \\ &= \sum_{Z_i \in C_p} -2\frac{z_{it}^2}{\bar{c}_{pt}^3}.\end{aligned}$$

Diferensiasi kedua dari fungsi objektif terhadap $\bar{c}_{pt}$ bernilai positif karena ${z_{it}}^2$ akan bernilai positif karena berpangkat genap, sedangkan ${\bar{c}_{pt}}^3$ akan bernilai negatif karena berpangkat ganjil. Terbukti pusat klaster yang dihasilkan meminimalkan fungsi objektif karena nilai dari diferensiasi kedua dari fungsi objektif terhadap $\bar{c}_{pt}$ bernilai positif.

### 3. 4 Algoritma DHCC

Berbeda dengan pendekatan *agglomerative*, algoritma DHCC dimulai dengan sebuah klaster yang berisi semua objek kategorik, dan berulang kali memilih satu klaster untuk dipisahkan menjadi dua sub klaster. Pohon biner digunakan untuk merepresentasikan struktur hierarki hasil pengelompokan dengan cara yang mirip dalam algoritma pengelompokan hierarki konvensional (Zhao & Karypis, 2005). Algoritma DHCC dijelaskan sebagai berikut (Xiong, et al., 2012) :

1. Mentransformasi data asli menjadi matriks indikator ($\boldsymbol{Z}$).
2. Menginisialisasi pohon biner dengan akar tunggal yang beranggotakan semua objek.
3. Memilih satu daun dari pohon biner, klaster ($C_p$), untuk dipisahkan menjadi dua sub klaster ($C_p^L, C_p^R$) berdasarkan perhitungan MCA pada matriks indikator ($Z^{(p)}$).
4. Mengoptimalkan hasil pemisahan pada langkah ke-3.
5. Mengulangi langkah ke-3 dan ke-4 sampai tidak ada daun pada bohon biner yang dapat dipisahkan untuk meningkatkan kualitas klaster.




**Gambar 3.1 *Flow chart* algoritma DHCC**

Pemisahan menjadi dua sub klaster yang didasarkan pada perhitungan MCA dengan tujuan sub klaster yang dihasilkan semakin homogen ekivalen dengan meminimalkan total inersia dari matriks residual standar pada sub klaster yang dihasilkan (Xiong, et al., 2012). Dimensi pertama dari ruang yang ditransformasi berdasarkan MCA menjelaskan variansi terbesar dari total inersia, maka pemisahan berdasarkan dari dimensi pertama dapat meminimalkan total inersia dari matriks residual standar ekivalen dengan sub klaster yang dihasilkan akan semakin homogen. Mengingat permasalahan efisiensi komputasi, dimensi yang lain tidak dapat menjelaskan variansi sebesar dimensi pertama sehingga untuk pemisahan hanya digunakan dimensi pertama.

Pemisahan dilakukan sebagai berikut. Untuk membagi dua klaster $(C_p)$ dengan objek sebanyak $|C_p|$, MCA diterapkan pada matriks indikator $\boldsymbol{Z}^{(p)}$ dengan orde $|C_p| \times J$ untuk mendapatkan koordinat utama baris yaitu $\boldsymbol{F}^{(p)}$. Objek $Z_i$ yang koordinat pertamanya $\boldsymbol{F}_{i1}^{(p)} \leq 0$ atau $\boldsymbol{U}_{i1}^{(p)} \leq 0$ masuk ke sub klaster kiri, yang dilambangkan dengan $C_P^L$, dan objek $Z_i$ yang koordinat pertamanya $\boldsymbol{F}_{i1}^{(p)} > 0$ atau $\boldsymbol{U}_{i1}^{(p)} > 0$ masuk ke sub klaster kanan, yang dilambangkan dengan $C_P^R$. Dalam pemisahan ini, MCA memainkan peran penting dalam menganalisis data secara keseluruhan, dan variansi serta distribusi data dari objek.

Pada optimisasi (langkah ke-4) dengan maksud meningkatkan kualitas pemisahan dengan merelokasi objek dari klaster yang dihasilkan pada pemisahan. Setelah pemisahan, tahap optimisasi mencoba meningkatkan kualitas pemisahan dengan mencari sub klaster, $C_p^L$ atau $C_p^R$ mana yang lebih cocok untuk setiap objek anggota $C_p$. Algoritma dari optimisasi dijelaskan sebagai berikut :

1. Menghitung fitur klaster dari $C_p^L$ dan $C_p^R$ yaitu $\boldsymbol{o}_p^L$, $|C_p^L|$, $\boldsymbol{o}_p^R$, dan $|C_p^R|$.
2. Untuk setiap objek $Z_i$ dalam $C_p^L$ :
   a. $\boldsymbol{o}_p^R$=$\boldsymbol{o}_p^R + Z_i$ ; $\left|C_p^R\right| = \left|C_p^R\right| + 1$.
   b. Jika $d_{Chi}\left(Z_i, C_p^R\right) < d_{Chi}\left(Z_i, C_p^L\right)$, objek $Z_i$ dipindahkan ke $C_p^R$.
   c. $\boldsymbol{o}_p^R$=$\boldsymbol{o}_p^R$−$Z_i$ ; $\left|C_p^R\right| = \left|C_p^R\right| - 1$.

   Untuk setiap objek $Z_i$ dalam $C_p^R$:
   a. $\boldsymbol{o}_p^L$=$\boldsymbol{o}_p^L + Z_i$ ; $\left|C_p^L\right| = \left|C_p^L\right| + 1$.
   b. Jika $d_{Chi}\left(Z_i, C_p^L\right) < d_{Chi}\left(Z_i, C_p^R\right)$, objek $Z_i$ dipindahkan ke $C_p^L$.
   c. $\boldsymbol{o}_p^L$=$\boldsymbol{o}_p^L$−$Z_i$ ; $\left|C_p^L\right| = \left|C_p^L\right| - 1$.
3. Memperbarui fitur klaster dari $C_p^L$ dan $C_p^R$.
4. Mengulangi langkah ke-2 dan langkah ke-3 sampai tidak ada perubahan dari anggota dari $C_p^L$ dan $C_p^R$.

Fitur $|C_p|$ adalah jumlah objek didalam klaster bersesuaian dengan $|C_p^L|$ dan $|C_p^R|$. Setiap elemen dari $\boldsymbol{o}_p$, dapat diwakili oleh $z_j$, adalah banyak kemunculan dari nilai kategori ke-$j$. Elemen ke-$j$ dari pusat klaster dapat dihitung sebagai $\bar{c}_{pj} = \sqrt{z_j/|C_p|}$, yang selanjutnya dapat digunakan untuk menghitung jarak *Chi-Square* jika objek $Z_i$ berada di klaster $C_p$. Jarak *Chi-Square* pada Persamaan (3.14) mengharuskan bahwa objek menjadi bagian dari klaster. Jadi saat menghitung ketidaksamaan antara objek $Z_i$ dan klaster $C_p$, dimana $Z_i$ tidak berada di $C_p$, untuk sementara objek $Z_i$ dipindahkan ke $C_p$, menggunakan $\boldsymbol{o}_p + Z_i$ sebagai fitur kemunculan kategori dan $\left|C_p\right| + 1$ sebagai jumlah objek di klaster.

Ukuran kualitas yang digunakan pada langkah ke-5 pada algoritma DHCC berdasarkan kombinasi kekompakan diukur dengan entropi dan pemisahan diukur dengan ukuran klaster. Kekompakan (juga disebut homogenitas) dari sebuah klaster terkait erat dengan konsep entropi (Barbara, et al., 2002). Ketidakpastian tentang kemunculan nilai kategorik saat memisahkan klaster digunakan untuk mengukur kualitas dari klaster ($C_p$). Kualitas dari klaster ($C_p$) diukur sebagai berikut :

$$Q(C_p) = \frac{\left|C_p\right|}{n}\left(\sum_{j=1}^{J} p(v_j|C_p) log p(v_j|C_p) - \sum_{j=1}^{J} p(v_j|T) log p(v_j|T)\right) \quad (3.17)$$

dengan $p(v|T)$ adalah kemungkinan munculnya kategori $(v)$ dalam keseluruhan data. Memisahkan sebuah klaster selalu meningkatkan homogenitas dari dua sub klaster yang dihasilkan, berlaku $Q(C_p) \leq Q(C_p^L) + Q(C_p^R)$. Ukuran dari klaster sebagai faktor penyeimbang dari ukuran kualitas, yang didefinisikan sebagai :

$$Q(C) = \sum_{C_i \in C} \frac{|C_i|}{n} Q(C_i). \quad (3.18)$$

Tepatnya Persamaan (3.18) digunakan pada langkah ke-5 algoritma DHCC. Semua daun pada pohon biner yang merepresentasikan klaster akan dipisahkan. Jika pemisahan daun meningkatkan kualitas klaster maka pemisahan tersebut diterima, jika tidak maka daun yang merepresentasikan klaster menjadi klaster terakhir. Pemisahan dihentikan jika tidak ada klaster yang dapat dipisahkan untuk meningkatkan kualitas klaster. Klaster yang tidak terlibat dalam pemisahan, yang kontribusinya tetap sama sebelum dan sesudahnya, dapat diabaikan dalam menentukan apakah pemisahan dapat diterima. Artinya, jika pemisahan klaster $C_p$ memenuhi $Q(\{C_p\}) < Q(\{C_p^L, C_p^R\})$ maka pemisahan tersebut diterima, jika tidak memenuhi maka pemisahan tidak diterima.

# BAB IV
# STUDI KASUS

Dalam bab ini, akan dilakukan analisis klaster anggota UKM Mapagama sebagai penerapan dari algoritma DHCC (*divisive hierarchical clustering of categorical data)*. Untuk melihat performa dari bagian optimisasi pada algoritma DHCC, analisis klaster dilakukan dengan algoritma DHCC tanpa unsur optimisasi dan algoritma DHCC dengan unsur optimisasi. Agar perbandingan menjadi seimbang, indeks utilitas kategori diukur pada jumlah klaster yang sama.

Penulis menggunakan bantuan *software R Studio (R.*4.0.3*), Microsoft Excel* 365, *Microsoft Word* 365 untuk menyelesaikan studi kasus pada bab ini. Adapun perangkat keras yang digunakan adalah Laptop Lenovo Z40-75 dengan *processor* AMD A10-7300 Radeon R6, RAM 8 GB, dan sistem operasi Windows 10 Pro 64-bit.

## 4. 1 Deskripsi Kasus

Dalam penelitian ini, terdapat dua hal utama yang ingin diteliti. Pertama adalah melakukan segmentasi anggota UKM Mapagama menggunakan algoritma DHCC. Kedua adalah melihat performa optimisasi yang terdapat pada algoritma DHCC, dengan menerapkan algoritma DHCC tanpa optimisasi dan algoritma DHCC dengan optimisasi. Unsur optimisasi yang terdapat pada algoritma DHCC akan dilihat apakah meningkatkan performa algoritma dalam analisis klaster.

UKM Mapagama sebagai organisasi mahasiswa yang sifat kegiatannya sukarela sangat tergantung pada ketersediaan anggota tersebut untuk aktif. Hal ini disebabkan oleh tidak adanya faktor eksternal seperti kontrak yang mengikat ataupun kompensasi material yang bisa menjadi daya tarik untuk terus berkegiatan di dalamnya. Oleh karena itu, organisasi jenis tersebut rawan mengalami hilangnya anggota atau enggannya anggota untuk berkegiatan apabila mereka kehilangan faktor internal di dalam diri mereka untuk terus menjalankan organisasi. Segmentasi anggota dari organisasi bersifat sukarela akan bermanfaat bagi pengurus organisasi dalam menentukan arah dan kebijakan organisasi karena menentukan kebijakan pada segmen yang tepat dapat memaksimalkan potensi yang dimiliki oleh anggota.

## 4. 2 Deskripsi Data

Data yang digunakan dalam studi kasus ini merupakan data sekunder, yakni data karakteristik anggota UKM Mapagama yang bersumber dari pengurus harian UKM Mapagama periode 2020/2021. Karakteristik anggota tersebut adalah jawaban dari pertanyaan yang diberikan dan bersifat tertutup. Pertanyaan-pertanyaan disajikan dalam tabel berikut.

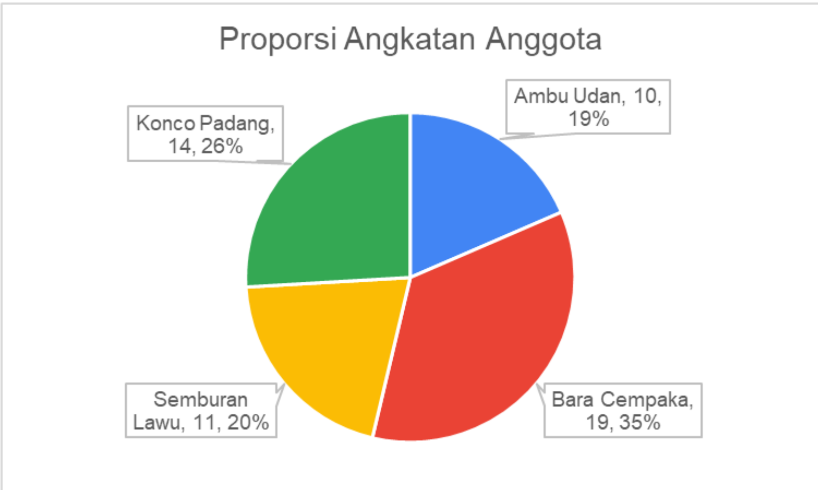
**Tabel 4.1 Deskripsi data karakteristik anggota**

| Variabel | Pertanyaan | Domain |
|---|---|---|
| P1 | Apakah ekspektasi Anda tentang Mapagama sesuai dengan apa yang Anda temui setelah menjadi Anggota Mapagama? | Ya dan Tidak |
| P2 | Apakah Anda nyaman dalam keseharian bersama anggota Mapagama baik di sekretariat ataupun tempat lainnya? | Ya dan Tidak |
| P3 | Jenis kegiatan apa yang Anda minati di Mapagama? | Petualangan, Non-Petualangan, dan Gabungan |
| | Rasa Kepemilikan Secara Psikologis | |
| P4 | Saya merasa memiliki kontrol atas bagaimana Mapagama bekerja. | Skala Likert |
| P5 | Saya merasa memberikan sesuatu untuk Mapagama. | Skala Likert |
| P6 | Saya merasa sebagai penyokong Mapagama. | Skala Likert |
| P7 | Saya pikir Mapagama banyak merepresentasikan prinsip/nilai saya | Skala Likert |
| | Sikap terhadap Kegiatan Sukarela di Mapagama | |
| P8 | Saya senang berkegiatan sukarela di Mapagama. | Skala Likert |
| P9 | Berkegiatan sukarela di Mapagama merupakan hal yang memuaskan tidak peduli berapapun waktunya. | Skala Likert |
| | Intensi Berkegiatan Sukarela di Mapagama | |
| P10 | Sangat mungkin saya berkegiatan sukarela di Mapagama. | Skala Likert |
| P11 | Saya pikir bagus bila saya berkegiatan sukarela di Mapagama. | Skala Likert |
| P12 | Bila diberi kesempatan saya akan berkegiatan sukarela di Mapagama. | Skala Likert |
| | Tekanan Waktu | |
| P13 | Saya tidak memiliki cukup waktu untuk berkegiatan sukarela. | Skala Likert |
| P14 | Berkegiatan sukarela menambah beban waktu saya. | Skala Likert |

Variabel P1 dan P2 dengan kategori ya diberi kode 1 dan kategori tidak diberi kode 2. Variabel P3 yang menunjukkan minat kegiatan diberi kode 1 untuk minat kegiatan petualangan, kode 2 untuk minat kegiatan non-petualangan, dan kode 3 untuk minat kegiatan gabungan antara petualangan dan non-petualangan. Variabel yang mengindikasikan rasa kepemilikan secara psikologis, sikap terhadap kegiatan sukarela, intensi berkegiatan sukarela, dan tekanan waktu diukur menggunakan skala likert dengan tingkatan 1 sampai 5 bersesuaian dengan kategori dari sangat tidak setuju, tidak setuju, netral, setuju, sampai sangat setuju.

## 4. 3 Analisis Deskriptif

Analisis deskriptif bertujuan untuk memberikan gambaran secara umum mengenai data yang ada. Data yang digunakan dalam studi kasus ini sebanyak 54 individu dari anggota UKM Mapagama. Anggota berasal dari empat angkatan termuda yang masih aktif sebagai anggota. Proporsi untuk tiap angkatan disajikan dalam diagram lingkaran dibawah ini :



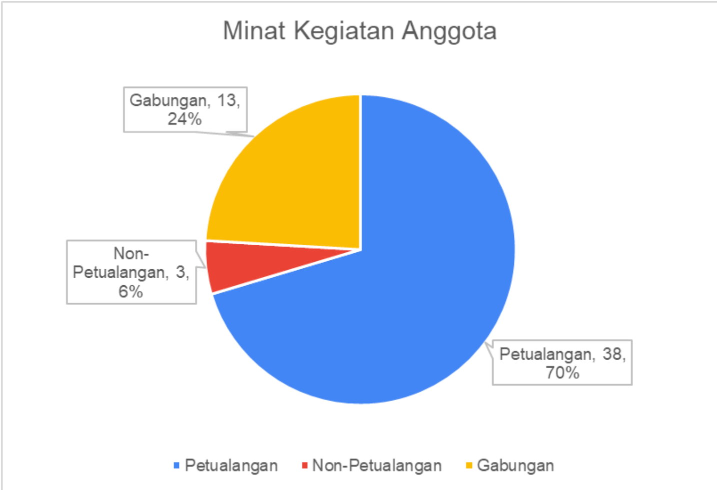

**Gambar 4.1 Proporsi angkatan anggota**

Dari angkatan ambu udan sebanyak 10 individu dengan presentase 19%, angkatan bara cempaka sebanyak 19 individu dengan presentase 35%, angkatan semburan lawu sebanyak 11 individu dengan presentase 20%, dan angkatan konco padang sebanyak 14 individu dengan presentase 26%.

Menurut data karakteristik anggota UKM mapagama, sebagian besar anggota memiliki ekspektasi yang sesuai terhadap organisasi sebanyak 74%, sisanya sebesar 26%  merasa bahwa kondisi organisasi tidak sesuai dengan apa yang diharapkan. Lalu, 89% dari anggota merasa nyaman dalam keseharian bersama sesama anggota lainnya, sedangkan sisanya sebesar 11% tidak merasa nyaman. Minat kegiatan anggota sebagian besar pada kegiatan petualangan yaitu sebanyak 70% dari anggota, hanya 6% dari anggota yang berminat pada kegiatan non-petualangan, dan sisanya sebanyak 24% berminat pada gabungan antara kegiatan petualangan dan non-petualangan. Uraian deskrptif tersebut digambarkan dengan diagram lingkaran dibawah ini :


**Gambar 4.2 Ekspektasi anggota terhadap organisasi**


**Gambar 4.3 Kenyamanan dalam keseharian sesama anggota**




**Gambar 4.4 Minat kegiatan anggota**

## 4. 4 Algoritma DHCC tanpa Optimisasi

Algoritma DHCC dimulai dengan memasukkan semua individu pada satu klaster, dan secara iteratif memilih sebuah klaster untuk dipisah menjadi dua sub klaster. Struktur hierarki digambarkan dengan pohon biner dalam setiap pemisahan sebuah klaster. Pemisahan klaster dilakukan dengan mengimplementasikan MCA pada matriks indikator dari sebuah klaster. Iterasi pemisahan akan berhenti setelah tidak ada lagi klaster yang dapat dipisahkan untuk meningkatkan kualitas dari klaster.

### 4.4.1 Iterasi ke-1

Semua objek sebanyak 54 individu dimasukkan pada sebuah klaster $(C)$ pada langkah ini. Kemudian dengan mengimplementasikan MCA pada matriks indikator, klaster $(C)$ dipisahkan menjadi dua sub klaster $(C0)$ dan $(C1)$. Matriks indikator yang dilambangkan dengan $\boldsymbol{Z}$ disajikan sebagai berikut:

$$\boldsymbol{Z}_{54\times 57} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 0 \\ 0 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 1 & \dots & 0 \end{bmatrix}. \qquad (4.1)$$

Setelah data asli ditransformasi menjadi matriks indikator, matriks korespondensi dihitung dengan membagi matriks indikator dengan jumlah total kemunculan semua nilai kategori. Akibatnya, diperoleh matriks korespondensi:

$$\boldsymbol{P}_{54\times 57} = \begin{bmatrix} 0{,}00132 & 0 & 0 & \dots & 0 \\ 0 & 0{,}00132 & 0{,}00132 & \dots & 0 \\ 0 & 0{,}00132 & 0{,}00132 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}00132 & 0 & 0{,}00132 & \dots & 0 \end{bmatrix}. \qquad (4.2)$$

Selanjutnya, dengan asumsi independensi antara objek dengan atribut kategorik akan dihitung matriks residual standar:

$$\boldsymbol{S} = \boldsymbol{D_r}^{-\frac{1}{2}}(\boldsymbol{P} - \boldsymbol{rc^T})\boldsymbol{D_c}^{-\frac{1}{2}}$$

$$= \begin{bmatrix} 0{,}0185 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 0{,}0185 \end{bmatrix}_{54\times 54}^{-\frac{1}{2}} \left( \begin{bmatrix} 0{,}00132 & 0 & 0 & \dots & 0 \\ 0 & 0{,}00132 & 0{,}00132 & \dots & 0 \\ 0 & 0{,}00132 & 0{,}00132 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}00132 & 0 & 0{,}00132 & \dots & 0 \end{bmatrix}_{54\times 57} \right.$$

$$\left. - \begin{bmatrix} 0{,}0185 \\ \vdots \\ 0{,}0185 \end{bmatrix}_{54\times 1} [0{,}053 \quad \dots \quad 0{,}0066]_{1\times 57} \right) \begin{bmatrix} 0{,}053 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 0{,}0066 \end{bmatrix}_{57\times 57}^{-\frac{1}{2}}$$

$$= \begin{bmatrix} 0{,}0109 & -0{,}0185 & -0{,}0342 & \dots & -0{,}011 \\ -0{,}0313 & 0{,}0529 & 0{,}0043 & \dots & -0{,}011 \\ -0{,}0313 & 0{,}0529 & 0{,}0043 & \dots & -0{,}011 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}0109 & -0{,}0185 & 0{,}0043 & \dots & -0{,}011 \end{bmatrix}_{54\times 57}. \qquad (4.3)$$

Keterkaitan diantara individu dilihat dengan menerapkan *singular value decomposition* (SVD) pada matriks residual standar. Sehingga matriks $S$ dapat ditulis dengan:

$$\boldsymbol{S} = \boldsymbol{U\Sigma V}^T$$

$$= \begin{bmatrix} 0{,}01696 & -0{,}02669 & 0{,}06459 & \dots & 0{,}14443 \\ 0{,}01149 & -0{,}02804 & -0{,}13183 & \dots & -0{,}08357 \\ -0{,}00668 & -0{,}07705 & -0{,}02958 & \dots & -0{,}04333 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0{,}00545 & -0{,}16994 & 0{,}11752 & \dots & 0{,}21160 \end{bmatrix}_{54\times 54}$$

$$\begin{bmatrix} 0,58 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0,53 & 0 & 0 & \dots & 0 \\ 0 & 0 & \ddots & 0 & \dots & 0 \\ 0 & 0 & 0 & 2,17 \times 10^{-17} & \dots & 0 \end{bmatrix}_{54\times 57}$$

$$\begin{bmatrix} 0,00805 & -0,08312 & 0,10207 & \dots & -0,09874 \\ -0,01361 & 0,14050 & -0,17252 & \dots & -0,05841 \\ -0,03165 & -0,03469 & -0,00653 & \dots & -0,10436 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -0,04696 & 0,27959 & -0,06653 & \dots & -0,05331 \end{bmatrix}_{57\times 57} . \quad (4.4)$$

Hasil dari SVD memberikan komponen untuk menghitung koordinat utama baris ($F$) sesuai dengan persamaan:

$$\boldsymbol{F} = \boldsymbol{D_r}^{-\frac{1}{2}} \boldsymbol{U\Sigma}$$

$$= \begin{bmatrix} 0,0185 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 0,0185 \end{bmatrix}_{54\times 54}^{-\frac{1}{2}}$$

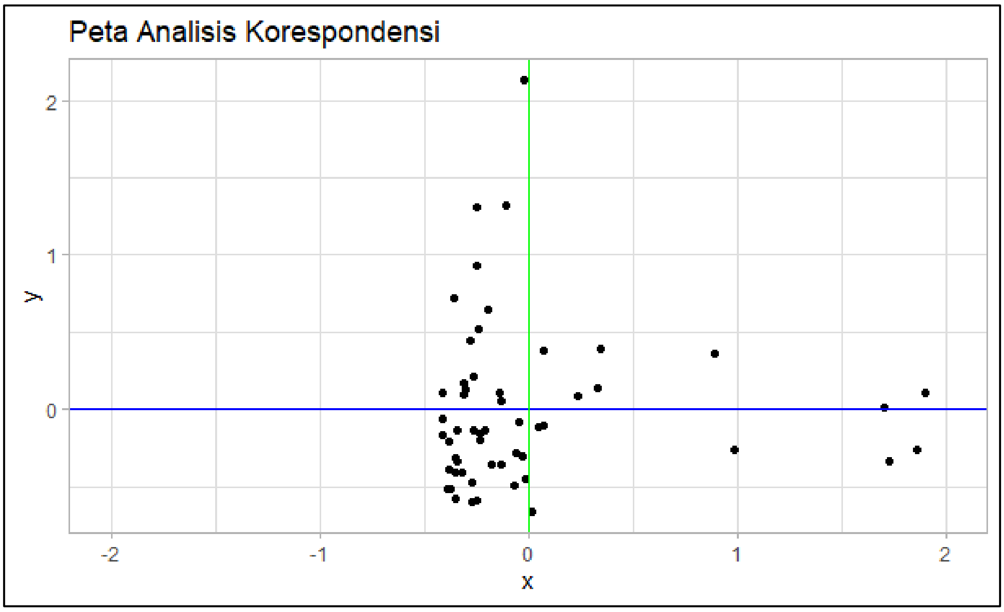
$$\begin{bmatrix} 0,01696 & -0,02669 & 0,06459 & \dots & 0,14443 \\ 0,01149 & -0,02804 & -0,13183 & \dots & -0,08357 \\ -0,00668 & -0,07705 & -0,02958 & \dots & -0,04333 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0,00545 & -0,16994 & 0,11752 & \dots & 0,21160 \end{bmatrix}_{54\times 54}$$

$$\begin{bmatrix} 0,58 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0,53 & 0 & 0 & \dots & 0 \\ 0 & 0 & \ddots & 0 & \dots & 0 \\ 0 & 0 & 0 & 2,17 \times 10^{-17} & \dots & 0 \end{bmatrix}_{54\times 57}$$

$$= \begin{bmatrix} 0,07224 & -0,10394 & 0,22082 & \dots & 2,29 \times 10^{-17} & \dots & 0 \\ 0,04894 & -0,10918 & -0,45072 & \dots & -1,33 \times 10^{-17} & \dots & 0 \\ -0,02846 & -0,29999 & -0,10115 & \dots & -6,89 \times 10^{-18} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \dots & 0 \\ 0,02319 & -0,66166 & 0,40179 & \dots & 3,37 \times 10^{-17} & \dots & 0 \end{bmatrix}_{54\times 57} . \quad (4.5)$$

Total inersia atau rata-rata jarak *Chi-square* pada perhitungan MCA yang berupa jumlah kuadrat dari elemen matriks residual standar dapat dipandang sebagai ukuran heterogenitas suatu klaster dari prespektif analisis klaster. Dimensi pertama dari ruang yang ditransformasi berdasarkan MCA menjelaskan variansi terbesar dari rata-rata jarak *Chi-square* sehingga pemisahan klaster berdasarkan dimensi pertama akan menurunkan rata-rata jarak *Chi-Square*. Dimensi yang lain diabaikan karena tidak menjelaskan variansi dari rata-rata jarak *Chi-Square* sebesar dimensi pertama dan juga agar komputasi semakin efisien. Rata-rata jarak *Chi-Square* yang semakin kecil menunjukkan suatu klaster semakin homogen. Individu yang memiliki koordinat utama baris pada dimensi pertamanya dengan nilai $\boldsymbol{F}_{i1}^{(p)} \leq 0$ masuk ke sub klaster kiri sedangkan individu dengan nilai $\boldsymbol{F}_{i1}^{(p)} > 0$ masuk ke sub klaster kanan. Kemudian diperoleh rata-rata jarak *Chi-Square* dari klaster $(C)$ sebesar 3,072. Setelah dipisahkan menjadi dua sub klaster, sub klaster kiri $(C0)$ memiliki rata-rata jarak *Chi-Square* sebesar 2,86 dan sub klaster kanan $(C1)$ memiliki rata-rata jarak *Chi-Square* sebesar 2,58. Rata-rata jarak *Chi-Square* dari masing-masing sub klaster yang dihasilkan semakin kecil setelah dipisahkan menandakan pemisahan suatu klaster menghasilkan dua sub klaster yang lebih homogen dibandingkan dengan klaster induknya.

Visualisasi dari pemisahan dapat dilihat dengan membuat plot dari individu dengan sumbu $x$ mewakili koordinat utama baris dimensi pertama dan sumbu $y$ mewakili koordinat utama baris dimensi kedua.




**Gambar 4.5 Peta analisis korespondensi**

Kualitas klaster dihitung untuk menentukan apakah pemisahan diterima atau tidak. Berdasarkan Persamaan (3.17), maka diperoleh kualitas dari klaster $(C)$:

$$Q(C) = \frac{|C|}{n}\left(\sum_{j=1}^{57} p(v_j|C)logp(v_j|C) - p(v_j|T)logp(v_j|T)\right)$$

$$= \frac{54}{54}\Big(\big(0{,}7407\log(0{,}7407) - 0{,}7407\log(7407)\big)$$

$$+ \big(0{,}2593\log(0{,}2593) - 0{,}2593\log(0{,}2593)\big) + \cdots$$

$$+ \big(0{,}0926\log(0{,}0926) - 0{,}0926\log(0{,}0926)\big)\Big)$$

$$= 0.$$

Dengan cara yang sama maka diperoleh kualitas klaster hasil pemisahan yaitu kualitas klaster dari sub klaster ($C0$) dan ($C1$) sebagai berikut:S

$$Q(C0) = \frac{|C0|}{n}\left(\sum_{j=1}^{54} p(v_j|C0)logp(v_j|C0) - p(v_j|T)logp(v_j|T)\right)$$

$$= \frac{41}{54}\Big(\big(0{,}7317\log(0{,}7317) - 0{,}7407\log(7407)\big)$$

$$+ \big(0{,}2683\log(0{,}2683) - 0{,}2593\log(0{,}2593)\big) + \cdots$$

$$+ \big(0{,}122\log(0{,}122) - 0{,}0926\log(0{,}0926)\big)\Big)$$

$$= 0{,}95554.$$

$$Q(C1) = \frac{|C1|}{n}\left(\sum_{j=1}^{50} p(v_j|C1)logp(v_j|C1) - p(v_j|T)logp(v_j|T)\right)$$

$$= \frac{13}{54}\Big(\big(0{,}7692\log(0{,}7692) - 0{,}7407\log(7407)\big)$$

$$+ \big(0{,}2308\log(0{,}2308) - 0{,}2593\log(0{,}2593)\big) + \cdots$$

$$+ \big(0{,}4615\log(0{,}4615) - 0{,}2963\log(0{,}2963)\big)\Big)$$

$$= -0{,}18921.$$

Kualitas dari klaster ($C$) yang baru berdasarkan masing-masing kualitas dari sub klaster yang dihasilkan dihitung dengan:

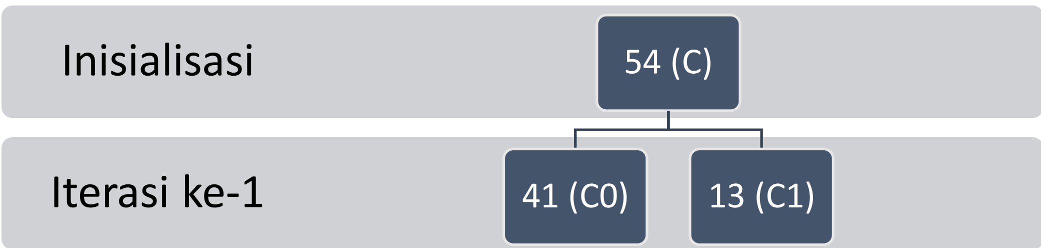
$$Q(C)_{baru} = \sum_{C_i \in C} \frac{|C_i|}{n} Q(C_i)$$

$$= \frac{41}{54} 0{,}95554 + \frac{13}{54} (-0{,}18921)$$
$$= 0{,}68.$$

**Tabel 4.2 Kualitas pemisahan iterasi ke-1**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C$ | 0 | 0,68 |

Sub klaster $C0$ terdiri dari 41 individu dan sub klaster $C1$ terdiri dari 13 individu. Pemisahan klaster $C$ dapat diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.6 Pohon biner iterasi ke-1**

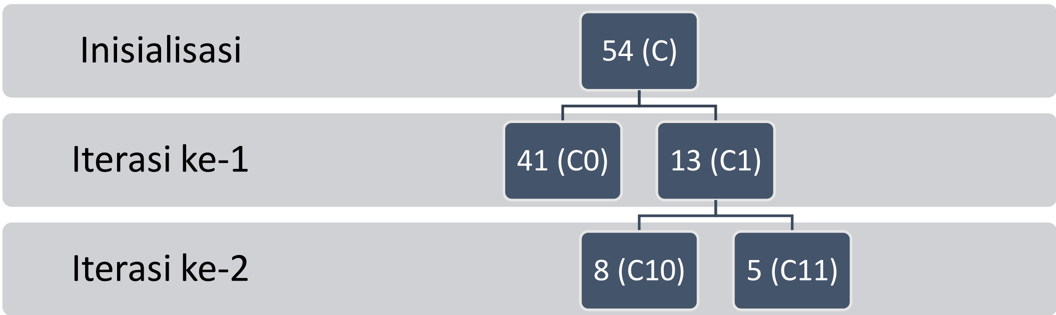
### 4.4.2 Iterasi ke-2

Klaster $C0$ akan dipisahkan menjadi sub klaster $C00$ dan $C01$, begitu juga dengan klaster $C1$ akan dipisahkan menjadi sub klaster $C10$ dan $C11$. Pemisahan $C0$ dan $C1$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator.

**Tabel 4.3 Kualitas pemisahan iterasi ke-2**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C0$ | 0,96 | 0,34 |
| $C1$ | -0,19 | -0,0074 |

Sub klaster $C00$ terdiri dari 25 individu dan sub klaster $C01$ terdiri dari 16 individu. Pemisahan klaster $C0$ tidak diterima karena kualitas dari klaster menurun setelah dipisahkan menjadi dua sub klaster. Kemudian sub klaster $C10$ terdiri dari 8 individu dan sub klaster $C11$ terdiri dari 5 individu. Pemisahan klaster $C1$ diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.7 Pohon biner iterasi ke-2**

### 4.4.3 Iterasi ke-3

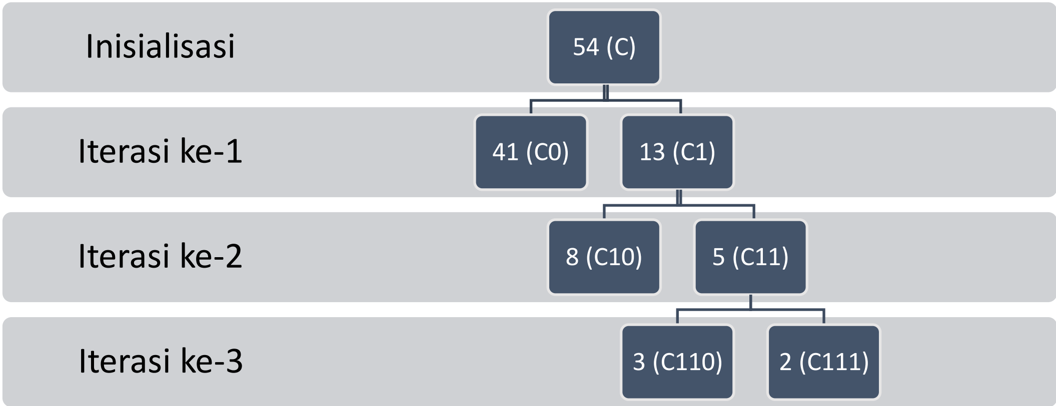
Klaster $C10$ akan dipisahkan menjadi sub klaster $C100$ dan $C101$, begitu juga dengan klaster $C11$ akan dipisahkan menjadi sub klaster $C110$ dan $C111$. Pemisahan $C10$ dan $C11$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator.

**Tabel 4.4 Kualitas pemisahan iterasi ke-3**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C10$ | 0,018 | 0,0027 |
| $C11$ | -0,109 | -0,001 |

Sub klaster $C100$ terdiri dari 3 individu dan sub klaster $C101$ terdiri dari 5 individu. Pemisahan klaster $C10$ tidak diterima karena kualitas dari klaster menurun setelah dipisahkan menjadi dua sub klaster. Kemudian sub klaster $C110$ terdiri dari 3 individu dan sub klaster $C111$ terdiri dari 2 individu. Pemisahan klaster $C11$

diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.8 Pohon biner iterasi ke-3**

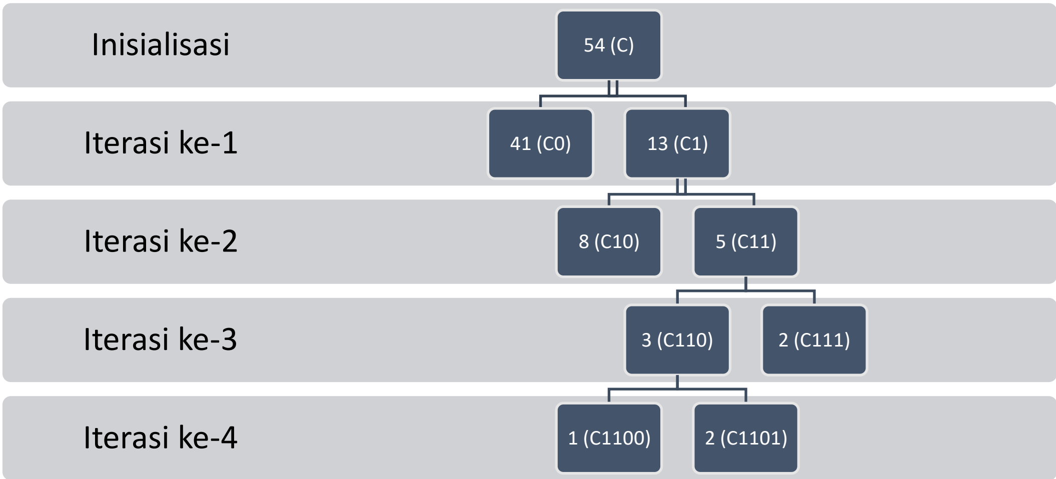
### 4.4.4 Iterasi ke-4

Klaster $C110$ akan dipisahkan menjadi sub klaster $C1100$ dan $C1101$, begitu juga dengan klaster $C111$ akan dipisahkan menjadi sub klaster $C1110$ dan $C1111$. Pemisahan $C110$ dan $C111$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator.

**Tabel 4.5 Kualitas pemisahan iterasi ke-4**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C110$ | -0,0011 | 0,0029 |
| $C111$ | 0,043 | 0,0027 |

Sub klaster $C1100$ terdiri dari 1 individu dan sub klaster $C1101$ terdiri dari 2 individu. Pemisahan klaster $C110$ diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Kemudian sub klaster $C1110$ terdiri dari 1 individu dan sub klaster $C1111$ terdiri dari 1 individu. Pemisahan klaster $C111$ tidak diterima karena kualitas dari klaster menurun setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.9 Pohon biner iterasi ke-4**

### 4.4.5 Iterasi ke-5

Klaster $C1101$ akan dipisahkan menjadi sub klaster $C11010$ dan $C11011$, sedangkan dengan klaster $C1100$ tidak akan dipisahkan lebih lanjut karena hanya memiliki satu anggota. Pemisahan $C1101$ dilakukan dengan mengimplimentasikan MCA pada matriks indikator.

**Tabel 4.6 Kualitas pemisahan iterasi ke-5**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C1101$ | 0,038 | 0,0027 |

Sub klaster $C11010$ terdiri dari 1 individu dan sub klaster $C11011$ terdiri dari 1 individu. Pemisahan klaster $C1101$ tidak diterima karena kualitas dari klaster menurun setelah dipisahkan menjadi dua sub klaster. Pada iterasi ini algoritma DHCC berhenti karena tidak ada lagi klaster yang dapat dipisahkan untuk meningkatkan kualitas klaster, sehingga pohon biner tidak berubah. Hasil akhir dari klaster hierarki diringkas pada tabel dibawah ini :

**Tabel 4.7 Hasil klaster algoritma DHCC tanpa optimisasi**

| **Klaster** | **Jumlah Anggota Klaster** |
|---|---|
| $C0$ | 41 |
| $C10$ | 8 |
| $C111$ | 2 |
| $C1100$ | 1 |
| $C1101$ | 2 |

## 4. 5 Algoritma DHCC dengan Optimisasi

Unsur optimisasi pada algoritma DHCC bekerja setelah pemisahan klaster menjadi dua sub klaster dengan MCA. Optimisasi tersebut mengkoreksi dari pemisahan dengan memindahkan anggota dari sub klaster yang memiliki jarak lebih dekat dengan dua sub klaster yang dihasilkan. Algoritma akan berhenti setelah tidak ada lagi klaster yang dapat dipisahkan untuk meningkatkan kualitas dari klaster setelah dikoreksi dengan unsur optimisasi.

### 4.5.1 Iterasi ke-1

Semua objek sebanyak 54 individu dimasukkan pada sebuah klaster ($C$) pada langkah ini. Kemudian dengan mengimplementasikan MCA pada matriks indikator, klaster ($C$) dipisahkan menjadi dua sub klaster ($C$0r) dan ($C$1r). Anggota sub klaster yang memiliki jarak lebih dekat dengan sebuah sub klaster akan dipindahkan pada optimisasi.

Penghitungan jarak *Chi-Square* yang mengharuskan individu merupakan bagian suatu klaster maka untuk melihat individu lebih dekat dengan sebuah sub klaster yang dihasilkan, individu tersebut perlu ditambahkan secara sementara pada sub klaster yang lainnya. Pemisahan awal berdasarkan MCA menghasilkan 41 individu berada di sub klaster kiri dan 13 individu berada di sub klaster kanan. Misalkan diambil individu pertama pada sub klaster kiri, maka jarak *Chi-Square* individu tersebut dengan sub klaster kiri:

$$
\begin{aligned}
\overline{c0}_j &= \sqrt{\frac{z_j}{|C0|}} \\
&= \begin{bmatrix} \sqrt{\frac{30}{41}} & \sqrt{\frac{11}{41}} & \sqrt{\frac{39}{41}} & \dots & \sqrt{\frac{5}{41}} \end{bmatrix}_{1\times 57} \\
&= [0{,}855 \quad 0{,}518 \quad 0{,}975 \quad \dots \quad 0{,}349]_{1\times 57}.
\end{aligned}
$$

$$
\begin{aligned}
d_{Chi}(Z_1, C0) &= \frac{(0-0{,}855)^2}{0{,}855} + \frac{(1-0{,}518)^2}{0{,}518} + \frac{(1-0{,}975)^2}{0{,}975} + \cdots \\
&\quad + \frac{(0-0{,}349)^2}{0{,}349} \\
&= 25{,}003.
\end{aligned}
$$

Untuk menghitung jarak *Chi-Square* individu pertama dengan sub klaster kanan, individu tersebut secara sementara dipindahkan ke sub klaster kanan. Sehingga jarak *Chi-Square*:

$$
\begin{aligned}
\overline{c1}_j &= \sqrt{\frac{z_j}{|C1|}} \\
&= \begin{bmatrix} \sqrt{\frac{10}{14}} & \sqrt{\frac{4}{14}} & \sqrt{\frac{10}{14}} & \dots & \sqrt{\frac{0}{14}} \end{bmatrix}_{1\times 57} \\
&= [0{,}845 \quad 0{,}535 \quad 0{,}845 \quad \dots \quad 0]_{1\times 57}.
\end{aligned}
$$

$$
\begin{aligned}
d_{Chi}(Z_1, C1) &= \frac{(0-0{,}845)^2}{0{,}845} + \frac{(1-0{,}535)^2}{0{,}535} + \frac{(1-0{,}845)^2}{0{,}845} + \cdots + \frac{(0-0)^2}{0} \\
&= 23{,}629.
\end{aligned}
$$

Berarti individu pertama pada sub klaster kiri ($C0$) dipindahkan ke sub klaster kanan ($C1$) karena jarak *Chi-Square* individu pertama dengan sub klaster kanan lebih kecil. Langkah ini dilakukan untuk setiap individu pada setiap sub klaster dan diulangi sampai tidak ada lagi individu yang dapat dipindahkan.

**Tabel 4.8 Kualitas pemisahan iterasi ke-1**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C$ | 0 | 0,88 |

Sub klaster $C0r$ terdiri dari 38 individu dan sub klaster $C1r$ terdiri dari 16 individu. Pemisahan klaster $C$ dapat diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :


**Gambar 4.10 Pohon biner iterasi ke-1**

### 4.5.2 Iterasi ke-2

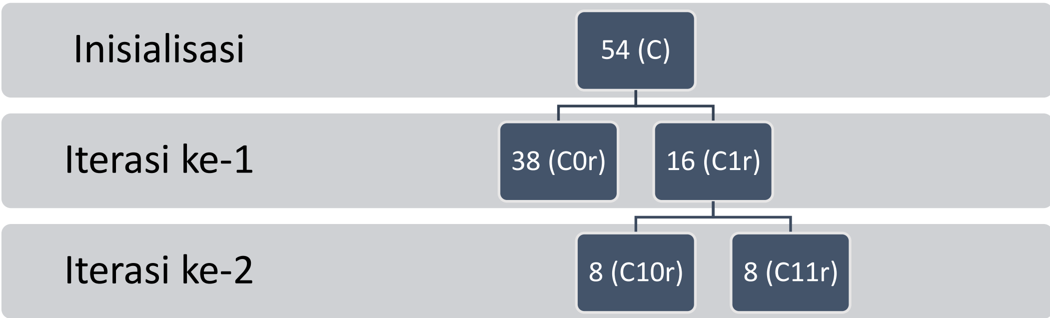
Klaster $C0r$ akan dipisahkan menjadi sub klaster $C00r$ dan $C01r$, begitu juga dengan klaster $C1r$ akan dipisahkan menjadi sub klaster $C10r$ dan $C11r$. Pemisahan $C0r$ dan $C1r$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator. Anggota sub klaster yang memiliki jarak lebih dekat dengan sebuah sub klaster akan dipindahkan pada optimisasi.

**Tabel 4.9 Kualitas pemisahan iterasi ke-2**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C0r$ | 1,43 | 0,42 |
| $C1r$ | -0,42 | -0,023 |

Sub klaster $C00r$ terdiri dari 23 individu dan sub klaster $C01r$ terdiri dari 15 individu. Pemisahan klaster $C0r$ tidak diterima karena kualitas dari klaster menurun setelah dipisahkan menjadi dua sub klaster. Kemudian sub klaster $C10r$

terdiri dari 8 individu dan sub klaster $C11r$ terdiri dari 8 individu. Pemisahan klaster $C1r$ diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.11 Pohon biner iterasi ke-2**

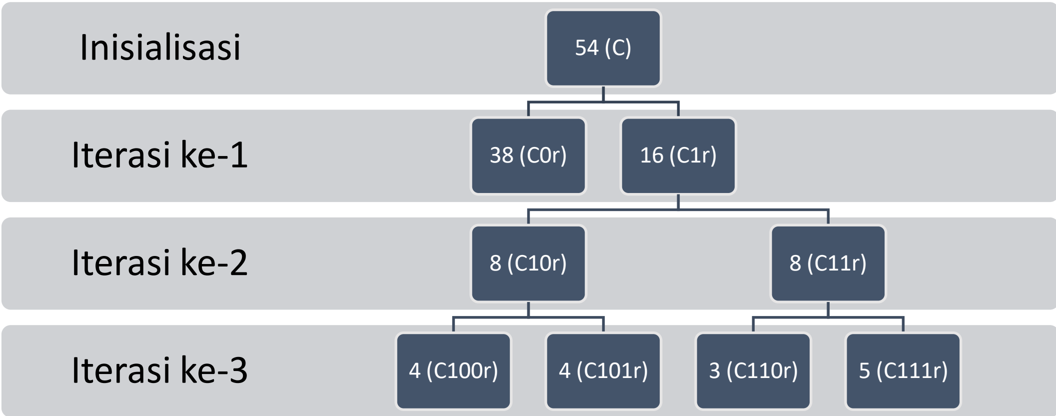
### 4.5.3 Iterasi ke-3

Klaster $C10r$ akan dipisahkan menjadi sub klaster $C100r$ dan $C101r$, begitu juga dengan klaster $C11r$ akan dipisahkan menjadi sub klaster $C110r$ dan $C111r$. Pemisahan $C10r$ dan $C11r$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator. Anggota sub klaster yang memiliki jarak lebih dekat dengan sebuah sub klaster akan dipindahkan pada optimisasi.

**Tabel 4.10 Kualitas pemisahan iterasi ke-3**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C10r$ | -0,031 | -0,0031 |
| $C11r$ | -0,12 | -0,0023 |

Sub klaster $C100r$ terdiri dari 4 individu dan sub klaster $C101r$ terdiri dari 4 individu. Pemisahan klaster $C10r$ diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Kemudian sub klaster $C110r$ terdiri dari 3 individu dan sub klaster $C111r$ terdiri dari 5 individu. Pemisahan klaster $C11r$ diterima karena kualitas dari klaster meningkat setelah dipisahkan menjadi dua sub klaster. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.12 Pohon biner iterasi ke-3**

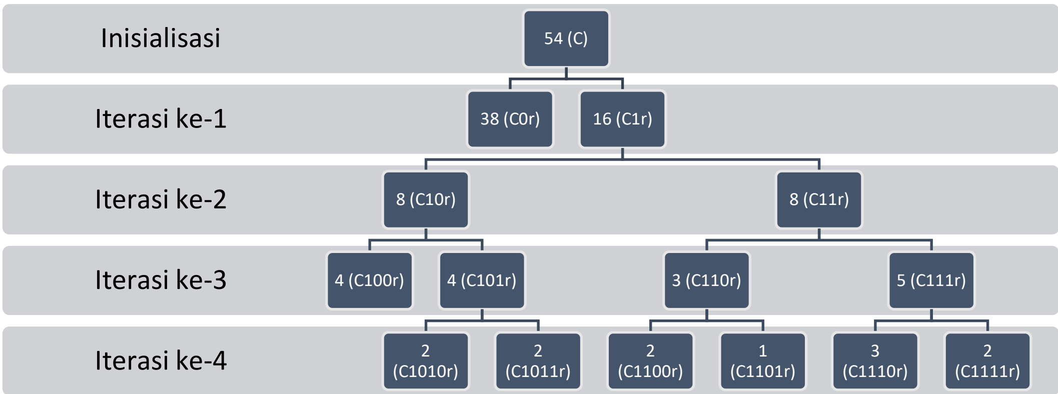
### 4.5.4 Iterasi ke-4

Klaster $C100r$ akan dipisahkan menjadi sub klaster $C1000r$ dan $C1001r$, klaster $C101r$ akan dipisahkan menjadi sub klaster $C1010r$ dan $C1011r$, klaster $C110r$ akan dipisahkan menjadi sub klaster $C1100r$ dan $C1101r$, yang terakhir klaster $C111r$ akan dipisahkan menjadi sub klaster $C1110r$ dan $C1111r$. Pemisahan $C100r$, $C101r$, $C110r$, dan $C111r$ dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator. Anggota sub klaster yang memiliki jarak lebih dekat dengan sebuah sub klaster akan dipindahkan pada optimisasi.

**Tabel 4.11 Kualitas pemisahan iterasi ke-4**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C100r$ | 0,017 | 0,0012 |
| $C101r$ | -0,06 | 0,0023 |
| $C110r$ | -0,033 | 0,0031 |
| $C111r$ | -0,006 | 0,0027 |

Pada iterasi ini, klaster $C101r$, $C110r$, dan $C111r$ diterima karena kualitas klaster meningkat setelah dipisahkan menjadi dua sub klaster. Pemisahan klaster $C100r$ tidak diterima karena kualitas klaster menurun setelah dipisahkan menjadi dua sub klaster. Sub klaster $C1010r$ terdiri dari 2 individu, $C1011r$ terdiri dari 2 individu, $C1100r$ terdiri dari 2 individu, $C1101r$ terdiri dari 1 individu, $C1110r$ terdiri dari 3 individu, dan yang terakhir sub klaster $C1111r$ terdiri dari 2 individu. Tingkatan pemisahan klaster pada langkah ini digambarkan dengan pohon biner di bawah ini :




**Gambar 4.13 Pohon biner iterasi ke-4**

### 4.5.5 Iterasi ke-5

Klaster $C1010r$ akan dipisahkan menjadi sub klaster $C10100r$ dan $C10101r$, klaster $C1011r$ akan dipisahkan menjadi sub klaster $C10110r$ dan $C10111r$, klaster $C1100r$ akan dipisahkan menjadi sub klaster $C11000r$ dan $C11001r$, klaster $C1110r$ akan dipisahkan menjadi sub klaster $C11100r$ dan $C11101r$, yang terakhir klaster $C1111r$ akan dipisahkan menjadi sub klaster $C11110r$ dan $C11111r$. Pemisahan dilakukan dengan mengimplimentasikan MCA pada masing-masing matriks indikator. Anggota sub klaster yang memiliki jarak lebih dekat dengan sebuah sub klaster akan dipindahkan pada optimisasi.

**Tabel 4.12 Kualitas pemisahan iterasi ke-5**

| Klaster | Sebelum pemisahan | Setelah pemisahan |
|---|---|---|
| $C1010r$ | 0,039 | 0,0027 |
| $C1011r$ | 0,024 | 0,0026 |

| | | |
|---|---|---|
| $C1100r$ | 0,043 | 0,0028 |
| $C1110r$ | 0,032 | 0,0041 |
| $C1111r$ | 0,024 | 0,0025 |

Pada iterasi ini algoritma DHCC berhenti karena tidak ada lagi klaster yang dapat dipisahkan untuk meningkatkan kualitas klaster, sehingga pohon biner tidak berubah. Hasil akhir dari klaster hierarki diringkas pada tabel dibawah ini :

**Tabel 4.13 Hasil klaster algoritma DHCC dengan optimisasi**

| **Klaster** | **Jumlah Anggota Klaster** |
|---|---|
| $C0r$ | 38 |
| $C100r$ | 4 |
| $C1010r$ | 2 |
| $C1011r$ | 2 |
| $C1100r$ | 2 |
| $C1101r$ | 1 |
| $C1110r$ | 3 |
| $C1111r$ | 2 |

## 4. 6 Performa Optimisasi

Pada algoritma DHCC tanpa optimisasi diperoleh pemisahan terbaik sampai pada 5 klaster, sedangkan pada algoritma DHCC dengan optimisasi diperoleh pemisahan terbaik sampai pada 8 klaster. Kedua hasil klaster tersebut akan dibandingkan kualitasnya dengan indeks utilitas kategori ($CU$). Perbandingan hasil klaster dilakukan pada banyak klaster yang sama karena nilai $CU$ akan terus meningkat dengan banyaknya jumlah klaster yang dihasilkan. Perbandingan untuk kedua hasil klaster dibandingkan pada 5 klaster yang mana algoritma DHCC tanpa optimisasi memisahkan terbaik sampai 5 klaster dan algoritma DHCC dengan optimisasi hanya digunakan sampai pemisahan 5 klaster.

**Tabel 4.14 Perbandingan algoritma DHCC tanpa optimisasi dan algoritma DHCC dengan optimisasi**

| Metode | Indeks Utilitas Kategori ($CU$) |
|---|---|
| Algoritma DHCC tanpa optimisasi | 1,683 |
| Algoritma DHCC dengan optimisasi | 2,026 |

Dari tabel perbandingan diatas, terlihat bahwa algoritma DHCC dengan optimisasi menghasilkan kualitas yang lebih baik karena memiliki indeks utilitas kategori ($CU$) yang lebih tinggi. Nilai indeks utilitas kategori ($CU$) yang tinggi menandakan kualitas klaster yang dihasilkan baik karena homogenitas objek dalam klaster akan meningkat akibat dari pemisahan menjadi klaster. Dengan demikian, unsur optimisasi yang ada pada algoritma DHCC terbukti meningkatkan homogenitas dari objek pada klaster yang dihasilkan.

## 4. 7 Karakteristik Anggota Klaster

Pengidentifikasian dari klaster adalah menjadi tahap yang penting pada analisis klaster klaster untuk melihat karakteristik dari tiap klaster yang dihasilkan. Upaya melihat karakteristik dari klaster berguna dalam meningkatkan informasi yang dihasilkan dari analisis klaster. Dalam kasus ini, UKM Mapagama sebagai organisasi dapat melihat bahwa terdapat beberapa kelompok dari anggotanya. Hasil klaster yang akan dilihat karakteristiknya yaitu klaster dari algoritma DHCC dengan optimisasi karena pada sub bab sebelumnya dijelaskan bahwa hasil klaster yang dihasilkan lebih baik dengan adanya optimisasi.

Karakteristik dilihat dari semua variabel kategorik yang terdapat pada data. Variabel P1, P2, P3 dilihat dari proporsi kemunculan nilai kategorik. Variabel lainnya yang mengindikasikan rasa kepemilikan secara psikologis, sikap terhadap kegiatan sukarela, intensi berkegiatan sukarela, dan tekanan waktu dilihat dari rata-rata analisis interval tiap variabel yang membentuknya. Berikut merupakan karakteristik dari klaster yang dihasilkan dengan algoritma DHCC dengan optimisasi :

1. Klaster 1.

Klaster 1 memiliki anggota sebanyak 38 individu atau 70,37% dari semua dari total semua individu. Individu pada klaster ini sebagian besar merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama (73,68% menjawab ya). Kemudian, individu pada klaster ini sebagian besar merasa nyaman dalam keseharian bersama anggota lainnya (94,74% menjawab ya). Jenis kegiatan yang diminati oleh sebagian besar individu pada klaster ini adalah kegiatan petualangan dan sebagian kecil lainnya berminat pada gabungan kegiatan petualangan dan non petualangan (71,05% menjawab petualangan). Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf biasa. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung baik. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf biasa yang cenderung tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf biasa.

2. Klaster 2.

   Klaster 2 memiliki anggota sebanyak 4 individu atau 7,41% dari semua dari total semua individu. Individu pada klaster ini semuanya merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini semuanya merasa nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh setengah individu pada klaster ini adalah kegiatan petualangan dan setengah lainnya berminat pada kegiatan non petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf biasa yang cenderung rendah. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung buruk. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf biasa yang cenderung tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf biasa yang cenderung tinggi.

3. Klaster 3.

Klaster 3 memiliki anggota sebanyak 2 individu atau 3,7% dari semua dari total semua individu. Individu pada klaster ini semuanya merasa tidak puas karena ekspektasi mereka tidak terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini setengahnya merasa nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh individu pada klaster ini adalah kegiatan petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf rendah. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung buruk. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf rendah. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf tinggi.

4. Klaster 4.

   Klaster 4 memiliki anggota sebanyak 2 individu atau 3,7% dari semua dari total semua individu. Individu pada klaster ini semuanya merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini semuanya merasa nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh individu pada klaster ini adalah kegiatan petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf biasa. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung baik. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf rendah. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf biasa yang cenderung tinggi.

5. Klaster 5.

   Klaster 5 memiliki anggota sebanyak 2 individu atau 3,7% dari semua dari total semua individu. Individu pada klaster ini semuanya merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini setengahnya merasa nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh individu pada klaster ini adalah gabungan kegiatan petualangan dan non petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf rendah yang cenderung sangat rendah. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung baik. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf tinggi yang cenderung sangat tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf biasa yang cenderung tinggi.

6. Klaster 6.

   Klaster 6 memiliki anggota sebanyak 1 individu atau 1,85% dari semua dari total semua individu. Individu pada klaster ini merasa tidak puas karena ekspektasi mereka tidak terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini merasa tidak nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh individu pada klaster ini adalah kegiatan petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf biasa yang cenderung rendah. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf tinggi cenderung sangat tinggi. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf rendah.

7. Klaster 7.

   Klaster 7 memiliki anggota sebanyak 3 individu atau 5,56% dari semua dari total semua individu. Individu pada klaster ini semuanya merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini sebagian besar merasa nyaman dalam keseharian bersama anggota lainnya (66,67% menjawab ya). Jenis kegiatan yang diminati oleh individu pada klaster ini adalah kegiatan petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf tinggi. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf sangat baik. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf tinggi yang cenderung sangat tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf tinggi.

8. Klaster 8.

   Klaster 8 memiliki anggota sebanyak 2 individu atau 3,7% dari semua dari total semua individu. Individu pada klaster ini setengahnya merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama. Kemudian, individu pada klaster ini semuanya merasa nyaman dalam keseharian bersama anggota lainnya. Jenis kegiatan yang diminati oleh setengah individu pada klaster ini adalah kegiatan petualangan dan setengah lainnya berminat pada kegiatan non petualangan. Rasa memiliki secara psikologis dari individu terhadap organisasi pada klaster ini berada pada taraf tinggi. Individu pada klaster ini menyikapi kegiatan sukarela pada taraf sangat baik. Intensi untuk berkegiatan sukarela dari individu pada klaster ini berada pada taraf sangat tinggi. Tekanan waktu yang dirasakan individu pada klaster ini berada pada taraf biasa yang cenderung rendah.

# BAB V
# PENUTUP

## 5. 1 Kesimpulan

Dari pembahasan pada bab-bab sebelumnya, maka didapatkan kesimpulan sebagai berikut :

1. Terdapat 8 segmen anggota UKM Mapagama dengan mengaplikasikan algoritma DHCC.
2. Terdapat suatu segmen yang dominan karena mencakup 70,37% dari keseluruhan anggota Mapagama. Karakteristik individu pada segmen ini sebagian besar merasa puas karena ekspektasi mereka terpenuhi setelah bergabung dengan UKM Mapagama (73,68% menjawab ya). Kemudian, individu pada segmen ini sebagian besar merasa nyaman dalam keseharian bersama anggota lainnya (94,74% menjawab ya). Jenis kegiatan yang diminati oleh sebagian besar individu pada segmen ini adalah kegiatan petualangan dan sebagian kecil lainnya berminat pada gabungan kegiatan petualangan dan non petualangan (71,05% menjawab petualangan). Rasa memiliki secara psikologis dari individu terhadap organisasi pada segmen ini berada pada taraf biasa. Individu pada segmen ini menyikapi kegiatan sukarela pada taraf biasa yang cenderung baik. Intensi untuk berkegiatan sukarela dari individu pada segmen ini berada pada taraf biasa yang cenderung tinggi. Tekanan waktu yang dirasakan individu pada segmen ini berada pada taraf biasa
3. Dari studi kasus yang dilakukan yaitu segmentasi anggota UKM Mapagama, optimisasi pada algoritma DHCC terbukti meningkatkan performa dalam analisis klaster karena meningkatkan nilai indeks utilitas kategori ($CU$).

## 5. 2 Saran

Penelitian ini mungkin saja terdapat beberapa kekurangan, sehingga berikut saran yang diberikan peneliti :

1. Algoritma analisis klaster pada tugas akhir ini hanya dapat digunakan untuk data kategorik, sehingga mungkin untuk penelitian selanjutnya dapat

dikembangkan algoritma analisis klaster yang dapat digunakan untuk data numerik dan kategorik.

2. Pada tugas akhir ini, tidak dilakukan validasi dari variabel kategorik yang menjadi indikator dari suatu variabel laten atau konstruk, seperti tekanan waktu yang diukur dengan dua pertanyaan karena fokus pada tugas akhir ini berada pada pengaplikasian algoritma DHCC. Seharusnya hasil dari analisis klaster akan lebih baik jika dilakukan validasi terlebih dahulu.
3. Karakteristik dari suatu klaster sangat subjektif dalam penelitian ini, terutama pada variabel yang diukur dengan skala likert. Untuk kedepannya perlu dikembangkan untuk melihat karakteristik dari skala likert selain analisis interval.

# DAFTAR PUSTAKA

Anton, H. & Rorres, C., 2004. *Elementary Linear Algebra 9th Edition.* USA: John Wiley & Sons.

Bain, L. & Engelhardt, M., 1992. *Introduction to Probability and Mathematical Statistics Second Edition.* Pacific Grove: Brooks/Cole.

Baker, K., 2005. [Online] Available at: http://site.iugaza.edu.ps/ahdrouss/files/2010/03/Singular_Value_Decomposition Tutorial.pdf [Diakses 10 April 2021].

Barbara, D., Li, Y. & Cuoto, J., 2002. COOLCAT : an entropy-based algorithm for categorical clustering. *In : Proceedings of the 11th ACM international conference on information and knowledge management, McLean, VA, USA,* pp. 582-589.

Caldwell, L. & Andereck, K., 1994. Motives for Initiating and Continuing Membership in a Recreation-related Voluntary Association. *Leis. Science,* Volume 16(1), pp. 33-44.

Clary, E. & Snyder, M., 1999. The Motivations to Volunteer : Theoretical and Practical Considerations. *Curr.Dir. Psychology and Sociology 8,* Volume 5, pp. 156-159.

Clary, E. et al., 1998. Understanding and Assessing the Motivations of Volunteers : a Functional Approach. *Journal Personal Sociology and Psychology,* Volume 74, pp. 1516-1530.

Cornelis, I., van Hiel, A. & De Cremer, D., 2013. Volunteer Work in Youth Organization : Predicting Distinct Aspects of Volunteering Behavior from Self and Other Oriented Motives. *Journal Application Sociology and Psychology 43,* Volume 2, pp. 456-466.

Do, H. & Kim, J., 2008. *Categorical Data Clustering Using The Combinations of Attribute Values.* New York: Springer.

Goldberg, J. L., 1991. *Matrix Theory with Applications.* United States of America: McGraw-Hill Inc.

Greenacre, M. & Blasius, J., 2006. *Multiple Correspondence Analysis and Related Methods.* London: Chapman & Hall.

Guha, S., Rastogi, R. & Shim, K., 1999. ROCK : a robust clustering algorithm for categorical attributes. *In : Proceedings of the 15th IEEE international conference on data engineering, Sydney, Australia,* pp. 512-521.

Han, J., Kamber, M. & Pei, J., 2011. *Data Mining : Concepts and Techniques 3rd Edition.* Amsterdam: Elsevier.

Johnson, R. & Wichern, D., 2018. *Applied Multivariate Statistical Analysis 6th Edition.* Upper Saddle River: Prentince Hall.

Omoto, A. & Snyder, M., 1995. Sustained Helping Without Obligation : Motivation, Longevity of Service, and Perceived Attitude Change Amongs AIDS Volunteers. *Journal Personal Sociology and Psychology,* Volume 68, pp. 671-686.

Sugiyono, P., 2011. *Metode Penelitian Kombinasi.* Yogyakarta: Alfabeta.

Tan, P., Steinbach, M. & Kumar, V., 2005. *Introduction to Data Mining.* Reading: Addison-Wesley.

Wang, P., 2004. Assessing Motivation for Sports Volunteerism. *North American Advances in Consumer Research,* Volume 31, pp. 420-425.

Wilson, J., 2000. Volunteering. *Annual Review of Sociology 26,* Volume 1, pp. 215-240.

Wu, J., Xiong, H. & Chen, J., 2009. Adapting The Right Measures for K-Means Clustering. *In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, Paris, France,* pp. 877-885.

Xiong, T., Wang, S., Mayers, A. & Monga, E., 2009. A New MCA-based Divisive Hierarchical algorithm for Clustering Categorical Data. *In : Proceedings of the 9th IEEE international conference on data mining, Miami, FL, USA,* pp. 1058-1063.

Xiong, T., Wang, S., Mayers, A. & Monga, E., 2012. DHCC : Divisive Hierarchical Clustering of Categorical Data. *Data Mining and Knowledge Discovery,* 24(1), pp. 103-135.

Zhao, Y. & Karypis, G., 2005. Hierarchical Clustering Algorithms for Document Dataset. *Data Mining and Knowledge Discovery,* Volume 10, pp. 141-168.

# LAMPIRAN

**Lampiran 1. Data karakteristik anggota UKM Mapagama periode 2020/2021**

| **Nama** | **X1** | **X2** | **X3** | **X4** | **X5** | **X6** | **X7** | **X8** | **X9** | **X10** | **X11** | **X12** | **X13** | **X14** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Agus | 1 | 2 | 1 | 3 | 4 | 3 | 2 | 5 | 4 | 3 | 3 | 3 | 3 | 4 |
| Uci | 2 | 1 | 3 | 3 | 5 | 3 | 3 | 3 | 3 | 3 | 5 | 3 | 2 | 3 |
| Rima | 2 | 1 | 3 | 3 | 3 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Hanif | 1 | 1 | 1 | 3 | 3 | 2 | 3 | 4 | 4 | 5 | 4 | 4 | 3 | 3 |
| Sonya | 1 | 1 | 1 | 3 | 4 | 3 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 3 |
| Malik | 1 | 1 | 1 | 4 | 2 | 2 | 3 | 4 | 4 | 4 | 3 | 3 | 2 | 1 |
| Ayyas | 2 | 1 | 1 | 3 | 3 | 2 | 3 | 4 | 4 | 3 | 3 | 2 | 4 | 3 |
| Eni | 1 | 1 | 3 | 4 | 5 | 3 | 4 | 4 | 3 | 3 | 4 | 3 | 4 | 4 |
| M Ilham | 1 | 1 | 1 | 3 | 4 | 3 | 4 | 3 | 3 | 2 | 3 | 3 | 5 | 4 |
| Burhan | 2 | 1 | 1 | 2 | 3 | 2 | 2 | 4 | 2 | 2 | 2 | 2 | 3 | 4 |
| Dian | 1 | 2 | 3 | 1 | 1 | 1 | 3 | 4 | 2 | 5 | 5 | 5 | 3 | 4 |
| Suryo | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 |
| Alfira | 1 | 1 | 3 | 2 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Adam | 2 | 1 | 2 | 2 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 2 | 2 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tata | 1 | 1 | 3 | 3 | 4 | 3 | 3 | 4 | 3 | 4 | 4 | 4 | 4 | 4 |
| Sita | 1 | 2 | 3 | 4 | 4 | 3 | 3 | 4 | 3 | 4 | 4 | 3 | 3 | 3 |
| Evan | 1 | 1 | 1 | 2 | 2 | 2 | 3 | 2 | 2 | 3 | 4 | 2 | 4 | 3 |
| Unggul | 1 | 1 | 1 | 2 | 3 | 1 | 2 | 3 | 2 | 4 | 4 | 4 | 3 | 3 |
| Bene | 2 | 2 | 1 | 1 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 4 | 5 |
| Abyan | 1 | 1 | 2 | 2 | 2 | 2 | 4 | 3 | 2 | 3 | 5 | 5 | 3 | 2 |
| Hendri | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Tokici | 2 | 1 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | 2 | 3 | 5 | 3 | 5 |
| Ai | 1 | 1 | 3 | 3 | 4 | 3 | 3 | 3 | 2 | 3 | 2 | 3 | 2 | 1 |
| Ichsan | 1 | 1 | 1 | 4 | 4 | 2 | 1 | 3 | 3 | 2 | 2 | 1 | 4 | 4 |
| Nugi | 1 | 2 | 1 | 4 | 5 | 4 | 2 | 4 | 5 | 4 | 5 | 3 | 3 | 4 |
| Soleh | 2 | 1 | 1 | 4 | 4 | 3 | 3 | 2 | 1 | 4 | 3 | 4 | 4 | 5 |
| Pandu | 1 | 1 | 1 | 3 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 2 | 3 |
| Ajun | 1 | 1 | 1 | 3 | 3 | 4 | 2 | 3 | 4 | 4 | 2 | 2 | 2 | 4 |
| Tyas | 1 | 1 | 1 | 3 | 3 | 3 | 2 | 4 | 3 | 4 | 4 | 4 | 3 | 2 |
| Hanggara | 1 | 1 | 1 | 4 | 4 | 1 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 |
| Joanna | 2 | 2 | 1 | 1 | 4 | 2 | 2 | 5 | 4 | 4 | 4 | 4 | 2 | 2 |
| Heppy | 1 | 1 | 1 | 3 | 3 | 3 | 4 | 2 | 2 | 3 | 3 | 3 | 2 | 2 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dhona | 1 | 1 | 3 | 1 | 2 | 1 | 3 | 4 | 4 | 4 | 4 | 5 | 5 | 2 |
| Toto | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 2 | 2 |
| James | 1 | 1 | 1 | 2 | 3 | 3 | 4 | 4 | 3 | 4 | 4 | 4 | 3 | 4 |
| Udin | 1 | 1 | 1 | 4 | 3 | 3 | 3 | 5 | 4 | 4 | 3 | 3 | 3 | 2 |
| Fatur | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 3 | 3 |
| Kresna | 1 | 1 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 4 | 5 |
| Septi | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 4 | 3 | 3 | 3 | 4 | 3 | 2 |
| Lyan | 1 | 1 | 1 | 3 | 2 | 3 | 3 | 4 | 3 | 3 | 4 | 4 | 5 | 4 |
| Hambali | 1 | 1 | 1 | 3 | 1 | 2 | 3 | 5 | 3 | 3 | 3 | 3 | 3 | 3 |
| Bayu | 1 | 1 | 1 | 4 | 3 | 4 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 4 |
| Intan | 2 | 1 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 2 |
| Angga | 2 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 1 | 4 | 5 | 3 | 2 | 3 |
| Arya | 1 | 1 | 1 | 2 | 1 | 2 | 3 | 4 | 3 | 3 | 4 | 4 | 3 | 3 |
| Harits | 1 | 1 | 3 | 4 | 3 | 2 | 4 | 4 | 3 | 4 | 4 | 4 | 3 | 3 |
| Lafif | 1 | 1 | 3 | 3 | 3 | 2 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 2 |
| Erdyan | 1 | 1 | 1 | 3 | 3 | 3 | 4 | 4 | 3 | 4 | 4 | 3 | 3 | 3 |
| Aulia | 2 | 1 | 1 | 2 | 3 | 2 | 4 | 3 | 3 | 4 | 3 | 3 | 2 | 2 |
| Hafidz | 1 | 1 | 1 | 4 | 3 | 2 | 4 | 4 | 4 | 4 | 3 | 4 | 3 | 3 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Faiq | 1 | 1 | 1 | 3 | 4 | 3 | 4 | 4 | 2 | 4 | 3 | 3 | 4 | 5 |
| Dimi | 2 | 1 | 1 | 2 | 3 | 3 | 2 | 3 | 3 | 3 | 4 | 3 | 4 | 4 |
| Byan | 1 | 1 | 1 | 3 | 3 | 3 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 |
| Hendra | 1 | 1 | 1 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 |

| | |
|---|---|
| X1 | Apakah ekspektasi Anda tentang Mapagama sesuai dengan realita yang Anda temui setelah menjadi Anggota Mapagama? |
| X2 | Apakah Anda nyaman dalam keseharian bersama anggota Mapagama baik di sekretariat ataupun tempat lainnya? |
| X3 | Jenis kegiatan apa yang Anda minati di Mapagama? |
| X4 | Saya merasa memiliki kontrol atas bagaimana Mapagama bekerja. |
| X5 | Saya merasa memberikan sesuatu untuk Mapagama. |
| X6 | Saya merasa sebagai penyokong Mapagama. |
| X7 | Saya pikir Mapagama banyak merepresentasikan prinsip/nilai saya |
| X8 | Saya senang berkegiatan sukarela di Mapagama. |
| X9 | Berkegiatan sukarela di Mapagama merupakan hal yang memuaskan tidak peduli berapapun waktunya. |
| X10 | Sangat mungkin saya berkegiatan sukarela di Mapagama. |

| | |
|---|---|
| X11 | Saya pikir bagus bila saya berkegiatan sukarela di Mapagama. |
| X12 | Bila diberi kesempatan saya akan berkegiatan sukarela di Mapagama. |
| X13 | Saya tidak memiliki cukup waktu untuk berkegiatan sukarela. |
| X14 | Berkegiatan sukarela menambah beban waktu saya. |

**Lampiran 2.** ***Syntax*** **R algoritma DHCC**

a. Fungsi pemisahan klaster

```
library(readxl)
library(tidyverse)
library(dplyr)
library(tibble)
library(Matrix)
prebi=function(m,dataZ){
  HK=list()
  n=dim(dataZ)[1]
  Z=as.matrix(as.data.frame(select(dataZ,-Nama),nrow=n))
  J=dim(Z)[2]
  P=Z/(n*m)
  r=matrix((1/n),nrow=n)
  Dr=diag(1,n)/n
  c=matrix(diag(t(Z)%*%Z)/(n*m),nrow=J)
  Dc=as.matrix(Diagonal(J,c))
  r1=1/sqrt(r)
  c1=1/sqrt(c)
  Dr1=as.matrix(Diagonal(n,r1))
  Dc1=as.matrix(Diagonal(J,c1))
  S=Dr1%*%(P-r%*%t(c))%*%Dc1
  S.svd=svd(S)

  Sig=as.matrix(Diagonal(n,S.svd$d))
  f=Dr1%*%S.svd$u%*%Sig
  dataf=dataZ %>% add_column(KUB=f[,1])
  daun0=select(subset(dataf,KUB<=0),-KUB)
  daun1=select(subset(dataf,KUB>0),-KUB)
  HK=list(dataZ,daun0,daun1)
  return(HK)
```

```
  }

clean=function(CP){
  CP=CP                                                    %>%
select_if(negate(function(col)is.numeric(col)&&sum(col)==0))
  return(CP)}
```

b. Fungsi optimisasi pemisahan klaster

```
refinement=function(Cp,Clp,Crp){
    repeat{
      olp=as.matrix(Clp %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))

      orp=as.matrix(Crp %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))

      nl=dim(Clp)[1]
      nr=dim(Crp)[1]
      Crpnew=tibble()
      Clpnew=tibble()
      for(i in 1 :nl){
        zi=as.matrix(Clp[i,-1])
        orp=orp+zi
        nr=nr+1
        centerl=sqrt(olp/nl)
        centerr=sqrt(orp/nr)
        chirp=(zi-centerr)^2/centerr
        chilp=(zi-centerl)^2/centerl
        chirp[is.nan(chirp)]=0
        chilp[is.nan(chilp)]=0
        schirp=sum(chirp)
        schilp=sum(chilp)
```

```
    if(schirp<schilp){
    Crpnew=rbind(Crpnew,Clp[i,])
      } else {
    Clpnew=rbind(Clpnew,Clp[i,])
    }
  orp=orp-zi
  nr=nr-1
}

for(i in 1 :nr){
  zi=as.matrix(Crp[i,-1])
  olp=olp+zi
  nl=nl+1
  centerl=sqrt(olp/nl)
  centerr=sqrt(orp/nr)
  chirp=(zi-centerr)^2/centerr
  chilp=(zi-centerl)^2/centerl
  chirp[is.nan(chirp)]=0
  chilp[is.nan(chilp)]=0
  schirp=sum(chirp)
  schilp=sum(chilp)

  if(schilp<schirp){
    Clpnew=rbind(Clpnew,Crp[i,])
  } else {
    Crpnew=rbind(Crpnew,Crp[i,])
  }
  olp=olp-zi
  nl=nl-1
}
```

```
    Clp=Clpnew
    Crp=Crpnew
  if(nr==dim(Crp)[1]|nl==dim(Clp)[1]|dim(Crp)[1]==0|dim(Clp)[1]==0){
    break
    }
  }
  HK=list(Cp,Clp,Crp)
  return(HK)
}
```

c. Fungsi kualitas pemisahan klaster

```
termination=function(DATA,CP,CLP,CRP){
  PCT=select(DATA,colnames(CP))
  PLT=select(DATA,colnames(CLP))
  PRT=select(DATA,colnames(CRP))
  ZPCT=as.matrix(PCT %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(DATA)[1]
  ZPLT=as.matrix(PLT %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(DATA)[1]
  ZPRT=as.matrix(PRT %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(DATA)[1]
  PC=as.matrix(CP %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(CP)[1]
  PL=as.matrix(CLP %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(CLP)[1]
  PR=as.matrix(CRP %>%
        summarize_if(is.numeric, sum, na.rm=TRUE))/dim(CRP)[1]
  pQC=PC*log(PC)-ZPCT*log(ZPCT)
  pQL=PL*log(PL)-ZPLT*log(ZPLT)
  pQR=PR*log(PR)-ZPRT*log(ZPRT)
  pQC[is.nan(pQC)]=0
  pQL[is.nan(pQL)]=0
```

```
  pQR[is.nan(pQR)]=0
  proQC=sum(pQC)
  proQL=sum(pQL)
  proQR=sum(pQR)
  QC=(dim(CP)[1]/dim(DATA)[1])*proQC
  QL=(dim(CLP)[1]/dim(DATA)[1])*proQL
  QR=(dim(CRP)[1]/dim(DATA)[1])*proQR
  QCnew=dim(CLP)[1]/dim(DATA)[1]*QL+dim(CRP)[1]/dim(DATA)[1]
  *QR
  Q=list(QC,QCnew)
  return(Q)
}
```

d. Fungsi indeks utilitas kategori

```
UTILITY=function(DATA,CP){
  PC=as.matrix(CP %>%
          summarize_if(is.numeric, sum, na.rm=TRUE))/dim(CP)[1]
  PCT=select(DATA,colnames(CP))
  ZPCT=as.matrix(PCT %>%
          summarize_if(is.numeric, sum, na.rm=TRUE))/dim(DATA)[1]
  PCsq=PC^2
  PTsq=ZPCT^2
  pU=PCsq-PTsq
  pU[is.nan(pU)]=0
  U=sum(pU)
  return(U)
}
```

e. *Syntax* aplikasi algoritma DHCC tanpa optmisasi

```
#Step1
C=prebi(14,dataZ)
C0=clean(C[[2]])
```

```
C1=clean(C[[3]])
QC=termination(dataZ,C[[1]],C0,C1)
#Step2

C00=clean(prebi(14,C0)[[2]])
C01=clean(prebi(14,C0)[[3]])
QC0=termination(dataZ,C0,C00,C01)

C10=clean(prebi(14,C1)[[2]])
C11=clean(prebi(14,C1)[[3]])
QC1=termination(dataZ,C1,C10,C11)
#Step3
C100=clean(prebi(14,C10)[[2]])
C101=clean(prebi(14,C10)[[3]])
QC10=termination(dataZ,C10,C100,C101)

C110=clean(prebi(14,C11)[[2]])
C111=clean(prebi(14,C11)[[3]])
QC11=termination(dataZ,C11,C110,C111)

#Step 4
C1100=clean(prebi(14,C110)[[2]])
C1101=clean(prebi(14,C110)[[3]])
QC110=termination(dataZ,C110,C1100,C1101)

C1110=clean(prebi(14,C111)[[2]])
C1111=clean(prebi(14,C111)[[3]])
QC111=termination(dataZ,C111,C1110,C1111)

#Step 5
C11010=clean(prebi(14,C1101)[[2]])
```

```
C11011=clean(prebi(14,C1101)[[3]])
QC1101=termination(dataZ,C1101,C11010,C11011)



UC0=UTILITY(dataZ,C0)
UC10=UTILITY(dataZ,C10)
UC111=UTILITY(dataZ,C111)
UC1100=UTILITY(dataZ,C1100)
UC1101=UTILITY(dataZ,C1101)
UC=(UC0*dim(C0)[1]/dim(dataZ)[1])+(UC10*dim(C10)[1]/dim(dataZ)[1
])+

(UC111*dim(C111)[1]/dim(dataZ)[1])+(UC1100*dim(C1100)[1]/dim(dat
aZ)[1])+
  (UC1101*dim(C1101)[1]/dim(dataZ)[1])




library(sqldf)
data<- read_excel("D:/UGM/Kuliah/Term 8/Tugas Akhir/Data/Survei DPA
(Responses).xlsx")
C0f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
        FROM C0
        LEFT JOIN data AS d
        USING (Nama)")
```

```
C10f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C10
    LEFT JOIN data AS d
    USING (Nama)")
C111f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C111
    LEFT JOIN data AS d
    USING (Nama)")
C1100f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1100
    LEFT JOIN data AS d
    USING (Nama)")
C1101f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1101
    LEFT JOIN data AS d
    USING (Nama)")
DHCC=list(C0f,C10f,C111f,C1100f,C1101f)
library(writexl)
write_xlsx(DHCC,"D:/UGM/Kuliah/Term 8/Tugas Akhir/Data/DHCC
tanpa optimisasi.xlsx")
```

f. *Syntax* aplikasi algoritma DHCC dengan optimisasi

```
#Step 1
C0=prebi(14,dataZ)[[2]]
```

```
C1=prebi(14,dataZ)[[3]]
C0r=clean(refinement(dataZ,C0,C1)[[2]])
C1r=clean(refinement(dataZ,C0,C1)[[3]])
QC=termination(dataZ,dataZ,C0r,C1r)
#Step 2
C00=prebi(14,C0r)[[2]]
C01=prebi(14,C0r)[[3]]
C00r=clean(refinement(C0r,C00,C01)[[2]])
C01r=clean(refinement(C0r,C00,C01)[[3]])
QC0=termination(dataZ,C0r,C00r,C01r)

C10=prebi(14,C1r)[[2]]
C11=prebi(14,C1r)[[3]]
C10r=clean(refinement(C1r,C10,C11)[[2]])
C11r=clean(refinement(C1r,C10,C11)[[3]])
QC1=termination(dataZ,C1r,C10r,C11r)

#Step3
C100=prebi(14,C10r)[[2]]
C101=prebi(14,C10r)[[3]]
C100r=clean(refinement(C10r,C100,C101)[[2]])
C101r=clean(refinement(C10r,C100,C101)[[3]])
QC10=termination(dataZ,C10r,C100r,C101r)

C110=prebi(14,C11r)[[2]]
C111=prebi(14,C11r)[[3]]
C110r=clean(refinement(C11r,C110,C111)[[2]])
C111r=clean(refinement(C11r,C110,C111)[[3]])
QC11=termination(dataZ,C11r,C110r,C111r)
#Step 4
C1000=prebi(14,C100r)[[2]]
```

```
C1001=prebi(14,C100r)[[3]]
C1000r=clean(refinement(C100r,C1000,C1001)[[2]])
C1001r=clean(refinement(C100r,C1000,C1001)[[3]])
QC100=termination(dataZ,C100r,C1000r,C1001r)

C1010=prebi(14,C101r)[[2]]
C1011=prebi(14,C101r)[[3]]
C1010r=clean(refinement(C101r,C1010,C1011)[[2]])
C1011r=clean(refinement(C101r,C1010,C1011)[[3]])
QC101=termination(dataZ,C101r,C1010r,C1011r)

C1100=prebi(14,C110r)[[2]]
C1101=prebi(14,C110r)[[3]]
C1100r=clean(refinement(C110r,C1100,C1101)[[2]])
C1101r=clean(refinement(C110r,C1100,C1101)[[3]])
QC110=termination(dataZ,C110r,C1100r,C1101r)

C1110=prebi(14,C111r)[[2]]
C1111=prebi(14,C111r)[[3]]
C1110r=clean(refinement(C111r,C1110,C1111)[[2]])
C1111r=clean(refinement(C111r,C1110,C1111)[[3]])
QC111=termination(dataZ,C111r,C1110r,C1111r)

#Step 5
C10100=prebi(14,C1010r)[[2]]
C10101=prebi(14,C1010r)[[3]]
C10100r=clean(refinement(C1010r,C10100,C10101)[[2]])
C10101r=clean(refinement(C1010r,C10100,C10101)[[3]])
QC1010=termination(dataZ,C1010r,C10100r,C10101r)

C10110=prebi(14,C1011r)[[2]]
```

```
C10111=prebi(14,C1011r)[[3]]
C10110r=clean(refinement(C1011r,C10110,C10111)[[2]])
C10111r=clean(refinement(C1011r,C10110,C10111)[[3]])
QC1011=termination(dataZ,C1011r,C10110r,C10111r)

C11000=prebi(14,C1100r)[[2]]
C11001=prebi(14,C1100r)[[3]]
C11000r=clean(refinement(C1100r,C11000,C11001)[[2]])
C11001r=clean(refinement(C1100r,C11000,C11001)[[3]])
QC1100=termination(dataZ,C1100r,C11000r,C11001r)

C11100=prebi(14,C1110r)[[2]]
C11101=prebi(14,C1110r)[[3]]
C11100r=clean(refinement(C1110r,C11100,C11101)[[2]])
C11101r=clean(refinement(C1110r,C11100,C11101)[[3]])
QC1110=termination(dataZ,C1110r,C11100r,C11101r)

C11110=prebi(14,C1111r)[[2]]
C11111=prebi(14,C1111r)[[3]]
C11110r=clean(refinement(C1111r,C11110,C11111)[[2]])
C11111r=clean(refinement(C1111r,C11110,C11111)[[3]])
QC1111=termination(dataZ,C1111r,C11110r,C11111r)


UC0=UTILITY(dataZ,C0r)
UC10=UTILITY(dataZ,C10r)
UC111=UTILITY(dataZ,C111r)
UC1100=UTILITY(dataZ,C1100r)
UC1101=UTILITY(dataZ,C1101r)
UC=(UC0*dim(C0)[1]/dim(dataZ)[1])+(UC10*dim(C10)[1]/dim(dataZ)[1
])+
```

```
(UC111*dim(C111)[1]/dim(dataZ)[1])+(UC1100*dim(C1100)[1]/dim(dat
aZ)[1])+
  (UC1101*dim(C1101)[1]/dim(dataZ)[1])

library(sqldf)
data<- read_excel("D:/UGM/Kuliah/Term 8/Tugas Akhir/Data/Survei DPA
(Responses).xlsx")
C0f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
      FROM C0r
      LEFT JOIN data AS d
      USING (Nama)")
C100f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
      FROM C100r
      LEFT JOIN data AS d
      USING (Nama)")
C1010f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
      FROM C1010r
      LEFT JOIN data AS d
      USING (Nama)")
C1011f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
      FROM C1011r
      LEFT JOIN data AS d
```

```
    USING (Nama)")
C1100f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1100r
    LEFT JOIN data AS d
    USING (Nama)")
C1101f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1101r
    LEFT JOIN data AS d
    USING (Nama)")
C1110f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1110r
    LEFT JOIN data AS d
    USING (Nama)")
C1111f=sqldf("SELECT
d.NAMA,d.X1,d.X2,d.X3,d.X4,d.X5,d.X6,d.X7,d.X8,d.X9,d.X10,d.X11,d
.X12,d.X13,d.X14
    FROM C1111r
    LEFT JOIN data AS d
    USING (Nama)")
DHCCop=list(C0f,C100f,C1010f,C1011f,C1100f,C1101f,C1110f,C1111f)
library(writexl)
write_xlsx(DHCCop,"D:/UGM/Kuliah/Term 8/Tugas Akhir/Data/DHCC
dengan optimisasi.xlsx")
```

**Lampiran 3. *Output* hasil analisis**

Algoritma DHCC tanpa optimisasi

Iterasi 1

```
> C0
# A tibble: 41 x 55
   Nama      X2    X3    X4    X5    X6    X7    X8    X9   X10   X11   X12   X13
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Rima       0     1     1     0     0     0     1     0     0     1     0     0
 2 Hanif      1     0     1     0     1     0     0     0     0     1     0     0
 3 Sonya      1     0     1     0     1     0     0     0     0     1     0     0
 4 Malik      1     0     1     0     1     0     0     0     0     0     1     0
 5 Ayyas      0     1     1     0     1     0     0     0     0     1     0     0
 6 Eni        1     0     1     0     0     0     1     0     0     0     1     0
 7 M Ilham    1     0     1     0     1     0     0     0     0     1     0     0
 8 Burhan     0     1     1     0     1     0     0     0     1     0     0     0
```

```
> C1
# A tibble: 13 x 51
   Nama      X2    X3    X4    X5    X6    X7    X8    X9   X10   X11   X12   X13
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Agus       1     0     0     1     1     0     0     0     0     1     0     0
 2 Uci        0     1     1     0     0     0     1     0     0     1     0     0
 3 Dian       1     0     0     1     0     0     1     1     0     0     0     1
 4 Adam       0     1     1     0     0     1     0     0     1     0     0     0
 5 Abyan      1     0     1     0     0     1     0     0     1     0     0     0
 6 Nugi       1     0     0     1     1     0     0     0     0     0     1     0
 7 Pandu      1     0     1     0     1     0     0     0     0     1     0     0
```

```
> QC
[[1]]
[1] 0

[[2]]
[1] 0.6799502
```

Iterasi 2

```
# A tibble: 25 x 43
   Nama      X2    X3    X4    X5    X6    X8   X10   X11   X12   X13   X14   X15
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Rima       0     1     1     0     0     1     0     1     0     0     0     1
 2 Hanif      1     0     1     0     1     0     0     1     0     0     0     1
 3 Sonya      1     0     1     0     1     0     0     1     0     0     0     0
 4 Malik      1     0     1     0     1     0     0     0     1     0     1     0
 5 Eni        1     0     1     0     0     1     0     0     1     0     0     0
 6 Suryo      1     0     1     0     1     0     0     1     0     0     0     1
 7 Alfira     1     0     1     0     0     1     1     0     0     0     0     1
 8 Tata       1     0     1     0     0     1     0     1     0     0     0     0
```

```
> C01
# A tibble: 16 x 50
   Nama      X2    X3    X4    X5    X6    X7    X8    X9   X10   X11   X12   X13
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Ayyas      0     1     1     0     1     0     0     0     0     1     0     0
 2 M Ilham    1     0     1     0     1     0     0     0     0     1     0     0
 3 Burhan     0     1     1     0     1     0     0     0     1     0     0     0
 4 Evan       1     0     1     0     1     0     0     0     1     0     0     0
 5 Unggul     1     0     1     0     1     0     0     0     1     0     0     0
 6 Bene       0     1     0     1     1     0     0     1     0     0     0     0
 7 Hendri     1     0     1     0     1     0     0     1     0     0     0     1
```

```
> QC0
[[1]]
[1] 0.9555364

[[2]]
[1] 0.3434087
```

```
> C10
# A tibble: 8 x 42
  Nama     X2    X3    X4    X5    X6    X9   X11   X12   X15   X16   X17   X18   X19
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Agus      1     0     0     1     1     0     1     0     0     1     0     0     0
2 Nugi      1     0     0     1     1     0     0     1     0     0     1     0     0
3 Pandu     1     0     1     0     1     0     1     0     0     1     0     0     0
4 Ajun      1     0     1     0     1     0     1     0     1     0     0     0     0
5 Hang~     1     0     1     0     1     0     0     1     0     1     0     1     0
6 Joan~     0     1     0     1     1     1     0     0     0     1     0     0     1
7 Bayu      1     0     1     0     1     0     0     1     1     0     0     0     0
8 Hend~     1     0     1     0     1     0     0     1     1     0     0     0     0
#     with 28 more variables: X20 <dbl>, X21 <dbl>, X24 <dbl>, X26 <dbl>, X27 <dbl>,
```

```
> C11
# A tibble: 5 x 39
  Nama     X2    X3    X4    X5    X7    X8    X9   X10   X11   X13   X14   X17   X18
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Uci       0     1     1     0     0     1     0     0     1     0     0     1     0
2 Dian      1     0     0     1     0     1     1     0     0     1     0     0     1
3 Adam      0     1     1     0     1     0     0     1     0     0     0     1     0
4 Abyan     1     0     1     0     1     0     0     1     0     0     1     0     0
5 Dhona     1     0     1     0     0     1     1     0     0     0     1     0     1
# ... with 25 more variables: X19 <dbl>, X20 <dbl>, X22 <dbl>, X25 <dbl>, X26 <dbl>,
#   X29 <dbl>, X30 <dbl>, X31 <dbl>, X33 <dbl>, X34 <dbl>, X35 <dbl>, X36 <dbl>,
#   X38 <dbl>, X39 <dbl>, X40 <dbl>, X43 <dbl>, X44 <dbl>, X47 <dbl>, X49 <dbl>,
```

```
> QC1
[[1]]
[1] -0.1892065

[[2]]
[1] -0.007446221
```

Iterasi 3

```
> C100
# A tibble: 3 x 29
  Nama     X2    X3    X4    X5    X6    X9   X11   X15   X16   X19   X20   X21   X24
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Agus      1     0     0     1     1     0     1     0     1     0     1     0     1
2 Ajun      1     0     1     0     1     0     1     1     0     0     0     1     1
3 Joan~     0     1     0     1     1     1     0     0     1     1     0     0     1
# ... with 15 more variables: X29 <dbl>, X31 <dbl>, X35 <dbl>, X38 <dbl>, X39 <dbl>,
#   X41 <dbl>, X42 <dbl>, X43 <dbl>, X46 <dbl>, X47 <dbl>, X48 <dbl>, X50 <dbl>,
#   X51 <dbl>, X55 <dbl>, X57 <dbl>
```

```
> C101
# A tibble: 5 x 31
  Nama     X2    X4    X5    X6   X11   X12   X15   X16   X17   X18   X21   X24   X26
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Nugi      1     0     1     1     0     1     0     0     1     0     1     1     0
2 Pandu     1     1     0     1     1     0     0     1     0     0     1     0     0
3 Hang~     1     1     0     1     0     1     0     1     0     1     0     0     0
4 Bayu      1     1     0     1     0     1     1     0     0     0     1     0     0
5 Hend~     1     1     0     1     0     1     1     0     0     0     1     0     1
# ... with 17 more variables: X27 <dbl>, X30 <dbl>, X31 <dbl>, X35 <dbl>, X36 <dbl>,
#   X39 <dbl>, X40 <dbl>, X43 <dbl>, X44 <dbl>, X47 <dbl>, X49 <dbl>, X50 <dbl>,
```

```
> QC10
[[1]]
[1] 0.01769806

[[2]]
[1] 0.002727164
```

```
> C110
# A tibble: 3 x 29
  Nama     X2    X3    X4    X7    X8   X10   X11   X14   X17   X19   X20   X22   X25
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Uci       0     1     1     0     1     0     1     0     1     0     1     0     1
2 Adam      0     1     1     1     0     1     0     0     1     0     0     1     0
3 Abyan     1     0     1     1     0     1     0     1     0     1     0     0     0
# ... with 15 more variables: X26 <dbl>, X29 <dbl>, X31 <dbl>, X33 <dbl>, X34 <dbl>,
#   X36 <dbl>, X38 <dbl>, X40 <dbl>, X44 <dbl>, X47 <dbl>, X49 <dbl>, X50 <dbl>,
#   X51 <dbl>, X55 <dbl>, X56 <dbl>
```

```
> C111
# A tibble: 2 x 22
  Nama     X2    X4    X5    X8    X9   X13   X14   X18   X25   X30   X33   X35   X39
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Dian      1     0     1     1     1     1     0     1     1     1     1     0     0
2 Dhona     1     1     0     1     1     0     1     1     1     1     0     1     1
# ... with 8 more variables: X40 <dbl>, X43 <dbl>, X44 <dbl>, X49 <dbl>, X51 <dbl>,
#   X53 <dbl>, X55 <dbl>, X57 <dbl>
```

```
> QC11
[[1]]
[1] -0.1087361

[[2]]
[1] 0.00100066
```

Iterasi 4

```
> C1100
# A tibble: 1 x 15
  Nama     X3    X4    X8   X11   X17   X20   X25   X29   X34   X38   X44   X47   X50
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Uci       1     1     1     1     1     1     1     1     1     1     1     1     1
# ... with 1 more variable: X56 <dbl>
```

```
> C1101
# A tibble: 2 x 22
  Nama     X2    X3    X4    X7   X10   X14   X17   X19   X22   X26   X29   X31   X33
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Adam      0     1     1     1     1     0     1     0     1     1     0     1     0
2 Abyan     1     0     1     1     1     1     0     1     0     1     1     0     1
# ... with 8 more variables: X36 <dbl>, X38 <dbl>, X40 <dbl>, X44 <dbl>, X49 <dbl>,
#   X50 <dbl>, X51 <dbl>, X55 <dbl>
```

```
> QC110
[[1]]
[1] -0.01066422

[[2]]
[1] 0.002943865
```

```
> C1110
# A tibble: 1 x 15
  Nama      X2    X5    X8    X9   X13   X18   X25   X30   X33   X40   X44   X49   X51
  <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Dian       1     1     1     1     1     1     1     1     1     1     1     1     1
# ... with 1 more variable: X57 <dbl>
```

```
> C1111
# A tibble: 1 x 15
  Nama      X2    X4    X8    X9   X14   X18   X25   X30   X35   X39   X43   X49   X53
  <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Dhona      1     1     1     1     1     1     1     1     1     1     1     1     1
# ... with 1 more variable: X55 <dbl>
```

```
> QC111
[[1]]
[1] 0.04301416

[[2]]
[1] 0.002749492
```

Iterasi 5

```
> C11010
# A tibble: 1 x 15
  Nama      X3    X4    X7   X10   X17   X22   X26   X31   X36   X40   X44   X49   X50
  <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Adam       1     1     1     1     1     1     1     1     1     1     1     1     1
# ... with 1 more variable: X55 <dbl>
```

```
> C11011
# A tibble: 1 x 15
  Nama      X2    X4    X7   X10   X14   X19   X26   X29   X33   X38   X44   X49   X51
  <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Abyan      1     1     1     1     1     1     1     1     1     1     1     1     1
# ... with 1 more variable: X55 <dbl>
```

```
> QC1101
[[1]]
[1] 0.03776979

[[2]]
[1] 0.00265342
```

Algoritma DHCC dengan optimisasi

Iterasi 1

```
> C0r
# A tibble: 38 x 52
   Nama      X2    X3    X4    X5    X6    X8    X9   X10   X11   X12   X13   X14
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Sonya      1     0     1     0     1     0     0     0     1     0     0     0
 2 Malik      1     0     1     0     1     0     0     0     0     1     0     1
 3 Ayyas      0     1     1     0     1     0     0     0     1     0     0     0
 4 M Ilham    1     0     1     0     1     0     0     0     1     0     0     0
 5 Suryo      1     0     1     0     1     0     0     0     1     0     0     0
 6 Alfira     1     0     1     0     0     1     0     1     0     0     0     0
 7 Tata       1     0     1     0     0     1     0     0     1     0     0     0
 8 Hendri     1     0     1     0     1     0     1     0     0     0     1     0
```

```
> C1r
# A tibble: 16 x 55
  Nama      X2    X3    X4    X5    X6    X7    X8    X9   X10   X11   X12   X13
  <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Evan       1     0     1     0     1     0     0     0     1     0     0     0
2 Unggul     1     0     1     0     1     0     0     0     1     0     0     0
3 Burhan     0     1     1     0     1     0     0     0     1     0     0     0
4 Ichsan     1     0     1     0     1     0     0     0     0     0     1     0
5 Kresna     1     0     1     0     0     1     0     0     0     1     0     0
6 Bene       0     1     0     1     1     0     0     1     0     0     0     0
```

```
> QC
[[1]]
[1] 0

[[2]]
[1] 0.8803109
```

Iterasi 2

```
> C00r
# A tibble: 23 x 42
   Nama      X2    X3    X4    X5    X6    X8   X10   X11   X12   X13   X14   X15
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 Sonya      1     0     1     0     1     0     0     1     0     0     0     0
 2 Ayyas      0     1     1     0     1     0     0     1     0     0     0     1
 3 Tata       1     0     1     0     0     1     0     1     0     0     0     0
 4 Tyas       1     0     1     0     1     0     0     1     0     0     0     1
 5 James      1     0     1     0     1     0     1     0     0     0     0     1
 6 Fatur      1     0     1     0     1     0     0     1     0     0     0     1
 7 Lyan       1     0     1     0     1     0     0     1     0     0     1     0
 8 Hambali    1     0     1     0     1     0     0     1     0     1     0     0
```

```
> C01r
# A tibble: 15 x 45
   Nama      X2    X3    X4    X6    X8    X9   X10   X11   X12   X13   X14   X15
   <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
 1 M Ilham    1     0     1     1     0     0     0     1     0     0     0     0
 2 Suryo      1     0     1     1     0     0     0     1     0     0     0     1
 3 Alfira     1     0     1     0     1     0     1     0     0     0     0     1
 4 Hendri     1     0     1     1     0     1     0     0     0     1     0     0
 5 Ai         1     0     1     0     1     0     0     1     0     0     0     0
 6 Soleh      0     1     1     1     0     0     0     0     1     0     0     0
 7 Heppy      1     0     1     1     0     0     0     1     0     0     0     1
 8 Toto       1     0     1     1     0     0     0     1     0     0     0     1
```

```
> QC0
[[1]]
[1] 1.42866

[[2]]
[1] 0.4182166
```

```
> C10r
# A tibble: 8 x 45
  Nama     X2    X3    X4    X5    X6    X7    X9   X10   X11   X12   X14   X15   X16
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Evan      1     0     1     0     1     0     0     1     0     0     1     0     0
2 Ungg~     1     0     1     0     1     0     0     1     0     0     0     1     0
3 Burh~     0     1     1     0     1     0     0     1     0     0     0     1     0
4 Ichs~     1     0     1     0     1     0     0     0     0     1     0     0     1
5 Kres~     1     0     1     0     0     1     0     0     1     0     1     0     0
6 Bene      0     1     0     1     1     0     1     0     0     0     1     0     0
7 Abyan     1     0     1     0     0     1     0     1     0     0     1     0     0
```

```
> C11r
# A tibble: 8 x 44
  Nama     X2    X3    X4    X5    X6    X7    X8    X9   X10   X11   X12   X13   X14
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Dian      1     0     0     1     0     0     1     1     0     0     0     1     0
2 Adam      0     1     1     0     0     1     0     0     1     0     0     0     0
3 Nugi      1     0     0     1     1     0     0     0     0     0     1     0     0
4 Pandu     1     0     1     0     1     0     0     0     0     1     0     0     0
5 Hang~      1     0     1     0     1     0     0     0     0     0     1     0     0
6 Joan~     0     1     0     1     1     0     0     1     0     0     0     0     0
7 Dhona     1     0     1     0     0     0     1     1     0     0     0     0     1
```

```
> QC1
[[1]]
[1] -0.4220172

[[2]]
[1] -0.02271652
```

Iterasi 3

```
> C100r
# A tibble: 4 x 31
  Nama     X2    X4    X6    X7   X10   X11   X14   X15   X18   X19   X20   X24   X25
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Evan      1     1     1     0     1     0     1     0     0     1     0     0     1
2 Ungg~     1     1     1     0     1     0     0     1     1     0     0     1     0
3 Kres~     1     1     0     1     0     1     1     0     0     0     1     0     1
4 Abyan     1     1     0     1     1     0     1     0     0     1     0     0     0
# ... with 17 more variables: X26 <dbl>, X28 <dbl>, X29 <dbl>, X33 <dbl>, X38 <dbl>,
#   X39 <dbl>, X41 <dbl>, X43 <dbl>, X44 <dbl>, X46 <dbl>, X48 <dbl>, X49 <dbl>,
#   X51 <dbl>, X52 <dbl>, X55 <dbl>, X56 <dbl>, X58 <dbl>
```

```
> C101r
# A tibble: 4 x 34
  Nama     X2    X3    X4    X5    X6    X9   X10   X11   X12   X14   X15   X16   X18
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Burh~     0     1     1     0     1     0     1     0     0     0     1     0     0
2 Ichs~     1     0     1     0     1     0     0     0     1     0     0     1     0
3 Bene      0     1     0     1     1     1     0     0     0     1     0     0     1
4 Ajun      1     0     1     0     1     0     0     1     0     0     1     0     0
# ... with 20 more variables: X19 <dbl>, X21 <dbl>, X23 <dbl>, X24 <dbl>, X28 <dbl>,
#   X29 <dbl>, X30 <dbl>, X33 <dbl>, X34 <dbl>, X35 <dbl>, X37 <dbl>, X39 <dbl>,
#   X41 <dbl>, X45 <dbl>, X46 <dbl>, X50 <dbl>, X51 <dbl>, X52 <dbl>, X57 <dbl>,
```

```
> QC10
[[1]]
[1] -0.03098296

[[2]]
[1] -0.003151906
```

```
> C110r
# A tibble: 3 x 30
  Nama      X2    X3    X4    X5    X6    X8    X9   X13   X14   X16   X18   X19   X24
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Dian      1     0     0     1     0     1     1     1     0     0     1     0     0
2 Joan~     0     1     0     1     1     0     1     0     0     1     0     1     1
3 Dhona     1     0     1     0     0     1     1     0     1     0     1     0     0
# ... with 16 more variables: X25 <dbl>, X30 <dbl>, X31 <dbl>, X33 <dbl>, X35 <dbl>,
#   X39 <dbl>, X40 <dbl>, X43 <dbl>, X44 <dbl>, X48 <dbl>, X49 <dbl>, X50 <dbl>,
#   X51 <dbl>, X53 <dbl>, X55 <dbl>, X57 <dbl>
```

```
> C111r
# A tibble: 5 x 34
  Nama      X2    X3    X4    X5    X6    X7   X10   X11   X12   X15   X16   X17   X18
  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Nugi      1     0     0     1     1     0     0     0     1     0     0     1     0
2 Adam      0     1     1     0     0     1     1     0     0     0     0     1     0
3 Pandu     1     0     1     0     1     0     0     1     0     0     1     0     0
4 Hang~     1     0     1     0     1     0     0     0     1     0     1     0     1
5 Bayu      1     0     1     0     1     0     0     0     1     1     0     0     0
# ... with 20 more variables: X21 <dbl>, X22 <dbl>, X24 <dbl>, X26 <dbl>, X27 <dbl>,
#   X30 <dbl>, X31 <dbl>, X36 <dbl>, X39 <dbl>, X40 <dbl>, X44 <dbl>, X47 <dbl>,
#   X49 <dbl>, X50 <dbl>, X51 <dbl>, X52 <dbl>, X53 <dbl>, X55 <dbl>, X56 <dbl>,
```

```
> QC11
[[1]]
[1] -0.1223536

[[2]]
[1] -0.002376498
```

Iterasi 4

```
> QC100
[[1]]
[1] 0.01665095

[[2]]
[1] 0.001233404
```

```
> QC101
[[1]]
[1] -0.05920169

[[2]]
[1] 0.002323129
```

```
> QC110
[[1]]
[1] -0.03339402

[[2]]
[1] 0.003124356
```

```
> QC111
[[1]]
[1] -0.005629767

[[2]]
[1] 0.002671658
```

Iterasi 5

```
> QC1010
[[1]]
[1] 0.03894157

[[2]]
[1] 0.002694628
```

```
> QC1011
[[1]]
[1] 0.02378291

[[2]]
[1] 0.00264971
```

```
> QC1100
[[1]]
[1] 0.04301416

[[2]]
[1] 0.002749492
```

```
> QC1110
[[1]]
[1] 0.03227955

[[2]]
[1] 0.004142208
```

```
> QC1111
[[1]]
[1] 0.02371544

[[2]]
[1] 0.002481604
```