

**LAPORAN PENELITIAN**  
**TAHUN ANGGARAN 2024**

**PENGEMBANGAN ANALISIS *HIERARCHICAL AGGLOMERATIVE CLUSTERING* UNTUK DATA KATEGORIK DAN PEMBUATAN PACKAGE NYA MENGGUNAKAN BAHASA PEMROGRAMAN R**

Nomor DIPA	:	DIPA-025.04.2.423812/2024
Tanggal	:	24 November 2023
Satker	:	(423812) UIN Maulana Malik Ibrahim Malang
Kode Kegiatan	:	(2132) Peningkatan Akses, Mutu,Relevansi dan Daya Saing Pendidikan Tinggi Keagamaan Islam
Kode Output Kegiatan	:	(BEI) Bantuan Lembaga
Sub Output Kegiatan	:	(003) BOPTN
Kode Komponen	:	(004) Dukungan Operasional Penyelenggaraan Pendidikan
Kode Sub Komponen	:	(SC) Penelitian Terapan Pengembangan Nasional (SBK)

Oleh:

Dr. Hairur Rahman, M.Si. (198004292006041003)

Prof. Dr. Hj. Sri Harini, M.Si. (197310142001122002)

Dr. Fachrur Rozi, M.Si. (198005272008011012)





**KEMENTERIAN AGAMA**  
**LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT (LP2M)**  
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG**  
**2024**

KEMENTERIAN AGAMA REPUBLIK INDONESIA  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG  
LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT

Kampus I: Jalan Gajayana 50 Malang 65144, Telp. (0341) 551354, Fax. (0341) 572533  
Website: [www.ln2m.uin-malang.ac.id](http://www.ln2m.uin-malang.ac.id) E-mail: [ln2m@uin-malang.ac.id](mailto:ln2m@uin-malang.ac.id)

## HALAMAN PERSETUJUAN

Laporan penelitian dengan judul  
**“Pengembangan Analisis *Hierarchical Agglomerative Clustering* untuk Data Kategorik dan Pembuatan *Packagenya* Menggunakan Bahasa Pemrograman R”**

Oleh:

Dr. Hairur Rahman, M.Si. (198004292006041003)

Prof. Dr. Hj. Sri Harini, M.Si. (197310142001122002)

Dr. Fachrur Rozi, M.Si. (198005272008011012)

Telah diperiksa dan disetujui *reviewer* dan komite penilai pada tanggal 20 November 2024

Malang, 20 November 2024

*Reviewer 1,*

Reviewer 2,

## Komite Penilai,

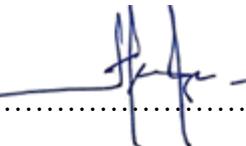


### **HALAMAN PENGESAHAN**

Laporan Penelitian ini disahkan oleh Lembaga Penelitian dan Pengabdian kepada  
Masyarakat Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Pada tanggal 20 November 2024

Peneliti

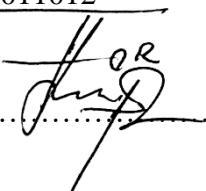
Ketua : Dr. Hairur Rahman, M.Si.  
NIP. 198004292006041003

Tanda Tangan ..... 

Anggota I : Prof. Dr. Hj. Sri Harini, M.Si.  
NIP. 197310142001122002

Tanda Tangan ..... 

Anggota II : Dr. Fachrur Rozi, M.Si.  
NIP. 198005272008011012

Tanda Tangan ..... 

Ketua LP2M  
UIN Maulana Malik Ibrahim Malang

Prof. Dr. H. Agus Maimun, M.Pd.  
NIP. 19650817 199803 1 003

## DAFTAR ISI

HALAMAN PERSETUJUAN .....	ii
HALAMAN PENGESAHAN .....	iii
DAFTAR ISI .....	iv
DAFTAR GAMBAR.....	v
DAFTAR TABEL .....	vi
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	4
1.3. Tujuan Penelitian .....	4
BAB II TINJAUAN PUSTAKA .....	5
2.1. Analisis Klaster.....	5
2.2. Hierarchical Agglomerative Clustering (HAC) .....	5
2.3. Data Kategorik.....	8
2.4. Bahasa Pemrograman R dan Pembuatan Package R .....	9
BAB III METODE PENELITIAN .....	10
BAB IV HASIL PENELITIAN.....	12
4.1. Pengembangan HAC menggunakan Jarak Hamming.....	12
4.2. Pembangunan <i>Package</i> Program HAC menggunakan Jarak Hamming.....	13
4.3. Contoh Penggunaan Package Program HAC menggunakan Jarak Hamming.....	17
BAB V TEMUAN PENELITIAN.....	21
BAB VI PENUTUP.....	23
DAFTAR PUSTAKA.....	24
LAMPIRAN .....	27
Lampiran 1. Hak Cipta Program.....	27

## **DAFTAR GAMBAR**

**Gambar 1.** Hasil Dendogram dari HAC pada Kasus Spike Protein virus SARS COV ..... 20

## **DAFTAR TABEL**

## **BAB I**

### **PENDAHULUAN**

#### **1.1. Latar Belakang**

Analisis klaster adalah salah satu metode statistik multivariat yang bertujuan untuk mengelompokkan objek-objek ke dalam beberapa kelompok berdasarkan kesamaan atau perbedaan karakteristiknya. Analisis klaster banyak digunakan dalam berbagai bidang, seperti biologi, psikologi, pemasaran, dan bisnis. Dalam analisis klaster, terdapat berbagai metode yang dapat digunakan, salah satunya adalah Hierarchical Agglomerative Clustering (HAC).

HAC merupakan metode analisis klaster yang bekerja dengan cara menggabungkan objek-objek yang paling mirip secara bertahap hingga semua objek membentuk satu kelompok. HAC dapat digunakan untuk menganalisis data numerik dan data kategorik. Namun, metode HAC yang ada saat ini hanya dapat digunakan untuk menganalisis data numerik. Hal ini menyebabkan keterbatasan dalam penggunaan HAC untuk menganalisis data kategorik, yang sering ditemukan dalam penelitian sosial dan bisnis.

Sampai saat ini, HAC telah mengalami banyak pengembangan. Penelitian ini, yang dilakukan oleh Zhiling Cai, Xiao-fei Yang, Tianyi Huang, dan William Zhu pada tahun 2020, memperkenalkan sebuah pendekatan baru dalam clustering aglomeratif dengan menggabungkan koefisien rekonstruksi dan jarak berpasangan. Pendekatan ini, yang dikenal sebagai RCPD, menunjukkan kemampuan yang signifikan dalam mengidentifikasi struktur lokal data sambil tetap tahan terhadap noise dan penciran. RCPD mengambil keuntungan dari jarak antar titik data serta representasi linier di antara titik-titik tersebut, membuatnya efektif dalam menangani data yang kompleks. Hasil eksperimental menunjukkan bahwa metode ini mengungguli banyak pendekatan clustering terkini pada 11 dataset dunia nyata. [1].

Ian Jeantet, Z. Miklós, dan D. Gross-Amblard pada tahun 2020 mengembangkan sebuah metode baru yang diberi nama Overlapping Hierarchical Clustering (OHC). OHC mengatasi keterbatasan metode clustering hirarkis tradisional dengan memungkinkan tumpang tindih antarkluster sampai tercapai atraksi kluster yang kuat, berdasarkan kriteria kepadatan. Metode ini menghasilkan struktur hirarkis yang dikenal sebagai quasi-dendrogram, diwakili sebagai grafik acyclic berarah, menggabungkan kelebihan hierarki dengan presisi clustering yang kurang arbitrer. Penelitian ini menunjukkan bagaimana OHC

dapat memperbaiki eksplorasi dan analisis data, dengan validasi melalui eksperimen ekstensif pada set data nyata [2].

Dalam penelitian oleh A. Saeedi, L. David, dan E. Rahm pada tahun 2021, diusulkan perluasan pada Hierarchical Agglomerative Clustering (HAC) untuk meningkatkan kualitas dan efisiensi dalam mencocokkan entitas dari berbagai sumber. Penelitian ini menunjukkan bahwa dengan menggunakan sumber data yang bebas duplikasi, metode baru ini mencapai kualitas pencocokan yang lebih baik dibandingkan metode sebelumnya. Selain itu, algoritma yang diusulkan dapat dijalankan secara paralel pada kluster terdistribusi, meningkatkan skalabilitasnya untuk menangani volume data yang besar. Evaluasi pada dataset yang beragam membuktikan efektivitas pendekatan ini, menyajikan solusi baru dalam pengelolaan dan analisis data dari sumber-sumber yang berbeda [3].

Dalam penelitian tahun 2021 oleh Nicholas Monath, Kumar Avinava Dubey, dan rekan-rekannya, diperkenalkan metode clustering hirarkis aglomeratif yang dapat diaplikasikan pada skala data yang sangat besar, hingga miliaran titik data. Metode ini unik karena mampu mempertahankan kualitas clustering tanpa mengorbankan kecepatan, sebuah kemajuan signifikan dibandingkan metode yang ada. Melalui analisis teoretis dan eksperimen empiris yang ekstensif, metode ini terbukti efektif baik dalam pengaturan clustering hirarkis maupun flat, menunjukkan hasil yang mengungguli metode-metode state-of-the-art lainnya dalam benchmark yang tersedia untuk publik. Penerapannya pada dataset yang sangat besar, seperti 30 miliar query, menunjukkan skalabilitas dan efektivitasnya yang luar biasa [4].

Studi oleh S. Pasupathi, V. Shanmuganathan, M. Kaliappan, Y. H. Robinson, dan Mucheol Kim pada tahun 2021 fokus pada penggunaan clustering hirarkis aglomeratif untuk menganalisis tren dalam data besar time series, dengan kasus studi khusus pada data kecelakaan lalu lintas. Pendekatan ini berhasil mengelompokkan data time series yang besar menjadi kluster-kluster berdasarkan karakteristik serupa, memungkinkan analisis tren yang lebih efisien. Pendekatan clustering ini terbukti lebih optimal daripada teknik statistik tradisional, menawarkan cara baru dalam memahami dan mengelola fenomena kecelakaan lalu lintas yang kompleks. Hasil penelitian ini memberikan wawasan penting dalam upaya pengurangan tragedi kecelakaan lalu lintas global [5].

Tahun 2021, Emmendorfer dan Canuto mengenalkan Generalization of the Average Linkage (GAL) yang merupakan pengembangan dari Average Linkage. Hasil penelitian tersebut menunjukkan bahwa dengan penggunaan GAL, memberikan hasil yang baik, melampaui semua metode referensi yang lain [6].

Di tahun 2021, Chan dan kawan-kawan mengembangkan HAC menjadi Agglomerative Info-Clustering dengan mengelompokkan variabel acak berdasarkan korelasi total yang di normalisasi. Klaim hasil penelitian ini adalah dengan pendekatan ini memberikan hasil hierarki kluster yang sama dengan metode divisive. Namun, dari segi kecepatan, metode proses Agglomerative Info-Clustering jauh lebih cepat dibandingkan metode divisive [7].

Di tahun 2022, Jafarzadegan dan kawan-kawan mengembangkan HAC berdasarkan pendekatan berbasis ensemble. Kerangka HAC dikembangkan berdasarkan pengelompokan klaster berdasarkan kriteria keasamaan inovatif yang melakukan pengelompokan melalui pendekatan ensemble. Algoritma tersebut terdiri dari 3 langkah utama. Pada langkah pertama, sekelompok metode AHC tunggal digabungkan untuk mendekripsi hubungan antara sampel dan serta pembentukan gugusan awal. Kesamaan sampel dihitung menggunakan kriteria kesamaan inovatif berdasarkan gugusan yang dibuat. Pada langkah kedua, semua klaster awal yang dibuat oleh metode yang berbeda dikelompokkan kembali untuk membentuk hiper-klaster. Setelah pengelompokan klaster, setiap sampel ditetapkan ke hiper-klaster dengan kesamaan maksimum untuk membuat klaster akhir pada langkah ketiga. Studi eksperimen yang komprehensif telah dilakukan untuk mengevaluasi kinerja algoritma yang diusulkan berdasarkan beberapa dataset. Hasilnya jelas menunjukkan bahwa kerangka kerja berbasis HAC ensemble yang diusulkan berkinerja lebih baik daripada metode yang umum saat ini [8].

Selain penelitian yang membahas pengembangan HAC, ada pula penelitian mengenai cara mengajarkan HAC. Xizhen Cai dan Qing Wang pada tahun 2020 memperkenalkan sebuah alat pendidikan dan aktivitas belajar aktif untuk mengajar clustering hierarkis aglomeratif dalam kelas statistik. Mereka menggunakan papan bulletin dan pushpin berwarna untuk memvisualisasikan proses clustering, memfasilitasi pembelajaran kolaboratif dan interaktif. Melalui metode ini, mahasiswa belajar untuk menggabungkan pushpin menjadi satu kluster tunggal, menerapkan kriteria penggabungan seperti single linkage, complete linkage, atau group average linkage. Pendekatan kreatif ini memperkuat pemahaman konsep dan praktik clustering, memberikan pengalaman pembelajaran yang lebih mendalam dan interaktif bagi mahasiswa statistik [9].

Di lain sisi, juga ada penelitian yang membahas keunggulan HAC. Seperti Murtagh dan Contreras (2017) memberikan gambaran umum tentang algoritma HAC yang efisien dalam mengelompokkan data untuk mining dan visualisasi data. Mereka meninjau berbagai implementasi efisien yang tersedia di R dan lingkungan perangkat lunak lainnya [10].

Selain itu ada juga penelitian yang membahas mengenai kekurangan dari HAC. Penelitian yang dilakukan oleh Alabart dan kawan-kawan pada tahun 2022 menunjukkan bahwa dalam hierarki cluster terdapat kelemahan yaitu hasil HAC kemungkinan tidak unik (*nonunique*). Hal tersebut disebabkan oleh jarak yang terikat (tied distances). Penelitian ini mencoba mengkaji beberapa hasil artikel di scopus mengenai HAC yang menunjukkan 46% artikel yang menggunakan HAC untuk menganalisis penanda mikrosatelit memiliki lebih dari satu solusi clustering hierarki [11].

Untuk mengatasi keterbatasan tersebut, perlu dikembangkan metode HAC yang dapat digunakan untuk menganalisis data kategorik. Metode HAC untuk data kategorik ini diharapkan dapat membantu para peneliti dan praktisi dalam menganalisis data kategorik dengan lebih mudah dan akurat. Selain itu, perlu juga dikembangkan paket perangkat lunak yang dapat digunakan untuk menerapkan metode HAC untuk data kategorik. Paket perangkat lunak ini diharapkan dapat memudahkan para pengguna dalam melakukan analisis klaster dengan metode HAC untuk data kategorik.

Hasil dari penelitian ini diharapkan dapat berkontribusi dalam pengembangan metode analisis klaster untuk data kategorik serta memberikan kemudahan bagi para peneliti dan praktisi dalam menganalisis data kategorik dengan metode HAC.

## **1.2. Rumusan Masalah**

Rumusan masalah pada penelitian ini adalah:

1. Bagaimana cara mengembangkan metode Hierarchical Agglomerative Clustering (HAC) yang dapat digunakan untuk menganalisis data kategorik?
2. Bagaimana cara membuat paket perangkat lunak yang dapat digunakan untuk menerapkan metode HAC untuk data kategorik?

## **1.3. Tujuan Penelitian**

Tujuan dari penelitian ini adalah:

1. Mengembangkan metode HAC yang dapat digunakan untuk menganalisis data kategorik.
2. Membuat paket perangkat lunak yang dapat digunakan untuk menerapkan metode HAC untuk data kategorik.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Analisis Klaster**

Analisis klaster merupakan salah satu metode statistik multivariat yang bertujuan untuk mengelompokkan objek-objek ke dalam beberapa kelompok berdasarkan kesamaan atau perbedaan karakteristiknya [12]. Analisis klaster banyak digunakan dalam berbagai bidang, seperti biologi, psikologi, pemasaran, dan bisnis. Banyak sekali metode untuk analisis klaster seperti Hierarchical Agglomerative Clustering (HAC) [13], [14], K-Mean [15], Fuzzy Clustering [16] dan DBSCAN [17].

Hasil dari analisis klaster dapat digunakan untuk berbagai keperluan, seperti mengidentifikasi kelompok-kelompok yang berbeda dalam suatu populasi, memahami struktur suatu populasi, dan memprediksi perilaku anggota suatu kelompok. Analisis klaster juga dapat digunakan untuk membuat keputusan bisnis, seperti mengelompokkan pelanggan berdasarkan perilaku pembelian mereka atau mengelompokkan produk berdasarkan kesamaan karakteristiknya.

#### **2.2. Hierarchical Agglomerative Clustering (HAC)**

Dalam analisis klaster, terdapat berbagai metode yang dapat digunakan, salah satunya adalah Hierarchical Agglomerative Clustering (HAC). HAC merupakan metode analisis klaster yang bekerja dengan cara menggabungkan objek-objek yang paling mirip secara bertahap hingga semua objek membentuk satu kelompok [12]. HAC dapat digunakan untuk menganalisis data numerik. Langkah-langkah HAC secara umum meliputi:

##### **1) Perhitungan Jarak**

Langkah pertama dalam HAC adalah menghitung jarak antara setiap pasangan objek. Jarak antara dua objek dapat dihitung menggunakan berbagai ukuran jarak, seperti jarak Euclidean, jarak Manhattan, atau jarak Minkowski. Rumus Matematis yang Digunakan:

###### **a) Jarak Euclidean**

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

###### **b) Jarak Manhattan**

$$d(x, y) = \sum_{i=1}^n |x_i - y_i|$$

c) Jarak Minkowski

$$d(x, y) = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

2) Pembentukan Matriks Jarak

Setelah jarak antara setiap pasangan objek dihitung, maka jarak-jarak tersebut disusun dalam bentuk matriks jarak. Matriks jarak merupakan matriks simetris yang berisi jarak antara setiap pasangan objek. Rumus untuk membentuk matriks jarak adalah sebagai berikut:

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

dengan:

$D$  adalah matriks jarak.

$d_{ij}$  adalah jarak antara objek  $i$  dan objek  $j$ .

$n$  adalah jumlah objek.

Matriks jarak merupakan matriks simetris, yang berarti bahwa  $d_{ij} = d_{ji}$  untuk semua  $i$  dan  $j$ .

3) Teori Penggabungan / Agglomerative

Langkah selanjutnya dalam HAC adalah melakukan penggabungan objek secara bertahap. Penggabungan objek dilakukan dengan cara menggabungkan dua objek yang paling mirip. Objek yang paling mirip adalah objek yang memiliki jarak terdekat. Terdapat beberapa rumus penggabungan (*Linkage*) yang dapat digunakan pada tahap agglomerative dalam Hierarchical Agglomerative Clustering (HAC). Rumus-rumus tersebut digunakan untuk menghitung jarak antara kelompok-kelompok yang akan digabungkan. Beberapa rumus yang umum digunakan adalah:

a) Complete Linkage

$$d(C_1, C_2) = \max_{x \in C_1, y \in C_2} d(x, y)$$

Rumus ini menghitung jarak antara dua kelompok sebagai jarak terbesar antara dua objek dari kedua kelompok tersebut.

- b) Single Linkage

$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$

Rumus ini menghitung jarak antara dua kelompok sebagai jarak terkecil antara dua objek dari kedua kelompok tersebut.

- c) Average Linkage

$$d(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{x \in C_1, y \in C_2} d(x, y)$$

Rumus ini menghitung jarak antara dua kelompok sebagai jarak rata-rata antara semua pasangan objek dari kedua kelompok tersebut.

- d) Centroid Linkage

$$d(C_1, C_2) = d(\bar{x}_1, \bar{x}_2)$$

dengan:

$\bar{x}_1$  adalah centroid kelompok  $C_1$ .

$\bar{x}_2$  adalah centroid kelompok  $C_2$ .

Rumus ini menghitung jarak antara dua kelompok sebagai jarak antara centroid kedua kelompok tersebut.

- e) Ward's Method

$$d(C_1, C_2) = \frac{|C_1| + |C_2|}{|C_1 + C_2|} d(\bar{x}_1, \bar{x}_2) + \frac{|C_1||C_2|}{|C_1 + C_2|} d(\bar{x}_1, \bar{x}_2)$$

dengan:

$|C_1|$  adalah jumlah anggota kelompok  $C_1$ .

$|C_2|$  adalah jumlah anggota kelompok  $C_2$ .

$d(\bar{x}_1, \bar{x}_2)$  adalah jarak antara centroid kedua kelompok.

#### 4) Pembentukan Dendogram

Setelah semua objek digabungkan menjadi satu kelompok, maka hasil pengelompokan tersebut dapat digambarkan dalam bentuk dendogram. Dendogram adalah diagram pohon yang menunjukkan bagaimana objek-objek digabungkan menjadi kelompok-kelompok yang lebih besar.

Algoritma *Hierarchical Agglomerative Clustering* (HAC) adalah sebagai berikut:

- 1) Inisialisasi
  - a) Buat matriks jarak yang berisi jarak antara setiap pasangan objek.
  - b) Setiap objek dianggap sebagai kelompok tersendiri.
- 2) Iterasi

Ulangi langkah-langkah berikut hingga semua objek berada dalam satu kelompok:

  - a) Temukan dua kelompok yang paling mirip (memiliki jarak terdekat).
  - b) Gabungkan kedua kelompok tersebut menjadi satu kelompok baru.
  - c) Hitung jarak antara kelompok baru tersebut dengan semua kelompok lainnya.
  - d) Perbarui matriks jarak.
- 3) Hasil

Hasil dari HAC adalah dendrogram, yang menunjukkan bagaimana objek-objek dikelompokkan.

### **2.3. Data Kategorik**

Data kategorik adalah data yang mewakili kategori atau kelompok tertentu. Data kategorik dapat berupa nominal atau ordinal. Data nominal adalah data yang tidak memiliki urutan tertentu, seperti jenis kelamin atau warna rambut. Data ordinal adalah data yang memiliki urutan tertentu, seperti tingkat pendidikan atau tingkat pendapatan.

Data kategorik sering ditemukan dalam penelitian sosial dan bisnis [18]. Misalnya, dalam penelitian pemasaran, data kategorik dapat digunakan untuk mengelompokkan pelanggan berdasarkan jenis kelamin, usia, atau tingkat pendapatan mereka. Dalam penelitian kesehatan, data kategorik dapat digunakan untuk mengelompokkan pasien berdasarkan jenis penyakit, tingkat keparahan penyakit, atau pengobatan yang diterima.

Analisis data kategorik dapat dilakukan dengan menggunakan berbagai metode statistik, seperti analisis frekuensi, analisis kontingensi, dan analisis klaster [19]. Analisis frekuensi digunakan untuk menghitung jumlah atau proporsi objek dalam setiap kategori [20]. Analisis kontingensi digunakan untuk menguji apakah ada hubungan antara dua atau lebih variabel kategorik [21]. Analisis klaster digunakan untuk mengelompokkan objek-objek berdasarkan kesamaan atau perbedaan karakteristiknya [22]. Pemilihan metode analisis data kategorik yang tepat tergantung pada tujuan penelitian dan jenis data yang dianalisis.

## 2.4. Bahasa Pemrograman R dan Pembuatan Package R

Bahasa pemrograman R adalah bahasa pemrograman dan perangkat lunak bebas untuk komputasi statistik dan grafis. R dikembangkan oleh para statistikawan dan ilmuwan komputer untuk memenuhi kebutuhan analisis data yang kompleks dan fleksibel. R memiliki banyak fitur yang membuatnya populer di kalangan peneliti dan analis data, di antaranya:

- 1) Mudah dipelajari dan digunakan, dengan sintaks yang mirip dengan bahasa pemrograman lainnya seperti Python dan MATLAB.
- 2) Gratis dan *open source*, sehingga dapat digunakan oleh siapa saja tanpa perlu membayar lisensi.
- 3) Memiliki banyak *library* dan *package* yang dapat diunduh dan diinstal untuk berbagai keperluan analisis data.
- 4) *Community* yang besar dan aktif, yang menyediakan dukungan dan berbagi pengetahuan antar pengguna R.

*Package R* adalah kumpulan fungsi, data, dan dokumentasi yang terkait dengan topik tertentu. *Package R* dapat dibuat oleh siapa saja dan dibagikan dengan pengguna R lainnya. Untuk membuat *Package R*, Anda perlu mengikuti langkah-langkah berikut:

- 1) Buat direktori baru untuk *package* Anda.
- 2) Buat file `DESCRIPTION` di direktori tersebut. File ini berisi informasi tentang *package* Anda, seperti nama, versi, dan deskripsi singkat.
- 3) Buat file `NAMESPACE` di direktori tersebut. File ini berisi informasi tentang fungsi dan data yang terdapat dalam \_package\_ Anda.
- 4) Buat file `R` untuk setiap fungsi yang ingin Anda sertakan dalam *package* Anda.
- 5) Buat file `data` untuk setiap data set yang ingin Anda sertakan dalam *package* Anda.
- 6) Buat file `vignette` untuk dokumentasi *package* Anda.
- 7) Instal *package* Anda menggunakan fungsi `install.packages()`.

Setelah Anda menginstal *package* Anda, Anda dapat menggunakan fungsi dan data yang terdapat dalam *package* tersebut dalam skrip R Anda. Misalnya, jika Anda memiliki fungsi `my\_function()` dalam *package* `my\_package`, Anda dapat menggunakan fungsi tersebut dalam skrip R Anda dengan cara berikut:

```
library(my_package)  
my_function()
```

## **BAB III**

### **METODE PENELITIAN**

Dalam rangka menjawab rumusan masalah pada penelitian ini terdapat dua hal besar yang akan dilakukan, yaitu:

- 1) Pengembangan Metode HAC untuk Data Kategorik
  - a) Mengembangkan fungsi jarak untuk data kategorik.
  - b) Mengembangkan algoritma HAC yang dapat menggunakan fungsi jarak tersebut.
  - c) Menguji kinerja metode HAC yang dikembangkan dengan menggunakan data simulasi dan data nyata.
- 2) Pembuatan Paket Perangkat Lunak
  - a) Membuat paket perangkat lunak yang berisi fungsi-fungsi yang diperlukan untuk melakukan analisis HAC untuk data kategorik.
  - b) Menulis dokumentasi untuk paket perangkat lunak tersebut.

Tahapan yang perlu dilakukan adalah sebagai berikut:

- 1) Studi Literatur
  - a) Mengkaji penelitian-penelitian sebelumnya yang terkait dengan analisis klaster dan HAC.
  - b) Mengidentifikasi metode HAC yang ada saat ini dan keterbatasannya dalam menganalisis data kategorik.
  - c) Menentukan metode HAC yang tepat untuk dikembangkan untuk menganalisis data kategorik.
- 2) Pengembangan Metode HAC untuk Data Kategorik
  - a) Mengembangkan fungsi jarak baru untuk data kategorik.
  - b) Mengembangkan algoritma HAC baru yang dapat menggunakan fungsi jarak tersebut.
  - c) Menguji kinerja metode HAC yang dikembangkan dengan menggunakan data simulasi dan data nyata.
- 3) Pembuatan Paket Perangkat Lunak
  - a) Membuat paket perangkat lunak yang berisi fungsi-fungsi yang diperlukan untuk melakukan analisis HAC untuk data kategorik.
  - b) Menulis dokumentasi untuk paket perangkat lunak tersebut.
- 4) Pengujian and Evaluasi

- a) Menguji kinerja paket perangkat lunak yang dikembangkan dengan menggunakan data simulasi dan data nyata.
  - b) Mengevaluasi kemudahan penggunaan paket perangkat lunak tersebut.
- 5) Penyusunan Laporan Penelitian
- Menyusun laporan penelitian yang berisi tentang latar belakang penelitian, rumusan masalah, metode penelitian, hasil dan pembahasan.

## **BAB IV**

### **HASIL PENELITIAN**

#### **4.1. Pengembangan HAC menggunakan Jarak Hamming**

Algoritma *Hierarchical Agglomerative Clustering* (HAC) dapat diadaptasi untuk menggunakan jarak Hamming untuk mengelompokkan data kategorikal atau biner. Jarak Hamming adalah metrik untuk membandingkan dua string data biner. Ini mengukur jumlah posisi di mana simbol yang sesuai berbeda, atau dalam istilah yang lebih sederhana, ini menghitung jumlah perubahan yang diperlukan untuk mengubah satu string menjadi string lainnya. Adaptasi ini sangat berguna ketika berurusan dengan data kategorikal, di mana konsep 'jarak' tidak sesederhana dengan data numerik.

Modifikasi Matematis menggunakan jarak Hamming dilakukan melalui tahap-tahap berikut:

a. Perhitungan Jarak

Ubah langkah penghitungan jarak awal untuk menghitung jarak Hamming antara pasangan titik data. Untuk dua vektor data  $x$  dan  $y$  dengan panjang  $n$  yang sama, jarak Hamming  $d_H(x, y)$  dihitung sebagai:

$$d_H(x, y) = \sum_{i=1}^n I(x_i \neq y_i)$$

di mana  $I$  adalah fungsi indikator yang menghasilkan 1 jika kondisi  $x_i \neq y_i$  benar, dan 0 jika tidak. Ini menjumlahkan semua contoh di mana elemen-elemen yang sesuai dari  $x$  dan  $y$  berbeda.

b. Konstruksi Matriks

Buatlah matriks jarak awal dengan menggunakan jarak Hamming yang dihitung untuk semua pasangan titik data. Matriks ini akan simetris karena  $d_H(x, y) = d_H(y, x)$ , dengan angka nol di sepanjang diagonal (karena jarak setiap titik dengan dirinya sendiri adalah nol).

c. Proses Pengelompokan Aglomeratif

Proses pengelompokan aglomeratif tetap sama, di mana pada setiap langkah, dua cluster dengan jarak Hamming terkecil digabungkan. Pilihan kriteria keterkaitan (misalnya, tunggal, lengkap, rata-rata) akan menentukan bagaimana jarak antar cluster dihitung:

- Single Linkage: Jarak antara dua klaster adalah jarak Hamming minimum antara satu anggota klaster pertama dan satu anggota klaster kedua.

- Complete Linkage: Jarak antara dua cluster adalah jarak Hamming maksimum antara anggota tunggal cluster pertama dan anggota tunggal cluster kedua.
- Average Linkage: Jarak antara dua cluster adalah jarak Hamming rata-rata di antara semua pasangan anggota dari dua cluster.

d. Memperbarui Matriks Jarak

Setelah menggabungkan dua klaster, matriks jarak harus diperbarui untuk mencerminkan jarak antara klaster baru dan semua klaster lainnya. Metode pembaruan tergantung pada kriteria keterkaitan yang digunakan.

e. Kondisi Pengakhiran

Proses ini terus berlanjut hingga semua titik data digabungkan ke dalam satu klaster atau hingga jumlah klaster yang ditentukan tercapai, tergantung pada tujuan pengelompokan.

Pertimbangan Utama:

- a. Menggunakan jarak Hamming untuk HAC sangat efektif untuk set data kategorikal atau biner di mana ukuran jarak tradisional seperti jarak Euclidean tidak dapat diterapkan.
- b. Pemilihan kriteria keterkaitan dapat secara signifikan mempengaruhi struktur cluster yang dihasilkan, terutama mengingat sifat diskrit dari jarak Hamming.
- c. Pendekatan ini dapat menjadi intensif secara komputasi untuk dataset yang besar karena kebutuhan untuk menghitung ulang jarak antar cluster pada setiap langkah proses aglomerasi.

Dengan mengadaptasi HAC untuk menggunakan jarak Hamming, algoritma ini menjadi serbaguna untuk jenis data yang lebih luas, termasuk data kategorikal dan biner, sehingga meningkatkan kegunaannya dalam berbagai konteks analisis data.

## 4.2. Pembangunan *Package* Program HAC menggunakan Jarak Hamming

Pengembangan *package* program HAC ini diberi nama *package* NOHAC yang merupakan singkatan dari *Nominal Hierarchical Agglomerative Clustering*. Package ini dibangun dengan membuat 3 buah fungsi yaitu:

### 1) Fungsi hamming\_distance

Tujuan dari fungsi ini adalah untuk menghitung jarak Hamming antara dua vektor, vec1 dan vec2. Jarak Hamming adalah jumlah elemen yang berbeda di antara dua vektor dengan panjang yang sama. Fungsi ini bekerja dengan cara membandingkan elemen pada setiap posisi dalam kedua vektor, kemudian menghitung berapa banyak elemen yang tidak sama. Pada bagian vec1 != vec2, perbandingan elemen dilakukan secara

berpasangan, dan hasilnya adalah nilai logika TRUE untuk elemen yang berbeda dan FALSE untuk elemen yang sama. Fungsi sum kemudian menghitung total TRUE, yang menunjukkan jumlah perbedaan antara dua vektor tersebut, yang merupakan jarak Hamming..

```
#' Calculate Hamming Distance Between Two Vectors
#'
#' Computes the Hamming distance between two vectors. The
#' Hamming distance is the count of positions at which the
#' corresponding elements are different.
#'
#' @param vec1 First vector for comparison.
#' @param vec2 Second vector for comparison, should be of the
#' same length as `vec1`.
#'
#' @return An integer value representing the Hamming distance
#' between the two vectors.
#'
#' @examples
#' vec1 <- c(1, 0, 1, 1, 0)
#' vec2 <- c(1, 1, 1, 0, 0)
#' hamming_distance(vec1, vec2)
#'
#' @export
hamming_distance <- function(vec1, vec2) {
  sum(vec1 != vec2)
}
```

## 2) Fungsi compute\_hamming\_distance\_matrix

Fungsi compute\_hamming\_distance\_matrix bertujuan untuk menghitung matriks jarak Hamming antara setiap pasangan baris dalam sebuah data matriks atau data frame. Fungsi ini bekerja dengan cara pertama-tama menentukan jumlah baris dalam data melalui nrow(data), yang kemudian digunakan untuk membuat sebuah matriks berukuran size x size yang diinisialisasi dengan nilai NA. Fungsi ini memiliki dua loop bersarang, di mana loop luar dan dalam masing-masing berjalan untuk setiap baris dalam data. Di dalam kedua loop tersebut, fungsi hamming\_distance dipanggil untuk menghitung jarak Hamming antara dua baris data yang sedang dibandingkan, dan hasilnya disimpan dalam elemen yang sesuai di matriks jarak. Setelah seluruh perhitungan selesai, fungsi as.dist digunakan untuk mengonversi matriks jarak menjadi objek jarak (distance object) yang lebih sesuai untuk analisis klaster atau pengukuran jarak lainnya. Dengan cara ini, fungsi ini menghasilkan matriks yang berisi jarak Hamming antara semua pasangan baris dalam data.

```
#' Compute Hamming Distance Matrix
#'
```

```

#' Calculates a distance matrix for a dataset using the
# Hamming distance, where each element (i, j) in the matrix
# represents the distance between row i and row j of the input
# data.
#'
#' @param data A matrix or data frame of observations (rows)
# and variables (columns). Data should be suitable for Hamming
# distance calculation.
#'
#' @return An object of class `dist` representing the pairwise
# Hamming distances between all rows in the dataset.
#'
#' @examples
#' data <- matrix(sample(0:1, 20, replace = TRUE), nrow = 5)
#' distance_matrix <- compute_hamming_distance_matrix(data)
#'
#' @export
#' @importFrom stats as.dist hclust

compute_hamming_distance_matrix <- function(data) {
  size <- nrow(data)
  matrix_distance <- matrix(NA, nrow = size, ncol = size)
  for (i in 1:size) {
    for (j in 1:size) {
      matrix_distance[i, j] <- hamming_distance(data[i, ],
data[j, ])
    }
  }
  as.dist(matrix_distance)
}

```

### 3) Fungsi nohac

Fungsi nohac di R digunakan untuk melakukan pengelompokan hierarkis (hierarchical clustering) berdasarkan jarak Hamming antara baris-baris dalam sebuah matriks. Fungsi ini menerima beberapa argumen: X, yang merupakan matriks input; method, yang mengatur metode pengelompokan hierarkis (default "ward.D2"); dan kl, yaitu jumlah kluster yang diinginkan (default 2).

Pertama, fungsi memeriksa apakah X adalah matriks yang valid dengan setidaknya satu kolom. Jika tidak memenuhi syarat, fungsi akan menghentikan eksekusi dan memberikan pesan kesalahan. Kemudian, fungsi menghitung matriks jarak Hamming menggunakan fungsi compute\_hamming\_distance\_matrix, yang menghitung jarak antara setiap pasangan baris dalam X.

Setelah mendapatkan matriks jarak, fungsi melakukan pengelompokan hierarkis menggunakan fungsi hclust dengan metode yang ditentukan (misalnya, "ward.D2"). Hasil dari pengelompokan ini adalah objek kluster. Setelah itu, fungsi cutree digunakan

untuk memotong dendrogram hasil pengelompokan ke dalam kl jumlah kluster, dan hasil klustering ini ditambahkan sebagai kolom baru bernama clusters dalam matriks X.

Fungsi ini juga menghasilkan plot dendrogram untuk visualisasi hasil pengelompokan. Pada akhirnya, fungsi ini mengembalikan list yang berisi matriks jarak Hamming, objek kluster, dan matriks X yang telah diberi label kluster. Perlu dicatat bahwa plot dendrogram dihasilkan secara langsung dan tidak dikembalikan sebagai objek.

```
' Perform Hierarchical Agglomerative Clustering Using Hamming
Distance
#
#' This function performs hierarchical agglomerative
clustering on a dataset
#' using Hamming distance as the measure of dissimilarity
between observations.
#' It is particularly useful for binary or categorical data.
The function
#' computes the Hamming distance matrix, performs hierarchical
clustering,
#' and plots the resulting dendrogram.
#
#' @param X A matrix where rows are observations and columns
are binary or
#' categorical variables.
#' @param method The clustering method to be used. Available
methods include:
#' \itemize{
#'   \item \code{"ward.D"}: Minimizes variance within
clusters, an older version
#'   of \code{"ward.D2"}.
#'   \item \code{"single"}: Uses the minimum of the distances
between all
#'   observations of two clusters.
#'   \item \code{"complete"}: Uses the maximum of the
distances between all
#'   observations of two clusters.
#'   \item \code{"average"}: Uses the average of the distances
between all
#'   observations of two clusters.
#'   \item \code{"mcquitty"}: Similar to \code{"average"}, also
known as WPGMA.
#'   \item \code{"median"}: Uses the median of the distances
between all
#'   observations of two clusters, known as WPGMC.
#'   \item \code{"centroid"}: Uses the distance between the
centroids of two
#'   clusters.
#'   \item \code{"ward.D2"}: Minimizes the total within-
cluster variance,
#'   recommended for most cases.
#' }
#' Default is \code{"ward.D2"}.
#'
```

```

#' @return Returns a list with two components:
\code{hamming_distance_matrix},
#' the computed Hamming distance matrix, and \code{cluster},
an object of class
#' \code{hclust} representing the hierarchical clustering
model. Note: The
#' dendrogram plot is generated as a side effect and is not
directly returned
#' by the function.
#
#' @examples
#' # Generating a sample binary data matrix
#' set.seed(123)
#' data_sample <- matrix(sample(0:1, 20, replace = TRUE), nrow
= 5)
#' result <- nohac(data_sample)
#
#' @importFrom stats hclust
#' @export
nohac <- function(X, method = "ward.D2", k1 = 2) {
  # Ensure 'X' is a matrix with appropriate dimensions
  if (!is.matrix(X) || ncol(X) < 1) {
    stop("Input 'X' must be a matrix with at least one
column.")
  }

  # Compute the Hamming distance matrix
  hamming_distance_matrix <-
  compute_hamming_distance_matrix(X)

  # Perform hierarchical clustering
  cluster <- hclust(hamming_distance_matrix, method = method)
  clusters <- cutree(cluster, k = k1)
  X$clusters <- clusters

  # Plot the dendrogram
  plot(cluster, main = "Hierarchical Clustering with Hamming
Distance")

  # Return a list containing the Hamming distance matrix, the
clustering object, and potentially the plot
  return(list(
    hamming_distance_matrix = hamming_distance_matrix,
    cluster = cluster,
    X = X
    # Note: The plot itself is not an object that can be
    # returned in R. It's rendered directly.
  ))
}

```

### 4.3. Contoh Penggunaan Package Program HAC menggunakan Jarak Hamming

Contoh penggunaan ini mengambil kasus sekuen DNA dari spike protein virus SARS COV. Menggunakan berbagai paket dalam R, termasuk Biostrings dan DECIPHER untuk membaca, memanipulasi, dan melakukan multiple sequence alignment (MSA) pada urutan

DNA. Analisis ini diawali dengan membaca data sekuens dari file FASTA dan mengonversinya menjadi objek DNAStringSet, yang merupakan struktur data yang dirancang untuk menyimpan urutan DNA. Selanjutnya, beberapa urutan dipilih dan dikonversi menjadi format yang lebih mudah diolah, seperti data frame dan CSV. Setelah itu, proses multiple sequence alignment diterapkan menggunakan paket DECIPHER, yang membantu menemukan kesamaan atau pola di antara urutan DNA. Hasil MSA kemudian dikonversi ke dalam bentuk matriks dan data frame untuk mempermudah analisis lanjutan. Pada bagian akhir, package nohac digunakan untuk melakukan pengelompokan hierarkis (hierarchical clustering) terhadap hasil alignment, yang memungkinkan pengelompokan urutan DNA berdasarkan kesamaan struktur mereka, dengan jarak Hamming sebagai ukuran *dissimilarity*. Analisis ini merupakan pendekatan komprehensif untuk memahami struktur dan hubungan di antara data sekuens biologis.

```

if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")

BiocManager::install("Biostrings")

library(Biostrings)

# Baca file FASTA
fasta_file <- "E:\\RAW Data of SARS-CoV-2 Spike
Protein\\spike_seqs_NCBI.fasta"# Ganti dengan path yang sesuai
sequences <- readDNAStringSet(fasta_file)

# Lihat beberapa urutan
sequences

-----MENGAMBIL DATA BEBERAPA BARIS-----
# Misalkan 'sequences' adalah objek DNAStringSet Anda
# Mengambil 10 urutan pertama
first_ten_sequences <- sequences[3:12]

# Mengonversi ke data.frame
sequences_df <- data.frame(
  index = 1:10,
  width = width(first_ten_sequences),
  sequence = as.character(first_ten_sequences),
  names = names(first_ten_sequences)
)

# Menulis data.frame ke file CSV
write.csv(sequences_df, "first_ten_sequences.csv", row.names =
FALSE, quote = FALSE)

-----MULTIPLE SEQUENCE ALIGNMENT-----

```

```

#Multiple Sequence Alignment (MSA) adalah proses yang mengatur
urutan DNA, RNA, atau protein untuk mengidentifikasi wilayah
yang mirip antara urutan.
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")

BiocManager::install("DECIPHER")

library(DECIPHER)

# Mengambil 10 urutan pertama
first_ten_sequences <- sequences[3:1002]

# Melakukan MSA
myXStringSet <- DNAStringSet(first_ten_sequences)
alignedSequences <- AlignSeqs(myXStringSet)

# Melihat hasilnya
alignedSequences

-----TRANSFER HASIL MSA KE EXCEL-----
-----Mengonversi ke data.frame
sequences_df <- data.frame(
  index = 1:10,
  width = width(alignedSequences),
  sequence = as.character(alignedSequences),
  names = names(alignedSequences)
)

# Menulis data.frame ke file CSV
write.csv(sequences_df, "alignedSequences.csv", row.names =
FALSE, quote = FALSE)

-----KONVERSI KE DALAM BENTUK
MATRIKS-----
# Asumsi 'alignedSequences' adalah hasil dari
DECIPHER::AlignSeqs atau objek XStringSet yang serupa

# Mengonversi 'alignedSequences' menjadi matriks
aligned_matrix <- as.matrix(alignedSequences)

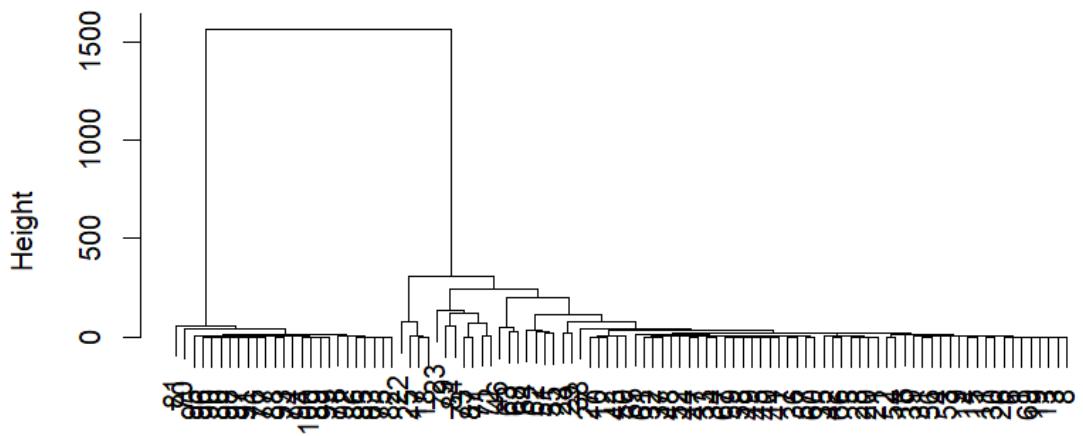
# Untuk konversi ke data.frame, jika diperlukan
aligned_df <- as.data.frame(aligned_matrix, stringsAsFactors =
TRUE)

# Sekarang, 'aligned_df' adalah data.frame dimana tiap kolom
merepresentasikan satu posisi
# dan setiap baris adalah satu urutan yang telah di-align

-----NOHAC-----
-----library(nohac)
result<-nohac(aligned_matrix)
result<-nohac(aligned_matrix, method = "complete")

```

### Hierarchical Clustering with Hamming Distance



```
hamming_distance_matrix  
hclust (*, "ward.D2")
```

**Gambar 1.** Hasil Dendogram dari HAC pada Kasus Spike Protein virus SARS COV

## **BAB V**

### **TEMUAN PENELITIAN**

#### **Temuan Penelitian dan Pengembangan HAC dengan Jarak Hamming**

Penelitian ini berhasil mengembangkan metode Hierarchical Agglomerative Clustering (HAC) yang dapat menganalisis data kategorik dengan menerapkan jarak Hamming sebagai ukuran dissimilarity. Adaptasi ini menunjukkan efektivitas yang signifikan ketika diaplikasikan pada data yang bersifat kategorikal, di mana jarak Hamming dapat mengukur perbedaan dengan menghitung jumlah elemen yang berbeda dalam pasangan data. Temuan ini relevan untuk data kategorik yang seringkali memerlukan pendekatan berbeda dibandingkan dengan data numerik.

#### **Efektivitas dan Implementasi Package 'NOHAC'**

Hasil dari penelitian ini juga dikemas dalam bentuk package ‘nohac’, hasil temuan kami dalam bentuk package ini memfasilitasi proses analisis klaster untuk data kategorik secara lebih efisien. Package ini dilengkapi dengan tiga fungsi utama: hamming\_distance, compute\_hamming\_distance\_matrix, dan nohac. Fungsi hamming\_distance bertujuan untuk menghitung jarak Hamming antara dua vektor kategorik, sementara compute\_hamming\_distance\_matrix menghasilkan matriks jarak Hamming antara setiap pasangan data, dan nohac berfungsi untuk melaksanakan pengelompokan hierarkis berbasis jarak Hamming. Hasil pengujian menunjukkan bahwa package ini memberikan hasil pengelompokan yang baik, serta memudahkan pengguna dalam proses visualisasi hasil melalui dendrogram, sehingga meningkatkan kemudahan analisis bagi para peneliti.

#### **Analisis Implementasi pada Kasus Nyata dan Simulasi**

Dalam aplikasi nyata, metode HAC berbasis jarak Hamming ini diuji pada data kategorikal dalam konteks sekuens DNA dari spike protein SARS-CoV-2. Melalui analisis menggunakan package 'NOHAC', sekuens DNA yang telah aligned kemudian diklasterkan berdasarkan kesamaan strukturalnya. Hasil menunjukkan bahwa metode HAC berbasis jarak Hamming mampu menghasilkan klaster yang relevan dan sesuai dengan struktur data biologis. Ini menunjukkan bahwa jarak Hamming dapat secara efektif mengidentifikasi hubungan antar-data yang kompleks dalam dataset kategorikal, sehingga metode ini dapat berfungsi sebagai alat yang kuat dalam analisis data berbasis kategori.

#### **Peningkatan Aksesibilitas dan Dampak Temuan**

Secara keseluruhan, pengembangan HAC berbasis jarak Hamming dan package 'NOHAC' memberikan kontribusi penting dalam metodologi analisis klaster untuk data kategorik. Temuan ini memungkinkan para peneliti di berbagai bidang, termasuk ilmu sosial dan

biologi, untuk mengimplementasikan metode klustering hierarkis secara lebih efisien dan relevan. Paket ini diharapkan akan memberikan dampak positif dalam penelitian-penelitian masa depan yang berfokus pada data kategorik, serta mendorong penerapan lebih lanjut di kalangan akademisi dan praktisi.

## **BAB VI**

### **PENUTUP**

Penelitian ini berhasil mengembangkan metode Hierarchical Agglomerative Clustering (HAC) berbasis jarak Hamming, yang mampu mengatasi keterbatasan HAC tradisional dalam menganalisis data kategorik. Dengan menerapkan jarak Hamming sebagai ukuran dissimilarity, metode ini memberikan pendekatan yang lebih adaptif dan relevan dalam mengelompokkan data kategorik, sehingga meningkatkan akurasi dan ketepatan klaster yang dihasilkan. Implementasi dalam package 'NOHAC' menyediakan alat yang efisien bagi peneliti dan praktisi untuk melakukan analisis klaster secara praktis, bahkan bagi pengguna yang memiliki pemahaman dasar tentang bahasa pemrograman R. Pengujian dan aplikasi nyata pada data sekvens DNA SARS-CoV-2 menunjukkan bahwa metode ini efektif dalam mengidentifikasi struktur data berdasarkan kesamaan, menjadikannya solusi yang handal untuk berbagai aplikasi data kategorik. Hasil ini menunjukkan bahwa pengembangan HAC berbasis jarak Hamming memberikan kontribusi penting dalam peningkatan analisis klaster untuk data yang bersifat kategorikal.

## DAFTAR PUSTAKA

- [1] Nealson, “Connected Commerce: Connectivity is Enabling Lifestyle Evolution,” 2018. <https://www.nielsen.com/apac/en/insights/report/2018/connected-commerce-connectivity-is-enabling-lifestyle-evolution/>.
- [2] R. Ammigan, J. L. Dennis, and E. Jones, “The differential impact of learning experiences on international student satisfaction and institutional recommendation,” *J. Int. Students*, vol. 11, no. 2, pp. 299–321, 2021, doi: 10.32674/jis.v11i2.2038.
- [3] M. Maqableh and M. Alia, “Evaluation online learning of undergraduate students under lockdown amidst COVID-19 Pandemic: The online learning experience and students’ satisfaction,” *Child. Youth Serv. Rev.*, vol. 128, no. August 2020, p. 106160, 2021, doi: 10.1016/j.childyouth.2021.106160.
- [4] N. Prawesty, “Pengaruh kualitas layanan Internet service provider seluler pada kepuasan mahasiswa terhadap perkuliahan Daring Masa Pandemi COVID-19,” 2021, [Online]. Available: <http://digilib.uinsby.ac.id/id/eprint/48638>.
- [5] L. Efriana, “Problems of Online Learning during Covid-19 Pandemic in EFL Classroom and the Solution,” *JELITA J. English Lang. Teach. Lit.*, vol. 2, no. 1, pp. 2721–1916, 2021.
- [6] A. Destianingsih and A. Satria, “Investigating Students’ Needs for Effective English Online Learning During Covid-19 for Polbeng Students,” *ELT-Lectura*, vol. 7, no. 2, pp. 147–153, 2020, doi: 10.31849/elt-lectura.v7i2.4657.
- [7] I. P. P. Suryawan and D. Permana, “Media Pembelajaran Online Berbasis Geogebra sebagai Upaya Meningkatkan Pemahaman Konsep Matematika,” *Prisma*, vol. 9, no. 1, p. 108, 2020, doi: 10.35194/jp.v9i1.929.
- [8] A. A. Utomo, A. Imron, and M. Syaiful, “Pengaruh Penjelasan Guru Terhadap Pemahaman Siswa pada Mata Pelajaran Sejarah,” *PESAGI (Jurnal Pendidik. dan Penelit. Sejarah)*, vol. 5, no. 8, pp. 1–12, 2017.
- [9] N. P. Sari, “Pengaruh Kompetensi Pedagogik Guru terhadap Tingkat Pemahaman Belajar Siswa pada Mata Pelajaran Kimia,” *Semin. Nas. Pendidikan, Sains dan Teknol.*, pp. 243–248, 2005.
- [10] S. Gultom, A. F. Hutaaruk, and A. M. Ginting, “Teaching Skills of Teacher in Increasing Student Learning Interest,” *Budapest Int. Res. Critics Inst. Humanit. Soc. Sci.*, vol. 3, no. 3, pp. 1564–1569, 2020, doi: 10.33258/birci.v3i3.1086.
- [11] A. Hutaaruk and R. Sidabutar, “Kendala pembelajaran daring selama masa pandemi

- di kalangan mahasiswa pendidikan matematika: Kajian kualitatif deskriptif,” *J. Math. Educ. Appl.*, vol. 02, no. 01, pp. 45–51, 2020, [Online]. Available: <https://jurnal.uhn.ac.id/index.php/sepren/article/view/364>.
- [12] J. M. R. Asio, E. Gadia, E. Abarintos, D. Paguio, and M. Balce, “Internet Connection and Learning Device Availability of College Students: Basis for Institutionalizing Flexible Learning in the New Normal,” *Stud. Humanit. Educ.*, vol. 2, no. 1, pp. 56–69, 2021, doi: 10.48185/she.v2i1.224.
- [13] O. D. Apuke and T. O. Iyendo, “University students’ usage of the internet resources for research and learning: forms of access and perceptions of utility,” *Heliyon*, vol. 4, no. 12, p. e01052, 2018, doi: 10.1016/j.heliyon.2018.e01052.
- [14] A. Siddiquah and Z. Salim, “The ICT facilities, skills, usage, and the problems faced by the students of higher education,” *Eurasia J. Math. Sci. Technol. Educ.*, vol. 13, no. 8, pp. 4987–4994, 2017, doi: 10.12973/eurasia.2017.00977a.
- [15] H. B. Essel, F. K. N. Nunoo, A. Tachie-Menson, and J. O. Amankwa, “Higher Education Students’ Ownership and Usage of Smart Phones and Tablets: The Case of Kwame Nkrumah University of Science and Technology (KNUST).,” *Int. J. Educ. Technol.*, vol. 5, no. 1, pp. 20–28, 2018, [Online]. Available: <http://educationaltechnology.net/ijet/>.
- [16] W. Peterson, “Deskripsi Kemampuan Mengajar Guru dan Faktor-Faktor yang Mempengaruhinya,” *Fac. Psychol. Dep. Psychol.*, no. 5030061, p. 5030061, 2015.
- [17] PDDIKTI, “Profil Perguruan Tinggi Universitas Islam Maulana Malik Ibrahim,” 2021. [https://pddikti.kemdikbud.go.id/data\\_pt/N0ZGOUU1MzktN0JGNC00NDdDLUEyNTgtODBFRTdDRjlBOERG](https://pddikti.kemdikbud.go.id/data_pt/N0ZGOUU1MzktN0JGNC00NDdDLUEyNTgtODBFRTdDRjlBOERG) (accessed Jun. 14, 2022).
- [18] N. Kock and P. Hadaya, “Minimum sample size estimation in PLS-SEM: The inverse square root and gamma-exponential methods,” *Inf. Syst. J.*, vol. 28, pp. 227–261, 2018.
- [19] J. F. Hair, G. T. M. Hult, C. M. Ringle, and M. Sarstedt, *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*. California: SAGE Publications, Inc., 2013.
- [20] R. Innayah, “Pengaruh Media Pembelajaran Online, Motivasi Belajar dan Kompetensi dosen terhadap Kualitas pembelajaran,” *J. Pendidik. Ekon. UM Metro*, vol. 8, no. 2, pp. 38–47, 2020.
- [21] N. E. Noviana and M. R. Solichin, “Pengaruh Penggunaan Media Pembelajaran

Online (whatsapp dan Zoom) Terhadap Prestasi Belajar Siswa Pada Masa Pandemi Covid-19,” *J. Pendidik. Ekon.*, vol. 9, no. 2, pp. 60–64, 2021, doi: 10.26740/jupe.v9n2.p60-64.

## LAMPIRAN

### Lampiran 1. Hak Cipta Program



**LAMPIRAN PENCIPTA**

No	Nama	Alamat
1	Sri Harini	PERUM GRIYA SHANTA E-312, Lowokwaru, Malang
2	Angga Dwi Mulyanto	Jl Raya Pilang No. 17 RT 004 RW 002 Desa Pilang, Wonoayu, Malang
3	Hairur Rahman	DSN. KRENGGAN, NGEBONG, Pakel, Tulungagung
4	Fachrur Rozi	Kutisari Indah Barat IV/90 Surabaya, Tenggilis Mejoyo, Surabaya

