

PENGELOMPOKAN DATA KATEGORIK DENGAN ALGORITMA ROCK

SKRIPSI

MPM 18/06

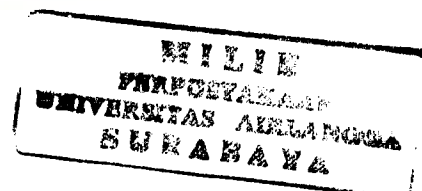
1/06



M SUGENG MULYONO

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS AIRLANGGA
SURABAYA**

2006



Handwritten signature

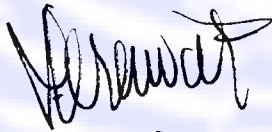
Handwritten signature

LEMBAR PENGESAHAN SKRIPSI

Judul : PENGELOMPOKAN DATA KATEGORIK DENGAN
ALGORITMA *ROCK*
Penyusun : M SUGENG MULYONO
NIM : 080112418
Tanggal Ujian : 24 Januari 2006

Disetujui Oleh :

Pembimbing I



Ir. Dyah Herawatie, M. Si
NIP. 132 061 804

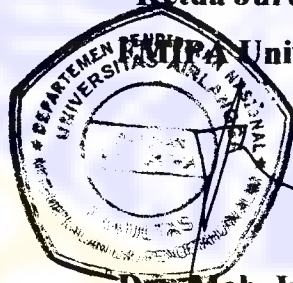
Pembimbing II



Drs. Eto Wuryanto, DEA
NIP. 131 933 015

Mengetahui :

Ketua Jurusan Matematika
Universitas Airlangga



Drs. Moh. Imam Utoyo, M. Si
NIP. 131 801 397

*Bukankah Kami telah melapangkan untukmu dadamu.
Dan Kami telah mnghilangkan darimu beban.
Yang memberatkan punggungmu.
Dan Kami tinggikan bagimu sebutan (nama)mu.
Karena sesungguhnya sesudah kesulitan itu ada kemudahan.
Maka, apabila kamu telah selesai (dari suatu urusan) kerjakanlah dengan
sungguh – sungguh (urusan) yang lain.
Dan hanya kepada Rabb-mulah hendaknya kamu berharap.
(QS. Al-Insyirah: 1-8)*

*“Ya Allah, hanya rahmat-Mu yang aku harapkan,
maka janganlah Engkau sandarkan urusanku kepada diriku sendiri
biarkanpun sekejap mata. Perbaikilah segala urusanku, tidak ada
yang berhak secara benar melainkan Engkau”
(HR. Abu Daud)*

Ucapan Terimakasih

Dengan rasa syukur alhamdulillah, penulis menyampaikan ucapan terimakasih kepada :

1. Ayahanda Alm.soetardjo atas kesabaran, suritauladan, dan semua petuah bijaknya yang akan selalu kuingat dan ibunda siti fatimah atas semua pengorbanan, kasih sayang, serta do'anya yang tiada henti mengalir buat kelancaran dan kesuksesan penulis
2. saudara - saudaraku tercinta mbak is, mas edy, mas agus, mas puguh, mas kus, adik rahma, mbak retno, adik lontar atas semua bantuannya & senantiasa mendorong agar penulis tetap tegar menghadapi segala rintangan.
3. Ir. Dyah Herawatie, M.Si dan Drs. Eto Wuryanto, DEA selaku dosen pembimbing yang dengan ketulusan dan kesabarannya telah membimbing penulis dalam menyelesaikan skripsi ini.
4. Drs. Ardi Kurniawan, M.Si selaku dosen wali, bapak dan ibu dosen pengajar serta seluruh karyawan di lingkungan FMIPA UNAIR, khususnya jurusan matematika terimakasih atas ilmu dan nasehatnya doakan agar bermanfaat.
5. Sahabat - sahabatku : Mohammad Alamsyah, Dhani "aphet", Junaidi, Didin, Imam Atas semua bantuan & kebersamaannya selama masa kuliah maupun diluar kampus.
6. Teman - teman semasa kuliah : Zaenal (sory sering tak ganggu), Ipunk, Daniel 'endro', Henry, Agus, Arif 'AS', Hanafi, Haryoko'01 (cptan dkrjakan), Erni (jgn males²), Ernina (thank

sudah mo gantikan ngajar), dewi, ida, Vonny, Dian A, Ira R, Niken, Ifti, Anita, Ira w, Dina, Nita, Raras, Soleha, Dian R, yuni, trisni, TriA, Lely, TriP, ibu ribut, ibu zumrotus akan selalu kuingat kebersamaan dengan kalian semua.

7. kakak - kakak kelas Nyamun, bamby, Pak ust. Imam S atas semua bantuanya; Imron, Abraham, Agung, bang Pea *dll* serta adik - adik kelas kurniwan (tmn seperjuangan), Yanto, Arif 'oyot', fauzi, dodol'03 *dll*
8. Temen² jurusan lain : Irwan, Dyo, irvan, Moel, Fauzan, om-acol, Tutuk, Ismanu, Pak "dhe", Arif H, Estu, Afif"grespo", taufik, Aprilia, Diana, Isti', Desiana, Erni *dll*
Miko,yudi,Iwan,Abe,Wawan-FKH, Muji-FKM, Morinda-Hukum (makasih oleh² dan doanya), wiwid -HI (thank kebersamaannya), dan buat kamu tentunya indah thank bgt for doa, support, and advice²-nya.
9. Bapak² dan rekan² staf pengajar primagama WK&Perak atas support dan doanya.
10. Temen² tempo doloe: Nasir,Wawan-ITS; Totok,Katon-UGM; Hermawan-UNIBRAW, Ratno-UNDIP *dll*terimakasih belum bosenn² juga jadi temanku
11. Mas Doni & Ibu kost serta keluarga maaf kalo sering telat bayar kost hihi...
12. Dan semua pihak yang mungkin belum tersebutkan yang telah membantu hingga skripsi ini terbentuk.

PEDOMAN PENGGUNAAN SKRIPSI

Skripsi ini tidak dipublikasikan, namun tersedia di perpustakaan dalam lingkungan Universitas Airlangga. Diperkenankan untuk dipakai sebagai referensi kepustakaan, tetapi pengutipan seijin penulis dan harus menyebutkan sumbernya sesuai kebiasaan ilmiah.

Dokumen skripsi ini merupakan hak milik Universitas Airlangga.

PRAKATA



Puji syukur yang sedalam – sedalamnya penulis panjatkan kehadiran Allah SWT atas berkat, rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pengelompokan Data Kategorik Dengan Algoritma *ROCK*”.

Sholawat serta salam semoga tetap tercurahkan atas Nabi Besar Muhammad SAW, pemimpin serta panutan kami, yang senantiasa sebagai inspirator serta pendorong segala gerak langkah kami untuk menyelesaikan skripsi ini.

Pada kesempatan ini, penulis juga menyampaikan ucapan terimakasih yang sebesar – besarnya kepada Ibu Ir. Dyah Herawatie, M.Si dan Bapak Drs. Eto Wuryanto, DEA sebagai pembimbing yang telah meluangkan waktunya untuk memberikan bimbingan dan nasehat serta rekan – rekan yang telah membantu dalam penyelesaian skripsi ini.

Akhir kata kami mengharapkan, semoga skripsi ini dapat memberikan sesuatu yang bermanfaat bagi pembaca pada umumnya dan jurusan matematika pada khususnya.

Surabaya, januari 2006

Penyusun

M Sugeng Mulyono, 2005. Pengelompokan data kategorik dengan algoritma *ROCK*. Skripsi ini dibawah bimbingan Ir. Dyah Herawatie, M.Si dan Drs. Eto Wuryanto, DEA. Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam. Universitas Airlangga.

ABSTRAK

Skripsi ini bertujuan untuk melakukan pengelompokan data berjenis kategori dengan menggunakan algoritma *ROCK* (*RObust Clustering using linKs*). Algoritma *ROCK* termasuk dalam klasifikasi algoritma pengelompokan hirarki *agglomerative* dalam analisis kelompok. Algoritma ini memperkenalkan konsep yang disebut dengan *neighbors* dan *link* dalam melakukan pengelompokan data.

Pengelompokan data kategorik dengan algoritma *ROCK* dilakukan dengan tiga langkah. Langkah pertama menghitung similaritas. Langkah kedua menentukan *neighbors* dan yang terakhir menghitung *link* antar obyek pengamatan. Besarnya *link* dipengaruhi oleh nilai θ yang diinputkan oleh *user*. Besarnya nilai θ yang diinputkan adalah $0 \leq \theta \leq 1$. Penggabungan kelompok dengan menggunakan algoritma *ROCK* didasarkan atas ukuran kebaikan (*goodness measure*) antar kelompok.

Dari Program S – Plus yang telah dibuat, diterapkan pada tiga data. Pada data I dengan menggunakan algoritma hirarki konvensional pengelompokan dijadikan menjadi dua kelompok akan tetapi bila menggunakan algoritma *ROCK* dengan $\theta = 0.5$ hanya bisa dijadikan paling kecil dua kelompok. Data II dengan $\theta = 0.42$ paling kecil didapatkan tiga kelompok. Data III dengan menggunakan $\theta = 0.7$ dan $k = 3$ pengelompokan didapatkan tiga kelompok. Hasil pengelompokan dengan algoritma *ROCK* tidak hanya ditentukan oleh besarnya nilai k akan tetapi juga memperhatikan *link* antar kelompok.

Kata Kunci : Algoritma *ROCK*, Data Kategorik, Analisis Kelompok.

M Sugeng Mulyono, 2005. Clustering of categorical data by using *ROCK* algorithm. This *skripsi* is under guidance of Ir. Dyah Herawatie, M.Si and Drs.Eto Wuryanto, DEA. Mathematics Department Faculty of Mathematics and Natural Science. Airlangga University.

ABSTRACT

This *skripsi* aim to cluster categorical data by using ROCK (RObust Clustering using linKs) algorithm. ROCK algorithm is one of agglomerative hierarchichal algorithm classification in cluster analysis. This algorithm introduce new concept which called neighbors and link in the data clustering.

Clustering of categorical data by using ROCK algorithm can be applied for three steps. The first step, counting similarity. The second determine the neighbors and last step count the link between observation object. The link value is effected by input user, θ value ($0 \leq \theta \leq 1$). Clustering ROCK algorithm based on goodness measure between cluster.

This S – Plus program is applied for three data. The first used konvensional hierarchichal algorithm the clustering result two group but used ROCK algorithm with $\theta = 0.5$ the clustering result at least two group. The second data with $\theta = 0.42$ the clustering result at least three group and in the third data with $\theta = 0.7$ and $k = 3$ the clustering result three cluster. These results are determined by both k value and link between cluster.

Key word : ROCK algorithm, Categorical data, Cluster analysis.

DAFTAR ISI

HALAMAN JUDUL.....	i
KATA PENGANTAR.....	ii
ABSTRAK.....	iii
ABSTRACT.....	iv
DAFTAR ISI.....	v
DAFTAR TABEL.....	vii
DAFTAR GAMBAR	viii
DAFTAR LAMPIRAN.....	ix
BAB I. PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Analisis Kelompok.....	4
2.1.1 Metode Hirarki.....	4
2.1.2 Metode Non-Hirarki.....	6
2.2 Data Kategorik.....	6
2.3 Ukuran Kemiripan dan Ketidakmiripan.....	7
2.4 Algoritma <i>ROCK</i>	8
2.5 S – Plus.....	10

BAB III METODE PENULISAN	13
BAB IV HASIL DAN PEMBAHASAN	15
4.1 Algoritma Pengelompokan Hirarki <i>Agglomerative</i>	15
4.2 Algoritma <i>ROCK</i>	15
4.2.1 Jumlah Kelompok	15
4.2.2 Penentuan Nilai k dan θ	16
4.3 Data	18
4.4 Pengelompokan Data Dengan Manual	19
4.4.1 Penerapan Algoritma Hirarki Konvensional	19
4.4.2 Penerapan Algoritma <i>ROCK</i>	25
4.5 Penerapan Algoritma Kedalam Program Komputer.....	31
4.6 Analisis Data.....	32
BAB V KESIMPULAN DAN SARAN	39
5.1 Kesimpulan	39
5.2 Saran.....	39
DAFTAR PUSTAKA	40
LAMPIRAN	

DAFTAR TABEL

Nomor	Judul Tabel	Halaman
1.	Pengamatan t dan r	7
2.	Karakteristik Individu	19
3.	Hasil pengelompokan data I dengan menggunakan algoritma hirarki konvensional dan algoritma <i>ROCK</i>	33
3.	Hasil pengelompokan data II dengan menggunakan algoritma hirarki konvensional algoritma <i>ROCK</i>	35
4.	Hasil pengelompokan data III dengan menggunakan Algoritma <i>ROCK</i>	37

DAFTAR GAMBAR

Nomor	Judul Gambar	Halaman
1.	Dendogram dari data I dengan metode keterkaitan tunggal (<i>single linkage</i>).....	32
2.	Dendogram obyek – oyek pengamatan dengan metode keterkaitan tunggal (<i>single linkage</i>) pada data II.....	34

DAFTAR LAMPIRAN

Nomor	Judul Lampiran
1.	Daftar program
2.	Data karakteristik Individu
3.	Data observasi terhadap jenis mobil
4.	Data ciri – ciri jamur

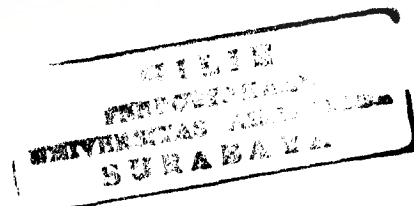
BAB I

PENDAHULUAN

1.1 Latar Belakang

Analisis kelompok (*cluster analysis*) merupakan salah satu metode yang ada dalam analisis multivariat. Tujuan dari analisis ini adalah mengelompokkan obyek-obyek pengamatan menjadi beberapa kelompok berdasarkan variabel- variabel yang diamati, sedemikian hingga obyek-obyek yang terdapat pada kelompok yang sama mempunyai karakteristik yang relatif homogen dan obyek-obyek antar kelompok mempunyai karakteristik yang berbeda (Sharma, 1996). Dalam Guha, et.al (1999) disebutkan penerapan analisis kelompok dalam bidang ekonomi. Misalnya, dalam database pelanggan (*market basket database*) yang memuat satu transaksi per pelanggan, masing-masing transaksi memuat satu set item apa saja yang dibeli oleh pelanggan. Data transaksi tersebut dapat digunakan untuk mengelompokkan pelanggannya sedemikian hingga pelanggan dengan pola belanja sejenis akan berada dalam kelompok yang sama.

Pada awalnya, hampir semua pengembangan metode pengelompokan terkonsentrasi pada data numerik. Dengan berbagai macam konsep jarak yang digunakan sebagai ukuran ketidak miripan (*disimilarity*), misalnya jarak Euclid, *Minkowski metric*, *city-block metric*. Ukuran ketidak miripan yang akan digunakan tergantung pada kebutuhan. Dalam konteks *database*, banyak



data yang berjenis kategori. Sehingga pengelompokan dengan menggunakan metode tersebut memberikan hasil yang kurang memuaskan.

Banyak ditemukan algoritma-algoritma yang digunakan untuk mengelompokkan data yang berjenis kategori salah satunya adalah *ROCK* (*RObust Clustering using linKs*). Algoritma ini dikembangkan untuk mengatasi salah satu masalah penting dalam *data mining*, yaitu mengelompokkan data. Didalam penelitian ini akan dikaji bagaimana algoritma *ROCK* melakukan pengelompokan data. Dipilihnya algoritma ini, dikarenakan algoritma ini dikatakan sebagai algoritma yang *robust* (tegar).

1.2 Rumusan Masalah

- 1 Bagaimana melakukan pengelompokan data dengan menggunakan algoritma *ROCK* ?
- 2 Bagaimana mengimplementasikan algoritma tersebut ke dalam program dalam bahasa S-Plus ?

1.3 Tujuan

1. Melakukan pengelompokan data dengan menggunakan algoritma *ROCK*..
2. Mengimplementasikan algoritma tersebut ke dalam program dalam bahasa S-Plus.

1.4 Manfaat

1. Hasil penulisan ini diharapkan dapat memberikan masukan bagi para peneliti yang ingin mengelompokkan data dengan menggunakan analisis kelompok, dan data yang dipunyai berjenis kategori. Karena belum banyak buku yang membahas analisis kelompok dengan data berjenis kategori.
2. Dapat membantu pekerjaan-pekerjaan yang terkait dengan *data mining* atau *knowledge discovery indatabase*, misalkan : pengambilan keputusan, perencanaan strategis, dan monitoring system.

BAB II

TINJAUAN PUSTAKA

2.1 Analisis kelompok

Analisis Kelompok adalah sebuah analisis yang secara umum bertujuan mengelompokkan obyek-obyek pengamatan menjadi beberapa kelompok berdasarkan pengukuran variabel-variabel yang diamati, sedemikian hingga obyek-obyek yang terdapat pada kelompok yang sama adalah relatif homogen dari pada obyek yang berada pada kelompok yang berbeda. (Everitt,1994)

Menurut Halkidi, at.al (2001), algoritma pengelompokan dapat diklasifikasikan menurut tipe atau jenis data, pendefinisian kriteria pengelompokan, serta teori dan konsep dasar dimana teknik pengelompokan didasarkan. Menurut metode yang digunakan untuk mendefinisikan kelompok-kelompok, algoritma pengelompokan dapat diklasifikasikan menjadi pengelompokan non-hirarki dan pengelompokan hirarki

2.1.1 Metode Hirarki

Metode pengelompokan hirarki digunakan untuk mengelompokkan pengamatan secara terstruktur berdasarkan sifat kemiripannya, dan banyaknya kelompok yang menyusun data belum diketahui. Ada dua cara untuk mendapatkan kelompok dengan metode ini, yaitu agglomeratif dan devisif. Metode pengelompokan hirarki agglomeratif berusaha mengelompokkan data

dengan menggabungkan pengamatan atau kelompok secara bertahap sehingga pada akhirnya hanya didapat satu kelompok saja. Sebaliknya, metode divisif dimulai dengan membentuk satu kelompok besar beranggotakan seluruh pengamatan. Kelompok tersebut kemudian dipisah menjadi kelompok yang lebih kecil, sampai satu kelompok yang hanya beranggotakan satu pengamatan saja. (Sharma, 1996)

Pengelompokan dengan algoritma hirarki *agglomerative* mempunyai beberapa macam yang dibedakan menurut jarak antar kelompok yaitu sebagai berikut :

1. Keterkaitan tunggal (*single linkage, nearest neighbors method*).

Ukuran jarak yang digunakan adalah :

$$d_{(U,V)W} = \min \{d_{UW}, d_{VW}\} \quad (2.1)$$

U dan V obyek yang digabung menjadi kelompok (UV)

$d_{(U,V)W}$ adalah jarak antara kelompok (UV) dan kelompok W

Nilai dari d_{UW} dan d_{VW} adalah jarak terkecil antara kelompok U dan W dan kelompok V dan W .

2. Keterkaitan lengkap (*complete linkage, farthest neighbor method*)

Ukuran jarak yang digunakan adalah :

$$d_{(U,V)W} = \max \{d_{UW}, d_{VW}\} \quad (2.2)$$

U dan V obyek yang digabung menjadi kelompok (UV)

$d_{(U,V)W}$ adalah jarak antara kelompok (UV) dan kelompok W

Nilai dari d_{UW} dan d_{VW} adalah jarak terjauh antara kelompok U dan W dan kelompok V dan W .

3. Keterkaitan rata-rata (*average linkage method*).

Ukuran jarak yang digunakan adalah :

$$d_{(U,V)W} = \frac{\sum_i \sum_k d_{ik}}{N_{(UV)}N_W} \quad (2.3)$$

U dan V obyek yang digabung menjadi kelompok (UV)

$d_{(U,V)W}$ adalah jarak antara kelompok (UV) dan kelompok W

d_{ik} adalah jarak antara obyek i dalam kelompok (UV) dan obyek k dalam kelompok W .

$N_{(UV)}$ dan N_w masing - masing adalah banyaknya obyek dalam kelompok (UV) dan dalam kelompok W .

4. Metode centroid (*Centroid method*)

Pengelompokan dengan metode centroid pengukuran jarak dilakukan dengan menghitung kembali nilai rata – rata obyek pengamatan.

(Johnson, 1982)

2.1.2 Metode Non - Hirarki

Metode non hirarki adalah metode pengelompokan yang didasarkan pada jumlah kelompok yang telah ditentukan, dengan menempatkan anggota kelompok yang relatif homogen dalam kelompok yang sama. (Sharma, 1996)

2.2 Data Kategorik

Data kategorik adalah data yang variabelnya berjenis kategori. Dalam Agresti (1990) dijelaskan bahwa variabel kategori adalah variabel yang skala pengukurannya terdiri dari sekumpulan kategori. Variabel ini biasa digunakan

dalam ilmu sosial atau biomedis. Sering juga ditemui dalam ilmu perilaku, ekologi, pendidikan dan marketing.

2.3 Ukuran Kemiripan dan Ketidak miripan

Ukuran kemiripan dan ketidak miripan pada dasarnya merupakan kebalikan dalam arti berlawanan arah. Ukuran kemiripan sering disebut sebagai koefisien kemiripan yang biasanya bernilai antara 0 dan 1. Misalkan suatu fungsi s adalah ukuran kemiripan, yang didefinisikan pada setiap pasang obyek pengamatan. Maka dengan mudah akan diperoleh ukuran ketidak miripan yang bersesuaian, yaitu :

$$d = (\text{konstanta} - s).$$

Chatfield dan Collins (1980) menjelaskan bagaimana menentukan ukuran kemiripan untuk data biner. Untuk memperoleh ukuran kemiripan antara dua pengamatan (t dan r) dilakukan dengan meringkas kedua pengamatan tersebut seperti pada tabel :

Tabel 2.1 Pengamatan t dan r

		pengamatan r		total
		1	0	
pengamatan t	1	a	b	a+b
	0	c	d	c+d
total		a+c	b+d	$p = a + b + c + d$

Dimana p adalah total banyaknya atribut, maka :

- Sokal and Nichener (*simple matching coefficient*) :

$$S_{rt} = \frac{a + d}{a + b + c + d} \quad (2.4)$$

- Russell and Rao :

$$S_{rt} = \frac{a}{p} \quad (2.5)$$

- Jaccard (*jaccard's coefficient*) didefinisikan dengan :

$$S_{rt} = \frac{a}{a+b+c} \quad (2.6)$$

dengan a , b , c dan d adalah jumlah frekuensi pada masing – masing sel.

2.4 Algoritma *ROCK*

Algoritma *ROCK* merupakan algoritma pengelompokan hirarki *agglomerative* yang mendasarkan gagasan pengelompokannya pada *link* dan *neighbor*. Dalam Guha, et.al (2000), dijelaskan tentang algoritma *ROCK*. Sebelum ke algoritma *ROCK*, ada beberapa definisi yang perlu diketahui.

Definisi 2.1 Kemiripan (*similarity*)

$$Sim(p_i, p_j) = \frac{|P_i \cap P_j|}{|P_i \cup P_j|} \quad (2.7)$$

P_i adalah kelompok ke- i dan P_j adalah kelompok ke- j

Dimana $|P_i|$ adalah banyaknya anggota P_i

(Guha et.al., 2000)

Definisi 2.2. *Neighbors*

Sepasang titik P_i dan P_j didefinisikan sebagai *neighbors* jika

$$Sim(P_i, P_j) \geq \theta. \quad 0 \leq \theta \leq 1$$

(Guha et.al., 2000)

Definisi 2.3. Link

Link (P_i, P_j) antara dua titik didefinisikan sebagai banyaknya neighbors antara P_i dan P_j

Secara intuisi jika link (P_i, P_j) besar maka kemungkinan besar bahwa P_i dan P_j masuk dalam kelompok yang sama.

(Guha et.al., 2000)

Definisi 2.4 Ukuran Kebajikan (*Goodness Measure*)

Ukuran kebaikan $g(C_i, C_j)$ untuk menggabungkan dua kelompok C_i, C_j didefinisikan sebagai berikut :

$$g(C_i, C_j) = \frac{\text{link}(C_i, C_j)}{[(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}]} \quad (2.8)$$

dengan $\text{link}(C_i, C_j)$ adalah jumlah *cross link* antara obyek-obyek dalam C_i dan C_j , yaitu :

$$\text{link}(C_i, C_j) = \sum_{p_q \in C_i, p_r \in C_j} \text{link}(p_q, p_r) \quad (2.9)$$

dan

$$f(\theta) = \frac{1-\theta}{1+\theta} \quad (2.10)$$

(Guha et.al., 2000)

Secara ringkas algoritma pengelompokannya adalah sebagai berikut:

1. Input data set, tentukan nilai k , θ dan anggap satu titik sebagai kelompok yang terpisah.
2. Hitung *link* dari setiap pasang kelompok yang merupakan *neighbours*

3. Gabungkan dua kelompok yang mempunyai ukuran kebaikan terbesar sebagai satu kelompok.
4. Kembali ke langkah 2. penggabungan dilanjutkan sampai salah satu dari dua kriteria dipenuhi , yaitu :
 - sejumlah kelompok tertentu (k) diperoleh
 - tidak ada *link* yang tersisa diantara kelompok-kelompok yang telah terbentuk.

2.5 S-Plus

S-plus adalah suatu paket program yang memungkinkan membuat program sendiri walaupun didalamnya sudah tersedia banyak program internal yang sudah siap digunakan. Kelebihan dari program ini adalah baik program internal maupun program yang pernah dibuat dapat digunakan sebagai sub program dari program yang akan dibuat. (Everitt, 1994)

a. Plclust

Plclust digunakan untuk membuat plot dendogram berdasarkan output dari perintah internal hclust.

Plclust(tree)

Tree: hasil fungsi yang diperoleh dari fungsi hclust untuk pengelompokan dengan metode hirarki. Nilai out putnya berupa merge (urutan penggabungan anggota-anggota kelompok), height (jarak pada penggabungan anggota-anggota kelompok), order (pengurutan letak anggota- anggota kelompok).

b. Hclust

Hclust digunakan untuk membuat pengelompokan dengan metode hirarki dengan memilih salah satu teknik pada penentuan jarak atau kemiripannya.

Hclust(dist, method = "compact")

Dist : nilai jarak antara setiap anggota pengamatan

Method : metode yang digunakan dalam proses penggabungan anggota pengamatan. Ada tiga metode yang digunakan yaitu : metode keterkaitan rata-rata (average), metode pautan tunggal (connected), metode pautan lengkap (compact).

c. Round

Round digunakan untuk membulatkan desimal.

Round(x, digits= 0)

x : angka atau obyek yang kompleks

Digits : banyaknya angka dibelakang koma.

d. Rbind

Rbind digunakan untuk menggabungkan baris pada matriks atau vector.

Rbind (... , diparse.level=1)

... : vector dan atau matriks

Diparse.level=1 : vector atau matriks yang digabung.

e. fix

`fix ()` digunakan untuk masuk kedalam S – PLUS Command dan untuk memulai penulisan program

f. function

`Function ()` digunakan untuk menunjukkan fungsi yang akan digunakan dalam program

g. length

`Length ()` merupakan perintah untuk menunjukkan banyaknya data

h. matrix

Perintah `matrix ()` digunakan untuk membuat matrik dalam program S – Plus

`Matrix (c(...), nrow=..., ncol=..., byrow=T)`

`c(...)` : nilai matrik yang diinputkan

`nrow` : banyaknya baris

`ncol` : banyaknya kolom

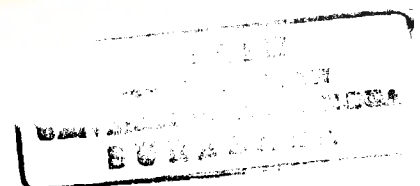
`byrow=T` : matrik disusun berdasarkan urutan baris.

i. while

Perintah `while ()` digunakan untuk looping dan berhenti jika pernyataan dipenuhi dalam program s-plus

j. scan

Perintah `scan ()` digunakan untuk membaca inputan data dalam program s-plus



BAB III

METODE PENULISAN

Dalam penulisan ini akan digunakan metode penulisan sebagai berikut :

1. Menelaah literatur yang membahas masalah analisis kelompok khususnya pada metode *ROCK*.
2. Membuat dan menyusun algoritma yang tepat untuk pengelompokan data dengan metode *ROCK* dengan algoritma adalah sebagai berikut :
 - a. Menginput data set, menentukan nilai k , θ dan menganggap satu titik sebagai kelompok yang terpisah.
 - b. Hitung similaritas masing-masing kelompok.
 - c. Tentukan *neighbour* dari setiap kelompok.
 - d. Hitung *link* dari setiap kelompok yang merupakan *neighbour*.
 - e. Gabungkan dua kelompok yang mempunyai ukuran kebaikan terbesar sebagai satu kelompok.
 - f. Kembali kelangkah d. penggabungan dilanjutkan sampai salah satu dari dua kriteria terpenuhi, yaitu :
 - Sejumlah k kelompok tertentu diperoleh
 - Tidak ada *link* yang tersisa diantara kelompok-kelompok yang telah terbentuk.
3. Menerapkan algoritma yang telah dibuat ke dalam program komputer yang menggunakan bahasa S-Plus.

4. Menerapkan program yang telah dibuat dengan menggunakan data sekunder.
5. Membandingkan hasil pengelompokan dengan menggunakan metode pengelompokan hirarki konvensional.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Algoritma Pengelompokan Hirarki *agglomerative*

Menurut **Sharma (1996)** Algoritma pengelompokan hirarki *agglomerative* adalah sebagai berikut :

1. Menganggap satu pengamatan sebagai satu kelompok.
2. Menentukan matriks jarak antar kelompok.
3. Menggabungkan dua kelompok yang mempunyai jarak terdekat atau terkecil.
4. Menentukan kembali jarak antara masing – masing kelompok.
5. Kembali kelangkah dua. Iterasi berhenti sampai semua pengamatan masuk dalam satu kelompok.

4.2 Algoritma *ROCK*

Algoritma *ROCK* merupakan algoritma pengelompokan hirarki *agglomerative* yang mendasarkan gagasan pengelompokannya pada *link* dan *neighbors*. Sepasang kelompok dikatakan *neighbors* jika mempunyai kemiripan sekurang – kurangnya berada dalam ambang batas (*threshold*) tertentu. Banyaknya *link* antar sepasang kelompok adalah banyaknya *common neighbors* atau *neighbors* bersama untuk sepasang kelompok tersebut.

4.2.1 Jumlah kelompok

Dalam pembahasan ini jumlah kelompok ditentukan oleh *user* akan tetapi *user* tidak serta merta dapat menentukan banyak kelompok.

Banyak kelompok dipengaruhi oleh θ dan *link* antar pasang kelompok atau obyek pengamatan.

4.2.2 Penentuan nilai k dan θ

K merupakan jumlah kelompok yang ingin diperoleh dan θ adalah ambang batas kemiripan. Hasil pengelompokan akan tergantung pada nilai k dan θ yang diinputkan. Besarnya nilai θ yang diinputkan oleh *user* yaitu $0 \leq \theta \leq 1$ sedangkan nilai $k \geq 1$ akan tetapi nilai k yang diinputkan tidak bisa langsung dapat dipenuhi harus melihat *link* antar kelompok.

Algoritma *ROCK* secara terperinci adalah sebagai berikut :

1. Menginput data
2. Menentukan nilai k , θ dan menganggap satu titik sebagai kelompok yang terpisah.
3. Menghitung similaritas masing-masing kelompok.
 - Anggap masing – masing obyek sebagai satu kelompok
 - Menentukan banyaknya kategori yang sama (irisan) dalam setiap variabel antar kelompok.
 - Menentukan banyaknya gabungan (union) kategori dalam setiap variabel antar kelompok.
 - Menghitung similaritas antar kelompok dengan rumus pada persamaan (2.7)
4. Menentukan *neighbors* dari setiap kelompok.
 - Menentukan nilai θ .

- Menentukan similaritas masing – masing kelompok.
 - Apabila similaritas masing – masing kelompok lebih besar atau sama dengan θ maka dianggap sebagai *neighbors* dengan definisi pada (2.2)
5. Menghitung *link* dari setiap kelompok yang merupakan *neighbors*
- Menentukan *neighbors* dari setiap kelompok.
 - Banyaknya *link* antar pasang kelompok ditentukan dari jumlah atau banyaknya *neighbors* dari pasangan kelompok tersebut.
6. Menghitung jumlah anggota kelompok.
- Jumlah anggota kelompok adalah banyaknya anggota kelompok tersebut.
7. Menghitung Ukuran Kebaikan (*goodness measure*) antar kelompok.
- Menghitung *link* antar pasang kelompok.
 - Menentukan besarnya $f(\theta)$.

$$f(\theta) = \frac{1-\theta}{1+\theta}$$

- Menghitung jumlah anggota kelompok
- Menentukan ukuran kebaikan (*goodness measure*) antar kelompok sebagai berikut :

$$g(C_i, C_j) = \frac{\text{link}(C_i, C_j)}{[(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}]}$$

8. Menggabungkan dua kelompok yang mempunyai ukuran kebaikan (*goodness measure*) terbesar sebagai satu kelompok.

9. Kembali kelangkah empat. penggabungan dilanjutkan sampai salah satu dari dua kriteria terpenuhi, yaitu :

- Sejumlah k kelompok tertentu diperoleh.
- Tidak ada *link* yang tersisa diantara kelompok-kelompok yang telah terbentuk.

Dalam pembahasan ini jumlah kelompok ditentukan oleh *user* akan tetapi *user* tidak serta merta dapat menentukan banyak kelompok. Banyak kelompok dipengaruhi oleh θ dan *link* antar pasang kelompok atau obyek pengamatan.

4.3 DATA

Program yang telah dibuat akan diuji dengan menggunakan data sekunder dan pada pembahasan ini akan dipilih data sebagai berikut :

- Data I adalah data yang diambil dari data **Johnson (1975)** tentang karakteristik suatu individu. Data ini akan dikelompokkan menjadi dua kelompok, berdasarkan variabel – variabelnya yaitu : tinggi badan, berat badan, warna mata, warna rambut, tangan yang aktif, dan jenis kelamin (lampiran 2).
- Data II adalah data yang diambil dari data **Jobson (1992)**. Data ini terdiri dari 20 observasi terhadap jenis mobil yang berbeda dengan memperhatikan banyaknya bahan bakar yang digunakan, ukuran silinder, berat mobil, ukuran mesin serta negara pemproduksi (lampiran 3).

- Data III adalah data yang diambil dari *UCI machine learning repository* (www.ics.edu/~mllearn/MLRepository). Data ini terdiri atas 22 variabel yang masing-masing menunjukkan ciri-ciri fisik jamur dan terdapat 8124 jamur yang teramati namun 200 obyek saja yang digunakan dalam pengujian menggunakan program komputer (lampiran 4).

Dengan adanya data – data tersebut diharapkan dapat memperjelas pembahasan dalam penulisan ini.

4.4 Pengelompokan Data Dengan Manual

Data diambil dari **Johnson (1975)** tentang karakteristik individu

Tabel 4.1 karakteristik individu

	Height	Weight	Eye colour	Hair colour	Handedness	Gender
individu 1	68 in	140 lb	green	blond	right	female
individu 2	73 in	185 lb	brown	brown	right	male
individu 3	67 in	165 lb	blue	blond	right	male
individu 4	64 in	120 lb	brown	brown	right	female
individu 5	76 in	210 lb	brown	brown	left	male

4.4.1 Penerapan algoritma hirarki konvensional

Penerapan algoritma hirarki konvensional menggunakan ukuran kemiripan.

Mendefinisikan variabel $X_1, X_2, X_3, X_4, X_5, X_6$ dalam bentuk *binary* sebagai

berikut :

$$X_1 = \begin{cases} 1 & \text{height} \geq 72 \text{ in} \\ 0 & \text{height} < 72 \text{ in} \end{cases}$$

$$X_4 = \begin{cases} 1 & \text{blond hair} \\ 0 & \text{brown hair} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{weight} \geq 150 \text{ lb} \\ 0 & \text{weight} < 150 \text{ lb} \end{cases}$$

$$X_5 = \begin{cases} 1 & \text{right handed} \\ 0 & \text{left handed} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{brown eyes} \\ 0 & \text{yang lain} \end{cases}$$

$$X_6 = \begin{cases} 1 & \text{female} \\ 0 & \text{male} \end{cases}$$

Berdasarkan data tersebut didapatkan hasil sebagai berikut:

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆
individu 1	0	0	0	1	1	1
individu 2	1	1	1	0	1	0
individu 3	0	1	0	1	1	0
individu 4	0	0	1	0	1	1
individu 5	1	1	1	0	0	0

Dengan menggunakan *jaccard's coefficient* = $\frac{a}{a+b+c}$ atas dasar ketentuan

nilai a, b, c, d dan p sebagai berikut :

		Item k		
		1	0	total
Item i	1	a	b	a+b
	0	c	d	c+d
total		a+c	b+d	p = a + b + c + d

didapatkan similaritas masing – masing obyek atau individu sebagai berikut

Individu 1 dan individu 2 :

		individu 2		
		1	0	total
individu 1	1	1	2	3
	0	3	0	3
total		4	2	6

$$S_{12} = \frac{a}{a+b+c} = \frac{1}{1+2+3} = \frac{1}{6}$$

Individu 1 dan individu 3 :

		individu 3		total
		1	0	
individu 1	1	2	1	3
	0	1	2	3
total		3	3	6

$$S_{13} = \frac{a}{a+b+c} = \frac{2}{2+1+1} = \frac{2}{4}$$

Individu 1 dan individu 4 :

		individu 4		total
		1	0	
individu 1	1	2	1	3
	0	1	2	3
total		3	3	6

$$S_{14} = \frac{a}{a+b+c} = \frac{2}{2+1+1} = \frac{2}{4}$$

Individu 1 dan individu 5 :

		individu 5		total
		1	0	
individu 1	1	0	3	3
	0	3	0	3
total		3	3	6

$$S_{15} = \frac{a}{a+b+c} = \frac{0}{0+3+3} = 0$$

Individu 2 dan individu 3 :

		individu 3		total
		1	0	
individu 2	1	2	2	4
	0	1	1	2
total		3	3	6

$$S_{23} = \frac{a}{a+b+c} = \frac{2}{2+2+1} = \frac{2}{5}$$

Individu 2 dan individu 4 :

		individu 4		total
		1	0	
individu 2	1	2	2	4
	0	1	1	2
total		3	3	6

$$S_{24} = \frac{a}{a+b+c} = \frac{2}{2+2+1} = \frac{2}{5}$$

Individu 2 dan individu 5 :

		individu 5		total
		1	0	
individu 2	1	3	1	4
	0	0	2	2
total		3	3	6

$$S_{25} = \frac{a}{a+b+c} = \frac{3}{3+1+0} = \frac{3}{4}$$

Individu 3 dan individu 4 :

		individu 4		total
		1	0	
individu 3	1	1	2	3
	0	2	1	3
total		3	3	6

$$S_{34} = \frac{a}{a+b+c} = \frac{1}{1+2+2} = \frac{1}{5}$$

Individu 3 dan individu 5 :

		individu 5		total
		1	0	
individu 3	1	1	2	3
	0	2	1	3
	total	3	3	6

$$S_{35} = \frac{a}{a+b+c} = \frac{1}{1+2+2} = \frac{2}{6}$$

Individu 4 dan individu 5 :

		individu 5		total
		1	0	
individu 4	1	1	2	3
	0	2	1	3
	total	3	3	6

$$S_{45} = \frac{a}{a+b+c} = \frac{1+1}{6} = \frac{2}{6}$$

Membentuk matriks kemiripan berdasarkan koefisien kemiripan masing – masing kelompok diatas

		Kelompok				
		1	2	3	4	5
Kelompok	1	1				
	2	$\frac{1}{6}$	1			
	3	$\frac{2}{4}$	$\frac{2}{5}$	1		
	4	$\frac{2}{4}$	$\frac{2}{5}$	$\frac{1}{5}$	1	
	5	0	$\frac{3}{4}$	$\frac{1}{5}$	$\frac{1}{5}$	1



Karena matrik yang didapatkan adalah matrik kemiripan maka pengelompokan dilakukan dengan menggabung obyek yang memiliki kemiripan terbesar menjadi satu kelompok terlebih dahulu. Pengelompokan dilanjutkan dengan menghitung ukuran kemiripan kemudian penggabungan dilanjutkan dengan menggabung obyek yang memiliki kemiripan terbesar menjadi satu kelompok demikian seterusnya.

Gabungkan kelompok 2 dan 5 sehingga didapatkan matriks kemiripan yang baru sebagai berikut :

$$\begin{array}{c}
 \text{Kelompok} \\
 \begin{array}{c} 1 \\ 2,5 \\ 3 \\ 4 \end{array}
 \end{array}
 \begin{array}{c}
 \text{Kelompok} \\
 \begin{array}{c} 1 \\ 2,5 \\ 3 \\ 4 \end{array}
 \end{array}
 \begin{bmatrix}
 1 & 2,5 & 3 & 4 \\
 1 & & & \\
 \frac{1}{6} & 1 & & \\
 \frac{2}{4} & \frac{2}{5} & 1 & \\
 \frac{2}{4} & \frac{2}{5} & \frac{1}{5} & 1
 \end{bmatrix}$$

Gabungkan kelompok 3 dan 1 sehingga didapatkan matriks kemiripan yang baru sebagai berikut :

$$\begin{array}{c}
 \text{Kelompok} \\
 \begin{array}{c} 1,3 \\ 2,5 \\ 4 \end{array}
 \end{array}
 \begin{array}{c}
 \text{Kelompok} \\
 \begin{array}{c} 1,3 \\ 2,5 \\ 4 \end{array}
 \end{array}
 \begin{bmatrix}
 1,3 & 2,5 & 4 \\
 1 & & \\
 \frac{2}{5} & 1 & \\
 \frac{2}{4} & \frac{2}{5} & 1
 \end{bmatrix}$$

Gabungkan kelompok (1,3) dan 4 sehingga didapatkan matriks kemiripan yang baru sebagai berikut :

$$\begin{array}{cc} & \text{Kelompok} \\ & \begin{array}{cc} 1,3,4 & 2,5 \end{array} \\ \text{Kelompok} & \begin{array}{cc} 1,3,4 & \left[\begin{array}{cc} 1 & \\ & \end{array} \right] \\ 2,5 & \left[\begin{array}{cc} \frac{2}{5} & 1 \end{array} \right] \end{array} \end{array}$$

Gabungkan kelompok (1, 3 ,4) dan (2, 5) menjadi satu kelompok.

Penggabungan berhenti, didapatkan satu kelompok yang berangotakan seluruh Anggota kelompok.

4.4.2 Penerapan algoritma *ROCK*

1. Menentukan jumlah kelompok yang diinginkan misalnya $k = 2$ dan θ yaitu besarnya similaritas minimal yang diinginkan misalnya 0,5
2. Menghitung Similaritas antar kelompok atau obyek pengamatan

$$\text{sim}(p_1, p_2) = \frac{|p_1 \cap p_2|}{|p_1 \cup p_2|}$$

$|p_1 \cap p_2|$ = banyak kategori yang sama antara obyek 1 dan obyek 2 ada 1 yaitu right handed untuk X_3 ,

$|p_1 \cup p_2|$ = banyak seluruh kategori pada obyek 1 dan obyek 2 ada 11 antara lain tinggi < 72 in, tinggi \geq 72 in, berat < 150 lb, berat \geq 150 lb, green eye, brown eye, blond hair, brown hair, right handed, female, male.

maka $\text{sim}(p_1, p_2)$ adalah $\frac{1}{11} = 0.09$

Untuk obyek yang sama maka nilai similaritasnya adalah 1 misalnya

$$\text{sim}(p_1, p_1) = 1$$

Analog untuk semua pasangan obyek

$$\text{sim}(p_1, p_3) = \frac{4}{8} = 0.5$$

$$\text{sim}(p_1, p_4) = \frac{4}{8} = 0.5$$

$$\text{sim}(p_1, p_5) = \frac{0}{12} = 0$$

$$\text{sim}(p_2, p_3) = \frac{3}{9} = 0.33$$

$$\text{sim}(p_2, p_4) = \frac{3}{9} = 0.33$$

$$\text{sim}(p_2, p_5) = \frac{5}{7} = 0.71$$

$$\text{sim}(p_3, p_4) = \frac{2}{10} = 0.2$$

$$\text{sim}(p_3, p_5) = \frac{2}{10} = 0.2$$

$$\text{sim}(p_4, p_5) = \frac{2}{10} = 0.2$$

3. Mendapatkan *neighbors* untuk masing-masing obyek

$$\text{sim}(p_i, p_j) \geq \theta$$

Dengan $\theta = 0.5$ maka didapatkan

neighbors $[p_1] = [p_1, p_3, p_4]$ karena besarnya nilai $\text{sim}(p_1, p_3)$

dan $\text{sim}(p_1, p_4)$ lebih besar atau sama dengan 0.5

Analog untuk obyek yang lain

$$\text{neighbors } [p_2] = [p_2, p_5]$$

$$\text{neighbors } [p_3] = [p_1, p_3]$$

$$\text{neighbors } [p_4] = [p_1, p_4]$$

$$\text{neighbors } [p_5] = [p_2, p_5]$$

4. Menghitung banyaknya *link* antar obyek

$\text{link}(p_i, p_j)$ adalah banyaknya *neighbour* antara p_i dan p_j

$\text{Link}(p_1, p_3) = 2$ karena p_1 dan p_3 ada dalam *neighbors* [p_1] dan *neighbors* [p_3].

Untuk $\text{link}(p_i, p_i)$ bernilai 0

Analog untuk masing-masing pasangan obyek

$$\text{Link}(p_1, p_2) = 0 \qquad \text{Link}(p_2, p_5) = 2$$

$$\text{Link}(p_1, p_4) = 2 \qquad \text{Link}(p_3, p_4) = 1$$

$$\text{Link}(p_1, p_5) = 0 \qquad \text{Link}(p_3, p_5) = 0$$

$$\text{Link}(p_2, p_3) = 0 \qquad \text{Link}(p_4, p_5) = 0$$

$$\text{Link}(p_2, p_4) = 0$$

Untuk mempermudah proses pengelompokan dapat dibuat matrik berdasarkan nilai *link* dari masing – masing kelompok atau obyek pengamatan.

Matriks *link* :

		Kelompok				
		1	2	3	4	5
Kelompok	1	0	0	2	2	0
	2		0	0	0	2
	3			0	1	0
	4				0	0
	5					0

5. Proses Pengelompokan dilakukan dengan menghitung *Goodness Measure* masing-masing pasangan.

$$g(C_i, C_j) = \frac{\text{link}(C_i, C_j)}{[(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}]}$$

$$\text{link}(C_i, C_j) = \sum_{p_q \in C_i, p_r \in C_j} \text{link}(p_q, p_r)$$

$$f(\theta) = \frac{1-\theta}{1+\theta}$$

untuk nilai $\theta = 0.5$ maka $f(\theta) = \frac{1-0.5}{1+0.5} = 0.33$

$$g(C_1, C_3) = \frac{2}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]}$$

$$= \frac{2}{2^{1.66} - 1^{1.66} - 1^{1.66}}$$

$$= 1.7$$

$$g(C_1, C_4) = \frac{2}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]}$$

$$= \frac{2}{2^{1.66} - 1^{1.66} - 1^{1.66}}$$

$$= 1.7$$

$$\begin{aligned}
 g(C_2, C_5) &= \frac{2}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{2}{2^{1.66} - 1^{1.66} - 1^{1.66}} \\
 &= 1.7
 \end{aligned}$$

$$\begin{aligned}
 g(C_3, C_4) &= \frac{1}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{1}{2^{1.66} - 1^{1.66} - 1^{1.66}} \\
 &= 0.85
 \end{aligned}$$

$$g(C_1, C_2) = 0 \qquad g(C_2, C_4) = 0$$

$$g(C_1, C_5) = 0 \qquad g(C_3, C_5) = 0$$

$$g(C_2, C_3) = 0 \qquad g(C_4, C_5) = 0$$

gabungkan kelompok 2 dan kelompok 5 karena memiliki nilai *Goodness*

Measure tinggi yaitu 1.7

Didapatkan matriks *link* yang baru :

		Kelompok				
		1	2,5	3	4	
Kelompok	1]	0	0	2	2
	2,5			0	0	0
	3				0	1
	4					0

$$\begin{aligned}
 g(C_1, C_3) &= \frac{2}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{2}{2^{1.66} - 1^{1.66} - 1^{1.66}} \\
 &= 1.7
 \end{aligned}$$

$$\begin{aligned}
 g(C_1, C_4) &= \frac{2}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{2}{2^{1.66} - 1^{1.66} - 1^{1.66}} \\
 &= 1.7
 \end{aligned}$$

$$\begin{aligned}
 g(C_3, C_4) &= \frac{1}{[(1+1)^{1+2(0.33)} - 1^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{1}{2^{1.66} - 1^{1.66} - 1^{1.66}} \\
 &= 0.85
 \end{aligned}$$

$$g(C_1, C_{(2,5)}) = 0 \quad g(C_{(2,5)}, C_4) = 0$$

$$g(C_{(2,5)}, C_3) = 0$$

gabungkan kelompok 1 dan kelompok 4 karena memiliki nilai *Goodness*

Measure tinggi yaitu 1.7.

Didapatkan matriks *link* yang baru

		kelompok			
		1,4	2,5	3	
Kelompok	1,4]	0	0	3
	2,5		0	0	0
	3		0	0	0

$$\begin{aligned}
 g(C_{(1,4)}, C_3) &= \frac{3}{[(2+1)^{1+2(0.33)} - 2^{1+2(0.33)} - 1^{1+2(0.33)}]} \\
 &= \frac{3}{3^{1.66} - 2^{1.66} - 1^{1.66}} \\
 &= 1.4
 \end{aligned}$$

gabungkan kelompok (1,4) dan kelompok 3 karena memiliki nilai *Goodness Measure* tinggi yaitu 1.4

Didapatkan matriks *link* yang baru :

$$\begin{array}{c} \text{kelompok} \\ 1,3,4 \quad 2,5 \\ \text{kelompok } 1,3,4 \begin{bmatrix} 0 & 0 \\ 2,5 & 0 \end{bmatrix} \end{array}$$

Penggabungan berhenti karena sudah tidak ada *link* antar kelompok atau obyek pengamatan.

4.5 Penerapan Algoritma Kedalam Program Komputer

Program komputer dibuat dengan menggunakan paket program S-Plus.

Algoritma yang tersusun pada algoritma *ROCK* dijabarkan dalam 10 program yang saling terkait (lampiran 1). Program 2.a sampai dengan program 2.h merupakan sub program dari program utama (program 1).

- Program utama merupakan implementasi dari algoritma *ROCK* untuk obyek – obyek pengamatan pada analisis kelompok.
- Program 2.a untuk menentukan similaritas masing – masing kelompok atau obyek pengamatan.
- Program 2.b untuk menentukan *neighbors* masing – masing kelompok atau obyek pengamatan.
- Program 2.c untuk menentukan matriks *link*.
- Program 2.d untuk menentukan jumlah anggota kelompok.
- Program 2.e untuk menentukan *Goodness Measure* dari tiap pasang kelompok dalam bentuk matrik.

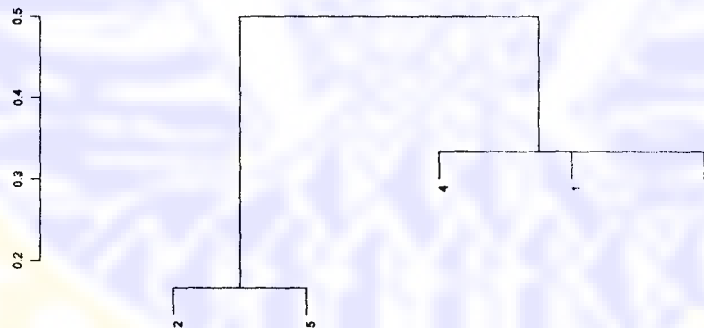
- Program 2.f untuk menentukan obyek pengamatan atau kelompok yang bergabung.
- Program 2.g untuk menentukan matriks *link* yang baru.
- Program 2.h untuk menentukan penggabungan kelompok.

4.6 Analisis data

1. Data I

Proses pengelompokan data I dengan menggunakan metode hirarki konvensional yaitu : keterkaitan tunggal (*single linkage*) dengan terlebih dahulu menghitung ukuran ketidakmiripannya. Penghitungan Ukuran ketidak miripan dilakukan dengan menggunakan menggunakan persamaan : $d = 1 - s$; s adalah ukuran kemiripan.

Kemudian diterapkan dalam program S-Plus dan didapatkan bentuk dendogram. Pada gambar 4.1



Gambar 4.1 Dendogram dari data I dengan metode keterkaitan tunggal (*single linkage*).

Dari dendogram pengelompokan dapat dikelompokkan menjadi satu kelompok yang beranggotakan seluruh obyek. Pengelompokan dengan dua kelompok merupakan kelompok terbaik yang dapat dibentuk dengan

anggota kelompok pertama adalah obyek (2) dan (5). Untuk anggota kelompok kedua adalah obyek (1), (3), dan obyek (4).

Hasil pengelompokan dengan algoritma *ROCK* untuk nilai $\theta = 0.34$ sampai $\theta = 0.5$ paling kecil didapatkan dua kelompok. Masing – masing kelompok memiliki anggota yang sama dengan pengelompokan hirarki konvensional (*single linkage*).

Tabel 4.2 Hasil pengelompokan data I dengan menggunakan algoritma hirarki konvensional dan algoritma *ROCK*.

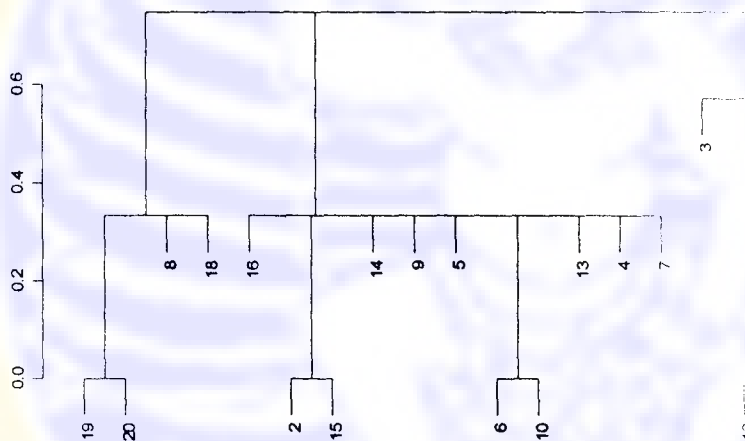
Algoritma pengelompokan	Kelompok	Obyek yang bergabung	Ciri – ciri
Algoritma hirarki konvensional (<i>single linkage</i>).	1	2, 5	Lelaki, tinggi yang hampir sama, warna rambut, warna mata, dan berat sama-sama diatas 150 lb kecuali tangan yang aktif
	2	1, 3, dan 4	Tinggi yang hampir sama, tangan yang aktif, untuk berat badan, jenis kelamin sama kecuali obyek 3
Algoritma <i>ROCK</i>	1	2,5	Lelaki, tinggi yang hampir sama, warna rambut, warna mata, dan berat sama-sama diatas 150 lb kecuali tangan yang aktif
	2	1, 3 dan 4	Tinggi yang hampir sama, tangan yang aktif, untuk berat badan, jenis kelamin sama kecuali obyek 3

Obyek (2) dan (5) memiliki kesamaan pada jenis kelamin, warna rambut, warna mata, dan tinggi badan yang hampir sama serta berat badan yang sama – sama diatas 150 lb tetapi berbeda dalam hal tangan yang aktif. Kelompok kedua yang beranggotakan obyek (1), (3), dan (4) memiliki kesamaan pada pada tangan yang aktif, tinggi badan yang hampir sama, jenis kelamin kecuali obyek (3) tiga (lelaki), , berat badan yang sama – sama dibawah 150 lb kecuali obyek (3), warna rambut kecuali obyek (4) tetapi berbeda dalam hal warna mata.

Hasil pengelompokan dengan algoritma *ROCK* untuk nilai θ dibawah 0.34 pengelompokan baru dapat dijadikan satu kelompok. Untuk nilai $\theta = 0.51$ sampai $\theta = 0.71$ paling kecil didapatkan empat kelompok.

2. Data II

Proses pengelompokan data II dengan menggunakan metode hirarki konvensional yaitu : keterkaitan tunggal (*single linkage*) dapat dibentuk dendogram. Pada gambar 4.2



Gambar 4.2 Dendogram obyek – obyek pengamatan dengan menggunakan metode keterkaitan tunggal (*single linkage*) pada data II.

Pengelompokan pada data II dengan menggunakan metode keterkaitan tunggal (*single linkage*) berdasarkan dendogram diatas didapatkan tiga kelompok yang merupakan pengelompokan terbaik yang dapat dibentuk. Kelompok pertama dengan anggota kelompok adalah obyek (1), (3), (11), (12), dan obyek (17). Kelompok kedua dengan

anggota kelompok adalah obyek (2), (4), (5), (6), (9), (10), (13), (14), (15) dan obyek (16). Kelompok ketiga adalah obyek (8), (18), (19) dan obyek (20).

Pengelompokan pada data II menggunakan algoritma *ROCK* dengan nilai $\theta = 0.26$ samapi dengan $\theta = 0.42$ pengelompokan paling kecil hanya bisa dibentuk menjadi tiga kelompok dengan anggota kelompok sama dengan pada pengelompokan hirarki konvensional (*single linkage*).

Tabel 4.3 Hasil pengelompokan data II dengan menggunakan algoritma hirarki konvensional dan algoritma *ROCK*.

Algoritma pengelompokan	Kelompok	Obyek yang bergabung	Ciri – ciri
Algoritma <i>ROCK</i>	1	1, 3, 11, 12, dan 17	Engsize (50), Cylind (8), For (2), Combrate(104 -110), Weight (4000) kecuali obyek 3 hanya memiliki kesamaan pada Combrate.
	2	2, 4, 5, 6, 7, 9, 10, 13 14 dan 15	Engsize (15-24), Cylind (4), For (1,2), Combrate(64 -84), Weight (2000-2750)
	3	8, 18, 19 dan 20	Engsize (28-20), Cylind (6), For (1), Combrate(93 -97), Weight (3500)
Algoritma hirarki konvensional (<i>complete linkage</i>)	1	1, 3, 11, 12, dan 17	Engsize (50), Cylind (8), For (2), Combrate(104 -110), Weight (4000) kecuali obyek 3 hanya memiliki kesamaan pada Combrate.
	2	2, 4, 5, 6, 7, 9, 10, 13 14 dan 15	Engsize (15-24), Cylind (4), For (1,2), Combrate(64 -84), Weight (2000-2750)
	3	8, 18, 19 dan 20	Engsize (28-20), Cylind (6), For (1), Combrate(93 -97), Weight (3500)

Kelompok satu adalah **Mobil** : *Pontiac Paris, Buick Century, Cadillac Fleetw, Ford Mustang* dan *Mercury Capri* memiliki kesamaan dalam hal **Engsize**(ukuran mesin) : 50, **Cylind**(banyaknya silinder) : sebanyak 8, **Combrate**(tingkat bahan bakar yang dibutuhkan) : antara

104 sampai 110, **Weight**(berat) : 4000, **For** : dibuat diamerika bagaian selatan kecuali untuk mobil *Buick Century* hanya memiliki kesamaan pada tingkat bahan baker yang dibutuhkan.

Kelompok kedua adalah **Mobil** : *Honda Civic, Subaru GL, Volvo 740 GLE, Plymouth Carravel, Honda Accord, Plymouth Horizon, Chrysler Daytona, Toyota Celica, Ford Escort, Toyota Tarcel*. Kebanyakan memiliki kesamaan dalam hal **Engsize**(ukuran mesin) : 15 sampai 24, **Cylind**(banyaknya silinder) : sebanyak 4, **Combrate**(tingkat bahan baker yang dibutuhkan) : 64 sampai 84, **Weight**(berat) : 2000 samapi 2750, **For** : dibuat diamerika bagaian selatan dan luar amerika.

Kelompok ketiga adalah **Mobil** : *Chev Camaro Toyota Cressida, Nissan 300 zx, Nisan Maxima* memiliki kesamaan dalam hal **Engsize**(ukuran mesin) : 28 – 20, **Cylind**(banyaknya silinder) : sebanyak 6, **Combrate**(tingkat bahan baker yang dibutuhkan) : 93 – 97, **Weight**(berat) : 3500, **For** : dibuat diluar amerika selatan.

Pengelompokan dengan menggunakan algoritma *ROCK* dengan nilai $\theta = 0.43$ sampai dengan $\theta = 0.66$ pengelompokan paling kecil hanya bisa dibentuk menjadi empat kelompok. Kelompok pertama adalah obyek (1), (11), (12), dan obyek (17). Kelompok kedua adalah obyek (2), (4), (5), (6), (9), (10), (13), (14), (15) dan obyek (16). Kelompok ketiga adalah obyek (3) saja. Kelompok keempat adalah obyek (8), (18), (19) dan obyek (20). Pengelompokan ini merupakan kelompok terbaik berdasarkan algoritma *ROCK*.

Pengelompokan dengan menggunakan nilai $\theta = 0.67$ sampai $\theta = 0.99$ didapatkan paling kecil 16 kelompok sedangkan untuk $\theta < 0.26$ pengelompokan baru bisa dijadikan satu kelompok.

3. Data III

Proses pengelompokan data III dengan algoritma *ROCK* menggunakan $\theta = 0.7$ didapatkan hasil pengelompokan sebanyak tiga kelompok (lihat lampiran 5). Untuk $\theta = 0.7$ pengelompokan berhenti pada tiga kelompok dan tidak bisa dijadikan menjadi satu kelompok hal ini disebabkan sudah tidak ada *link* antar kelompok.

Tabel 4.4 Hasil pengelompokan data III dengan menggunakan algoritma *ROCK*.

Algoritma Pengelompokan	Kelompok	Obyek yang bergabung
Algoritma <i>ROCK</i>	1	1 2 3 4 8 9 10 11 12 13 14 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 37 41 42 44 46 47 48 52 53 58 64 69 71 75 78 83 85 106 110 112 115 116 117 119 111 120 121 124 128 131 186 190 195
	2	5 6 7 15 31 32 33 34 35 36 38 39 40 43 45 49 50 51 54 55 56 57 59 60 61 62 63 65 66 67 68 70 72 73 74 76 77 79 80 81 82 84 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 107 108 109 113 114 118 122 123 125 126 127 129 130 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 179 180 181 182 183 184 187
	3	185 188 189 191 192 193 194 196 197 198 199 200

Pengelompokan data III dengan algoritma *ROCK* menggunakan $\theta = 0.7$ dan $k = 3$ dianggap yang terbaik dilihat dari obyek yang bergabung bila dibandingkan dengan nilai θ dan k yang lainnya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil penulisan skripsi ini dapat disimpulkan bahwa :

1. Algoritma *ROCK* digunakan untuk mengelompokkan data kategorik berdasarkan pada *neighbors* dan *link* antar kelompok atau obyek pengamatan.
2. Jumlah kelompok pada pengelompokan dengan algoritma *ROCK* tergantung dari k kelompok yang diinputkan dan *link* antar kelompok atau obyek pengamatan.
3. Pada data I dengan menggunakan $\theta = 0.5$ pengelompokan menghasilkan dua kelompok. Data II dengan $\theta = 0.26$ sampai $\theta = 0.42$ paling kecil didapatkan tiga kelompok. Data III dengan menggunakan $\theta = 0.7$ dan $k = 3$ pengelompokan didapatkan tiga kelompok

5.2 SARAN

Adapun saran yang perlu disampaikan pada penulisan ini adalah :

Untuk penelitian berikutnya, penelitian ini bisa menjadi bahan masukan untuk mengembangkan metode lain dalam pengelompokan data berjenis kategori.

DAFTAR PUSTAKA

- Agresti, Alan, 1990. *Categorical Data Analysis*. John Wiley & Sons, New York
- Anonim, 1993a. *S-Plus Programmer's Manual*. MathSoft, Seattle.
- Anonim, 1993b. *S-Plus : Guide To Statistical And Mathematical Analysis*. MathSoft, Seattle.
- Bryan F.J. Manly, 1986. *Multivariate Statistical Methods A Primer*. Chapman and Hall, London
- Chatfield Christopher and Collins Alexander J, 1980. *Introduction to Multivariate Analysis*, Chapman and Hall, London.
- Everitt B., 1994. *Cluster Analysis*. Heinemann Educational Books, London
- Guha, S., Rajeev Rastogi, dan Kyuseok Shim, 2000. ROCK : A robust clustering algorithm for categorical attributes. *Jurnal information system* Vol. 25, No. 5 pp 354-366. <http://citeseer.ist.psu.edu/126378.html>.
- Halkidi, M., Yannis Batistakis, Michalis Vazirgiannis, 2001. On Clustering Validation Techniques. *Journal Of Intelligent Information Systems*. <http://citeseer.nj.nec.com/513619.html>.
- Jobson, J.D., 1992, *Applied Multivariate Data Analysis*, Springer-Verlag Inc, New York
- Johnson, J.D., 1982, *Applied Multivariate Statistic Analysis*, Richard.A, University of Wisconsin-Madison., New Jersey.
- Newman, D.J., Hettich, S., Blake, C.L. dan Merz, C.J., 1998. UCI Repository of machine learning databases <http://www.ics.uci.edu/~mlearn/MLRepository.html>.
- Sharma, S., 1996. *Applied Multivariate Techniques*. John Wiley & Sons Inc, New York.
- Venables, W.N. and Ripley, B.D., 1994, *Modern Applied Statistics With S-Plus*, Springer-Verlag Inc, New York.

Lampiran 1.**Daftar program****1. Program Utama**

Program untuk mengelompokkan obyek – obyek pengamatan dengan algoritma *ROCK* pada analisis kelompok.

```

prog.utama.rock<-function(x)
{
  cat("Masukkan Jumlah Kelompok Yg diinginkan ")
  k <- scan("", n = 1)
  cat("Masukkan Nilai teta ")
  teta <- scan("", n = 1)
  if(teta < 0 || teta > 1)
    prog.utama(x)
  else {
    now <- proc.time()
    x <- as.matrix(x)
    n <- nrow(x)
    M <- link(x, teta)
    pos <- c(1:n)
    repeat {
      G <- matrik.g(M, pos, teta)
      O <- obyek.yg.bergabung(G)
      M <- link.baru(M, O)
      pos <- penggabungan.kelompok(O, pos)
      if(max(M) == 0 || max(pos) == k) {
        break
      }
      pos
    }
    maks <- max(pos)
    for(i in 1:maks) {
      cat("Anggota Kelompok ", i, " : ")
      for(j in 1:n) {
        if(pos[j] == i)
          cat(j, " ")
      }
      cat("\n")
    }
    time <- proc.time() - now
    cat("Waktu Penyelesaian Program : ", time, " detik", "\n")
  }
}

```

2. Sub program

a. Menentukan similaritas masing – masing kelompok atau obyek pengamatan

```
sim<-function(x, y)
{
  x <- as.vector(x)
  y <- as.vector(y)
  m <- length(x)
  irisan <- 0
  gabungan <- 0
  for(i in 1:m) {
    if(x[i] == y[i]) {
      irisan <- irisan + 1
      gabungan <- gabungan + 1
    }
    else gabungan <- gabungan + 2
  }
  similar <- irisan/gabungan
  return(similar)
}
```

b. menentukan *neighbour* masing – masing kelompok atau obyek pengamatan.

```
neighbour<-function(x, teta)
{
  x <- as.matrix(x)
  n <- nrow(x)
  nbrlist <- matrix(0, n, n)
  for(i in 1:n) {
    for(j in 1:n) {
      if(sim(x[i, ], x[j, ]) >= teta)
        nbrlist[i, j] <- 1
    }
  }
  return(nbrlist)
}
```

c. Menentukan matriks *link*.

```
link<-function(x, teta)
{
  x <- as.matrix(x)
  n <- nrow(x)
  bantu <- neighbour(x, teta)
  matlink <- matrix(0, n, n)
  for(i in 1:n) {
```

```

for(j in 1:n) {
  for(k in 1:n) {
    if(j != k) {
      if(bantu[i, j] == 1 && bantu[i, k] == 1)
        matlink[j, k] <- matlink[j, k] + 1
    }
  }
}
return(matlink)
}

```

d. Menentukan jumlah anggota kelompok.

```

jml.ang.kelompok<-function(obyek, poskelompok)
{
  byk.kelompok <- max(poskelompok)
  anggota.kelompok <- c(rep(0, byk.kelompok))
  for(i in 1:byk.kelompok) {
    for(j in 1:obyek) {
      if(poskelompok[j] == poskelompok[i])
        anggota.kelompok[i] <- 1 + anggota.kelompok[i]
    }
  }
  return(anggota.kelompok)
}

```

e. Menentukan *Goodness Measure* dari tiap pasang kelompok.

```

matrik.g<-function(x, y, teta)
{
  x <- as.matrix(x)
  n <- nrow(x)
  n.kelompok <- jml.ang.kelompok(n, y)
  f <- (1 - teta)/(1 + teta)
  p <- 1 + 2 * f
  matg <- matrix(0, n, n)
  for(i in 1:n) {
    for(j in 1:n) {
      matg[i, j] <- x[i, j]/((n.kelompok[i] + n.kelompok[j])^p
- (n.kelompok[i])^p - (n.kelompok[j])^p)
    }
  }
  return(matg)
}

```


f. Menentukan obyek pengamatan atau kelompok yang bergabung.

```

obyek.yg.bergabung<-function(x)
{
  x <- as.matrix(x)
  nbar <- nrow(x)
  par <- max(x)
  kelompok <- c(rep(0, 2))
  for(i in 1:nbar) {
    for(j in 1:nbar) {
      if(x[i, j] == par) {
        if(i > j) {
          kelompok[1] <- j
          kelompok[2] <- i
        }
        else {
          kelompok[1] <- i
          kelompok[2] <- j
        }
      }
    }
  }
  return(kelompok)
}

```

g. Menentukan matriks *link* yang baru.

```

link.baru<-function(x, k)
{
  x <- as.matrix(x)
  k <- as.vector(k)
  nbar <- nrow(x)
  for(i in 1:nbar) {
    for(j in 1:nbar) {
      if(i != j) {
        if(j == k[1] || j == k[2])
          x[i, j] <- x[i, k[1]] + x[i, k[2]]
        if(i == k[1] || i == k[2])
          x[i, j] <- x[k[1], j] + x[k[2], j]
      }
    }
  }
  x <- x[ - k[2], - k[2]]
  return(x)
}

```

h. Menentukan penggabungan kelompok

```
penggabungan.kelompok<-function(x, pos)
{
  x <- as.vector(x)
  pos <- as.vector(pos)
  n <- length(pos)
  for(i in 1:n) {
    if(pos[i] == x[2])
      pos[i] <- x[1]
  }
  for(i in 1:n) {
    if(pos[i] > x[2]) {
      pos[i] <- pos[i] - 1
    }
  }
  return(pos)
}
```

Lampiran 2.**Data I. Tentang karakteristik individu**

	Height	Weight	Eye colour	Hair colour	Handedness	Gender
individu 1	68 in	140 lb	green	blond	right	female
individu 2	73 in	185 lb	brown	brown	right	male
individu 3	67 in	165 lb	blue	blond	right	male
individu 4	64 in	120 lb	brown	brown	right	female
individu 5	76 in	210 lb	brown	brown	left	male

(Sumber : Johnson, 1975)

Pendefinisian variabel $X_1, X_2, X_3, X_4, X_5, X_6$ dalam bentuk *binary* sebagai

berikut :

$$X_1 = \begin{cases} 1 & \text{height} \geq 72 \text{ in} \\ 0 & \text{height} < 72 \text{ in} \end{cases}$$

$$X_4 = \begin{cases} 1 & \text{blond hair} \\ 0 & \text{brown hair} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{weight} \geq 150 \text{ lb} \\ 0 & \text{weight} < 150 \text{ lb} \end{cases}$$

$$X_5 = \begin{cases} 1 & \text{right handed} \\ 0 & \text{left handed} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{brown eyes} \\ 0 & \text{yang lain} \end{cases}$$

$$X_6 = \begin{cases} 1 & \text{female} \\ 0 & \text{male} \end{cases}$$

Lampiran 3.**Data II. Tentang observasi terhadap jenis mobil**

	Engsize	Cylind	Combrate	Weight	For
Pontiac Paris	5	3	4	5	0
Honda Civic	1	1	1	1	1
Buick Century	4	2	4	3	0
Subaru GL	1	1	1	2	1
Volvo 740 GLE	2	1	2	3	1
Plymouth Caravel	2	1	2	3	0
Honda Accord	1	1	2	2	1
Chev Camaro	3	2	3	4	0
Plymouth Horizon	2	1	2	2	0
Chrysler Daytona	2	1	2	3	0
Cadillac Fleetw	4	3	4	5	0
Ford Mustang	5	3	4	4	0
Toyota Celica	2	1	2	2	1
Ford Escort	1	1	2	2	0
Toyota Tarcel	1	1	1	1	1
Toyota Camry	2	1	1	2	1
Mercury Capri	5	3	4	4	0
Toyota Cressida	3	2	3	4	1
Nissan 300 zx	3	2	4	4	1
Nissan Maxima	3	2	4	4	1

(Sumber : Jobson, 1992)

Engsize: 15 – 18 = 1; 20 – 24 = 2; 28 – 30 = 3; 38 – 41 = 4; 50 = 5

(ukuran mesin)

Cylind : 4, 6, 8 menjadi 1, 2, 3

(banyaknya silinder)

Combrate : 64 – 71 = 1; 74 – 84 = 2; 93 – 97 = 3; 104 – 110 = 4

(tingkat bahan bakar yang dibutuhkan)

Weight : 2000-2250 = 1; 2500-2750 = 2; 3000 = 3; 3500 = 4; 4000 = 5.

(berat)

For : 1 = Buatan luar amerika selatan

2 = Buatan amerika selatan

Lampiran 4

Data III. Tentang ciri – ciri jamur

Obyek	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	X ₁₃	X ₁₄	X ₁₅	X ₁₆	X ₁₇	X ₁₈	X ₁₉	X ₂₀	X ₂₁	X ₂₂
1	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	4	5
2	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	5
3	4	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
4	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
5	4	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
6	3	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
7	3	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	4	1
8	6	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
9	6	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
10	6	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
11	3	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
12	4	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
13	4	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
14	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	4	5
15	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	1	1
16	6	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
17	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	1
18	3	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
19	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
20	4	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
21	3	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
22	3	1	1	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
23	4	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	4	1
24	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
25	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	1
26	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	1
27	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	5
28	3	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	1
29	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
30	3	4	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	1
31	4	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	1	1
32	3	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
33	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
34	4	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
35	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
36	3	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
37	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
38	3	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	1	1

	X ₂₂	X ₂₁	X ₂₀	X ₁₉	X ₁₈	X ₁₇	X ₁₆	X ₁₅	X ₁₄	X ₁₃	X ₁₂	X ₁₁	X ₁₀	X ₉	X ₈	X ₇	X ₆	X ₅	X ₄	X ₃	X ₂	X ₁	Obyek	
39	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
40	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
42	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
43	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
44	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
45	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
46	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
47	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
48	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
49	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
50	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
51	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
52	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
53	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
54	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
55	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
56	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
57	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
58	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
59	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
60	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
61	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
62	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
63	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
64	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
65	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
66	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
67	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
68	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
69	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
70	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
71	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
72	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
73	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
74	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
75	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
76	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	4
77	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
78	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
79	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3
80	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3

Obyek	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	X ₁₃	X ₁₄	X ₁₅	X ₁₆	X ₁₇	X ₁₈	X ₁₉	X ₂₀	X ₂₁	X ₂₂
81	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
82	3	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
83	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	1
84	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1
85	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	5
86	4	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	1	1
87	4	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
88	4	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
89	3	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
90	3	4	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	4	1
91	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
92	4	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
93	4	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
94	3	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
95	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
96	3	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
97	3	4	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	1	1
98	4	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
99	3	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	1	1
100	4	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
101	3	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
102	3	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
103	3	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
104	4	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
105	3	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
106	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	4	5
107	3	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1
108	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
109	3	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
110	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
111	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
112	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
113	4	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
114	3	1	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
115	3	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
116	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
117	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
118	4	4	4	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
119	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
120	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
121	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
122	4	4	4	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1

Obyek	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	X ₁₃	X ₁₄	X ₁₅	X ₁₆	X ₁₇	X ₁₈	X ₁₉	X ₂₀	X ₂₁	X ₂₂
81	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
82	3	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
83	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	1
84	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1
85	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	5
86	4	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	1	1
87	4	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
88	4	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
89	3	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
90	3	4	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	4	1
91	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
92	4	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
93	4	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
94	3	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
95	4	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
96	3	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
97	3	4	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
98	4	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
99	3	4	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	1	1
100	4	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
101	3	1	1	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
102	3	4	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
103	3	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
104	4	1	1	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
105	3	4	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
106	4	3	1	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	4	5
107	3	1	1	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1
108	3	1	1	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
109	3	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
110	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
111	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
112	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
113	4	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
114	3	1	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
115	3	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
116	6	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
117	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	6	5
118	4	4	4	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	1	4	1
119	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	6	5
120	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
121	4	1	4	2	7	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
122	4	4	4	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1

Obyek	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	X ₁₃	X ₁₄	X ₁₅	X ₁₆	X ₁₇	X ₁₈	X ₁₉	X ₂₀	X ₂₁	X ₂₂
165	4	1	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	1	1
166	4	4	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
167	4	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
168	4	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
169	3	1	4	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	1	1
170	3	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
171	4	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	4	1
172	3	1	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
173	4	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
174	3	4	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	4	1
175	3	4	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
176	3	4	4	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	4	1
177	4	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	1	1	1
178	4	4	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	2	4	1
179	3	1	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
180	4	1	4	2	7	3	2	1	1	2	4	1	1	8	8	1	3	2	2	1	1	1
181	3	4	4	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	1	4	1
182	3	1	4	2	7	3	2	1	1	2	4	1	4	8	8	1	3	2	2	2	4	1
183	3	4	4	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	4	1
184	3	1	4	2	7	3	2	1	1	2	4	4	1	8	8	1	3	2	2	2	1	1
185	3	4	10	1	1	3	1	1	1	1	2	4	4	8	8	1	3	2	6	2	3	1
186	3	3	9	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	5	5
187	4	1	9	2	7	3	2	1	1	2	4	4	4	8	8	1	3	2	2	2	1	1
188	1	4	10	1	1	3	1	1	1	1	2	4	4	8	8	1	3	2	6	2	4	3
189	1	3	10	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	2	4	3
190	3	3	9	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	2	4	5
191	1	4	10	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	4	3
192	3	3	10	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	4	3
193	3	4	9	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	4	1
194	1	4	9	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	2	3	3
195	3	4	9	1	8	3	1	2	1	1	4	4	4	8	8	1	3	2	6	1	5	5
196	3	4	10	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	3	1
197	1	3	10	1	1	3	1	1	1	1	2	4	4	8	8	1	3	2	6	2	4	3
198	1	4	10	1	1	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	4	1
199	1	4	9	1	2	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	4	1
200	3	4	10	1	1	3	1	1	1	1	2	4	4	8	8	1	3	2	6	1	3	3

(Sumber : UCI machine learning repository (www.ics.edu/~mlern/MLRepository))

Variabel-variabel yang diamati antara lain :

- cap-shape (X₁)** : 1=bell=b, 2=conical=c, 3=convex=x, 4=flat=f,
5=knobbed=k, 6=sunken=s
- cap-surface (X₂)** : 1=fibrous=f, 2=grooves=g, 3=scaly=y, 4=smooth=s
- cap-color (X₃)** : 1=rown=n, 2=buff=b, 3=cinnamon=c, 4=gray=g,
5=green=r, 6=pink=p, 7=purple=u, 8=red=e,
9=white=w, 10=yellow=y
- bruises? (X₄)** : 1=bruises=t, 2=no=f
- odor (X₅)** : 1=almond=a, 2=anise=l, 3=creosote=c, 4=fishy=y,
5=foul=f, 6=musty=m, 7=none=n, 8=pungent=p,
9=spicy=s
- gill-attachment (X₆)** : 1=attached=a, 2=descending=d, 3=free=f,
4=notched=n
- gill-spacing (X₇)** : 1=close=c, 2=crowded=w, 3=distant=d
- gill-size (X₈)** : 1=broad=b, 2=narrow=n
- gill-color (X₉)** : 1=black=k, 2=brown=n, 3=buff=b, 4=chocolate=h,
5=gray=g, 6=green=r, 7=orange=o, 8=pink=p,
9=purple=u, 10=red=e, 11=white=w, 12=yellow=y
- stalk-shape (X₁₀)** : 1=enlarging=e, 2=tapering=t
- stalk-root (X₁₁)** : 1=bulbous=b, 2=club=c, 3=cup=u, 4=equal=e,
5=rhizomorphs=z, 6=rooted=r, 0=missing=?
- stalk-surface-above-ring (X₁₂)** : 1=ibrous=f, 2=scaly=y, 3=silky=k,
4=smooth=s
- stalk-surface-below-ring (X₁₃)** : 1=ibrous=f, 2=scaly=y, 3=silky=k,
4=smooth=s
- stalk-color-above-ring (X₁₄)** : 1=brown=n, 2=buff=b, 3=cinnamon=c,
4=gray=g, 5=orange=o, 6=pink=p,
7=red=e, 8=white=w, 9=yellow=y
- stalk-color-below-ring (X₁₅)** : 1=brown=n, 2=buff=b, 3=cinnamon=c,
4=gray=g, 5=orange=o, 6=pink=p,
7=red=e, 8=white=w, 9=yellow=y
- veil-type (X₁₆)** : 1=partial=p, 2=universal=u
- veil-color (X₁₇)** : 1=brown=n, 2=orange=o, 3=white=w, 4=yellow=y
- ring-number (X₁₈)** : 0=none=n, 1=one=o, 2=two=t
- ring-type (X₁₉)** : 1=cobwebby=c, 2=evanescent=e, 3=flaring=f,
4=large=l, 5=none=n, 6=pendant=p,
7=sheathing=s, 8=zone=z

spore-print-color (X₂₀): 1=black=k, 2=brown=n, 3=buff=b,
4=chocolate=h, 5=green=r, 6=orange=o,
7=purple=u, 8=white=w, 9=yellow=y

population (X₂₁) : 1=abundant=a, 2=clustered=c, 3=numerous=n,
4=scattered=s, 5=several=v, 6=solitary=y

habitat (X₂₂) : 1=grasses=g, 2=leaves=l, 3=meadows=m,
4=paths=p, 5=urban=u, 6=waste=w, 7=woods=d

LAMPIRAN 5**DATA I (Johnson, 1975)****Menginput data sebagai matrik :**

```
> attach(DS43)
> aa<-as.matrix(DS43)
> aa
  V1 V2 V3 V4 V5 V6
1 0 0 0 1 1 1
2 1 1 1 0 1 0
3 0 1 0 1 1 0
4 0 0 1 0 1 1
5 1 1 1 0 0 0
```

Menentukan *neighbors* dalam bentuk matrik

```
> neighbour(aa,0.5)
  [,1] [,2] [,3] [,4] [,5]
[1,] 1 0 1 1 0
[2,] 0 1 0 0 1
[3,] 1 0 1 0 0
[4,] 1 0 0 1 0
[5,] 0 1 0 0 1
```

Menentukan *link* antar kelompok dalam bentuk matrik

```
> link(aa,0.5)
  [,1] [,2] [,3] [,4] [,5]
[1,] 0 0 2 2 0
[2,] 0 0 0 0 2
[3,] 2 0 0 1 0
[4,] 2 0 1 0 0
[5,] 0 2 0 0 0
> bb<-link(aa,0.5)
```

Menentukan jumlah anggota kelompok

```
> z<-c(1:5)
> z
[1] 1 2 3 4 5
> jml.ang.kelompok(5,z)
[1] 1 1 1 1 1
```

Menentukan *Goodness Measure* dalam bentuk matrik

```
> matrik.g(bb,z,0.5)
  [,1] [,2] [,3] [,4] [,5]
[1,] 0.000000 0.000000 1.7024144 1.7024144 0.000000
[2,] 0.000000 0.000000 0.0000000 0.0000000 1.702414
[3,] 1.702414 0.000000 0.0000000 0.8512072 0.000000
[4,] 1.702414 0.000000 0.8512072 0.0000000 0.000000
[5,] 0.000000 1.702414 0.0000000 0.0000000 0.000000
> cc<-matrik.g(bb,z,0.5)
```


DATA III (www.ics.edu/~mlearn/MLRepository).

```
> attach(DS42)
```

```
> aa<-as.matrix(DS42)
```

```
> prog.utama.rock(aa)
```

Masukkan Jumlah Kelompok Yg diinginkan 1: 3

Masukkan Nilai teta 1: 0.7

Anggota Kelompok 1 : 1 2 3 4 8 9 10 11 12 13 14 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 37 41 42 44 46 47 48 52 53 58 64 69 71 75 78 83 85 106 110 111 112
115 116 117 119 120 121 124 128 131 186 190 195

Anggota Kelompok 2 : 5 6 7 15 31 32 33 34 35 36 38 39 40 43 45 49 50 51 54 55 56
57 59 60 61 62 63 65 66 67 68 70 72 73 74 76 77 79 80 81 82 84 86 87 88 89 90
91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 107 108 109 113 114 118 122
123 125 126 127 129 130 132 133 134 135 136 137 138 139 140 141 142 143 144 145
146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165
166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184
187

Anggota Kelompok 3 : 185 188 189 191 192 193 194 196 197 198 199 200

Waktu Penyelesaian Program : 1271.36 detik

```
>
```


Menentukan obyek yang bergabung

```
> obyek.yg.bergabung(cc)
[1] 2 5
> k<-obyek.yg.bergabung(cc)
```

Mendapatkan *link* baru dalam bentuk matrik

```
> link.baru(bb,k)
      [,1] [,2] [,3] [,4]
[1,]  0  0  2  2
[2,]  0  0  0  0
[3,]  2  0  0  1
[4,]  2  0  1  0
```

Menentukan penggabungan kelompok

```
> penggabungan.kelompok(k,z)
[1] 1 2 3 4 2
```

Hasil pengelompokan

```
> prog.utama.rock(aa)
Masukkan Jumlah Kelompok Yg diinginkan 1: 2
Masukkan Nilai teta 1: 0.5
Anggota Kelompok 1 : 1 3 4
Anggota Kelompok 2 : 2 5
Waktu Penyelesaian Program : 0.0300293 detik
>
```

DATA II (Jobson, 1992)

```
> prog.utama.rock(xx)
Masukkan Jumlah Kelompok Yg diinginkan 1: 1
Masukkan Nilai teta 1: 0.6
Anggota Kelompok 1 : 1 11 12 17
Anggota Kelompok 2 : 2 4 5 6 7 9 10 13 14 15 16
Anggota Kelompok 3 : 3
Anggota Kelompok 4 : 8 18 19 20
Waktu Penyelesaian Program : 1.03 detik
>
```