



**APLIKASI ALGORITMA *HEIGHT MAP* UNTUK MENCARI
MAXIMAL INTERSECTIONS PADA DATA TERSENSOR
INTERVAL BIVARIAT**

SKRIPSI

Oleh

**D. Tiara Andani
NIM 131810101058**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017**



**APLIKASI ALGORITMA *HEIGHT MAP* UNTUK Mencari
MAXIMAL INTERSECTIONS PADA DATA TERSENSOR
INTERVAL BIVARIAT**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan studi pada Program Studi Matematika (S-1)
dan mencapai gelar Sarjana Sains

Oleh

**D. Tiara Andani
NIM 131810101058**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017**

PERSEMBAHAN

Puji syukur kehadiran Tuhan Yang Maha Esa, skripsi ini saya persembahkan untuk:

1. kedua orang tua saya, Bapak Sunaryo dan Ibu Yohana yang telah mendoakan dan mendukung serta memberikan kasih sayang;
2. keluarga besar yang selalu memberi doa dan dukungan;
3. seluruh dosen dan guru sejak sekolah dasar sampai perguruan tinggi, yang telah memberikan ilmu dan membimbing dengan penuh kesabaran;
4. almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMA Negeri 2Tanggul, SMP Negeri 1 Semboro dan SD Negeri Rejoagung 1.
5. teman-teman angkatan 2013 (ATLAS'13) yang selalu membantu, memberi dukungan dan kenangan.

MOTTO

“Demikianlah hendaknya terangmu bercahaya didepan orang, supaya mereka melihat perbuatanmu yang baik dan memuliakan Bapamu yang disorga.”

(Matius 5: 16)^{*)}



^{*)} Alkitab Matius 5 : 16

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : D. Tiara Andani

NIM : 131810101058

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Aplikasi Algoritma *Height Map* untuk Mencari *Maximal Intersections* pada Data Tersensor Interval Bivariat” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Juni 2017
Yang menyatakan,

D. Tiara Andani
NIM 131810101058

SKRIPSI

APLIKASI ALGORITMA *HEIGHT MAP* UNTUK MENCARI *MAXIMAL INTERSECTIONS* PADA DATA TERSENSOR INTERVAL BIVARIAT

Oleh
D. Tiara Andani
NIM. 131810101058

Pembimbing

Dosen Pembimbing Utama : Dr. Mohamat Fatekurohman, S.Si., M.Si

Dosen Pembimbing Anggota : Dian Anggraeni, S.Si., M.Si

PENGESAHAN

Skripsi berjudul “Aplikasi Algoritma *Height Map* Untuk Mencari *Maximal Intersections* Pada Data Tersensor Interval Bivariat” telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji :

Ketua,

Anggota I,

Dr. Mohamat Fatekurohman, S.Si., M.Si

Dian Anggraeni, S.Si., M.Si

NIP. 196906061998031001

NIP. 198202162006042002

Anggota II,

Anggota III,

Prof. Drs. I Made Tirta, M.Sc., Ph.D.

Dr. Alfian Futuhul Hadi, S.Si., M.Si.

NIP. 195912201985031002

NIP. 197407192000121001

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.

NIP. 196102041987111001

RINGKASAN

Aplikasi Algoritma *Height Map* Untuk Mencari *Maximal Intersections* Pada Data Tersensor Interval Bivariat; D. Tiara Andani; 131810101058; 2017; 33 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Kesehatan merupakan salah satu hal terpenting bagi manusia agar dapat bertahan hidup dan melakukan aktivitas. Salah satu faktor terpenting bagi kesehatan manusia adalah sistem kekebalan tubuh. Menurut WHO (2009), HIV (*Human Immunodeficiency Virus*) merupakan sebuah virus yang menyerang sistem kekebalan tubuh manusia. AIDS (*Acquired Immunodeficiency Syndrome*) merupakan tahap selanjutnya dari infeksi HIV.

Salah satu permasalahan yang penting dalam studi tentang HIV/AIDS adalah estimasi distribusi waktu inkubasi. Waktu inkubasi didefinisikan sebagai waktu antara terinfeksi HIV menjadi AIDS. Ketahanan hidup pasien HIV/AIDS dapat dipengaruhi oleh beberapa faktor dan dapat digunakan analisis *survival* untuk menganalisis. Analisis *survival* merupakan analisis statistik yang berfungsi untuk menganalisis data waktu hidup. Adanya data tersensor dalam analisis *survival* mengakibatkan data yang diperoleh relatif sedikit sehingga metode yang digunakan untuk fungsi distribusi tersebut adalah metode *nonparametric*.

Menghitung *Nonparametric Maximum Likelihood Estimator* (NPMLE) dibagi menjadi dua langkah: langkah reduksi parameter dan langkah optimasi. Menurut Maathuis (2005) ada dua algoritma reduksi baru yaitu algoritma *Tree* dan algoritma *Height Map*. Algoritma *Height Map* digunakan untuk mencari *maximal intersections* sedangkan algoritma *Tree* untuk menghitung *clique matrix*.

Pada penelitian kali ini digunakan algoritma *height map* untuk mencari *maximal intersections* pada data tersensor interval bivariat. Data yang digunakan dalam penelitian ini berbentuk persegi panjang pengamatan. Kemudian data persegi panjang tersebut ditransformasikan menjadi bentuk *canonical rectangles* dimana memiliki struktur irisan yang sama. Selanjutnya membangun *height map*

dari bentuk *canonical rectangles* dan didapatkan *maximal intersections* dari himpunan data pengamatan.



PRAKATA

Puji syukur kepada Tuhan Yang Maha Esa, yang telah membimbing dan memberkati sehingga penulis dapat menyelesaikan skripsi yang berjudul ”Aplikasi Algoritma *Height Map* Untuk Mencari *Maximal Intersections* Pada Data Tersensor Interval Bivariat”. Skripsi ini disusun guna memenuhi salah satu syarat untuk menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Dr. Mohamat Fatekurohman, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Dian Anggraeni, S.Si., M.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Prof. Drs. I Made Tirta, M.Sc., Ph.D. dan Dr. Alfian Futuhul Hadi, S.Si, M.Si. selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun demi kesempurnaan skripsi ini;
3. Ahmad Kamsyakawuni, S.Si., M.Kom. selaku Dosen Pembimbing Akademik yang selalu memberikan motivasi dan pengarahan selama penulis menjadi mahasiswa;
4. seluruh Dosen dan Karyawan Jurusan Matematika Fakultas MIPA Universitas Jember;
5. semua pihak yang tidak dapat disebutkan satu per satu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Juni 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBING	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xiv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	4
BAB 2. TINJAUAN PUSTAKA	5
2.1 Analisis <i>Survival</i>	5
2.2 Data Tersensor	6
2.3 <i>Canonical Rectangles</i>	9
2.4 Aljabar Boolean	12
2.4.1 Definisi Aljabar Boolean	12
2.4.2 Aljabar Boolean Dua-Nilai	13
2.4.3 Ekspresi Boolean	14
2.5 Definisi Algoritma	14
2.6 Algoritma <i>Height Map</i>	14

BAB 3. METODE PENELITIAN	16
3.1 Data Penelitian	16
3.2 Langkah-langkah Penelitian	16
BAB 4. HASIL DAN PEMBAHASAN	18
4.1 Hasil	18
4.1.1 Data Bivariat HIV/AIDS	18
4.1.2 Persegi Panjang Pengamatan (<i>Observation Rectangles</i>)	20
4.1.3 Bentuk <i>Canonical Rectangles</i>	23
4.1.4 <i>Height Map</i>	25
4.1.5 Simulasi Program	26
4.2 Pembahasan	29
BAB 5. PENUTUP	31
5.1 Kesimpulan	31
5.2 Saran	31
DAFTAR PUSTAKA	32
LAMPIRAN	34

DAFTAR GAMBAR

	Halaman
2.1 Contoh data tersensor tipe I.....	8
2.2 Contoh data tersensor tipe II.....	8
2.3 Contoh data tersensor tipe III	9
2.4 Contoh <i>Canonical Rectangles</i>	10
2.5 Memisahkan hubungan selama transformasi pengamatan persegi panjang menjadi <i>canonical rectangles</i>	11
2.6 Contoh dari 6 persegi panjang pengamatan dan <i>height map</i>	15
3.1 Skema langkah-langkah penelitian	17
4.1 Persegi panjang pengamatan.....	22
4.2 <i>Canonical Rectangles</i>	24
4.3 <i>Height map</i>	25
4.4 Tampilan awal program.....	26
4.5 Menghapus data	27
4.6 Menambah data.....	28
4.7 <i>Output</i> dari operasi <i>Height Map</i>	28
4.8 <i>Intersection</i> dari 21 data pasien HIV/AIDS.	29

DAFTAR TABEL

	Halaman
2.1 Kaidah Operator Biner	13
2.2 Kaidah Operator Uner	13
4.1 Interval waktu terinfeksi HIV sampai terjangkit AIDS.	19
4.2 Nilai R_1, \dots, R_{21} dari persegi panjang pengamatan.	21
4.3 Batas kiri bawah dan kanan atas R_1, \dots, R_{21} persegi panjang pengamatan.	21
4.4 Nilai R_1, \dots, R_{21} dari <i>canonical rectangles</i>	23
4.5 Batas kiri bawah dan kanan atas R_1, \dots, R_{21} <i>canonical rectangles</i>	24

DAFTAR LAMPIRAN

	Halaman
A. Data pasien HIV/AIDS RSUD Balung, Jember	43
B. Data bivariat pasien HIV/AIDS RSUD Balung, Jember hasil olahan	45
C. Data univariat pasien HIV/AIDS dari awal sampai akhir waktu inkubasi.	46
D. Data univariat pasien HIV/AIDS dari awal HIV sampai akhir AIDS	47
E. Data univariat pasien HIV/AIDS dari awal HIV sampai awal AIDS.....	48
F. <i>Flow chart</i> program aplikasi algoritma <i>height map</i>	49
G. Skrip mencari irisan persegi panjang.	52
H. Skrip mencari irisan terbesar.	53
I. Skrip GUI algoritma <i>Height Map</i>	54
J. <i>Antichains</i>	54
K. Merubah data bivariat pasien HIV/AIDS menjadi data univariat.....	54
L. <i>Maximal Antichains</i> dari data pasien HIV/AIDS RSUD Balung, Jember.....	54

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Kesehatan merupakan salah satu hal terpenting bagi manusia agar dapat bertahan hidup dan melakukan aktivitas. Salah satu faktor terpenting bagi kesehatan manusia adalah sistem kekebalan tubuh. Menurut WHO (2009), HIV (*Human Immunodeficiency Virus*) merupakan sebuah virus yang menyerang sistem kekebalan tubuh manusia. AIDS (*Acquired Immunodeficiency Syndrome*) merupakan tahap selanjutnya dari infeksi HIV. AIDS muncul setelah virus (HIV) menyerang sistem kekebalan tubuh selama beberapa tahun. Sistem kekebalan tubuh menjadi lemah, dan satu atau lebih penyakit dapat timbul. Fungsi sistem kekebalan tubuh dapat diukur dengan jumlah sel CD4. Menurut Murni dkk (2009), jumlah CD4 normal pada orang sehat berkisar antara 500 sampai 1.500. Setelah terinfeksi HIV, jumlah CD4 biasanya terus menurun. Jika jumlah CD4 turun dibawah 200, ini berarti sistem kekebalan cukup rusak dan sudah sampai pada tahap AIDS. Sampai saat ini, masih belum ada obat khusus untuk menyembuhkan infeksi HIV, tetapi ada pengobatan yang bisa memperlambat perkembangan penyakit. Pengobatan ini dapat membuat orang yang terinfeksi HIV dapat bertahan hidup lebih lama.

Salah satu permasalahan yang penting dalam studi tentang HIV/AIDS adalah estimasi distribusi waktu inkubasi. Waktu inkubasi didefinisikan sebagai waktu antara terinfeksi HIV menjadi AIDS. Selama masa inkubasi kekebalan tubuh menjadi lemah. Hal ini yang mengakibatkan pasien HIV/AIDS harus bertahan hidup. Ketahanan hidup pasien HIV/AIDS dapat dipengaruhi oleh beberapa faktor dan dapat digunakan analisis *survival* untuk menganalisis. Analisis *survival* merupakan analisis statistik yang berfungsi untuk menganalisis data waktu hidup. Data yang digunakan dalam analisis *survival* sebaiknya merupakan data tidak tersensor. Namun kenyataannya sering dijumpai kendala-kendala dalam melakukan suatu penelitian yang berhubungan dengan waktu hidup seperti keterbatasan dana, waktu dan tenaga sehingga sulit untuk mendapatkan data yang lengkap, maka dari itu data waktu hidup biasanya merupakan data yang

tak lengkap atau tersensor (Nelson, 1982). Menurut Collett (2003) ada tiga jenis penyensoran yakni sensor kanan, sensor kiri dan sensor interval. Sedangkan menurut Maathuis (2003), waktu terinfeksi HIV merupakan tersensor interval, dan waktu terjangkit AIDS dapat menggunakan tersensor kanan atau tersensor interval. Adanya data tersensor mengakibatkan data yang diperoleh relatif sedikit. Data tersensor juga tidak memenuhi syarat prosedur standar statistika yang digunakan pada analisis data, karena data tersensor biasanya berdistribusi tidak simetris. Model histogram akan semakin miring kekanan sesuai dengan interval waktu dengan jumlah pengamatan terbesar, sehingga tidak ada alasan untuk mengansumsikan bahwa data berdistribusi normal. Metode yang digunakan untuk fungsi distribusi tersebut adalah metode *nonparametric*.

Misalkan diberikan (X, Y) sebagai variabel, dimana X merupakan waktu terinfeksi HIV, Y merupakan waktu terjangkit AIDS dan F_0 sebagai fungsi distribusi bersama. Jika digunakan mekanisme sensor, (X, Y) saling bebas, maka (X, Y) tidak dapat diteliti secara langsung pada (x, y) tetapi pada daerah persegi panjang $R \subset \mathbb{R}^2$ yang berisi (x, y) . R merupakan data persegi panjang pengamatan, jadi data yang digunakan merupakan data persegi panjang pengamatan R_1, \dots, R_n . Dalam menghitung *Nonparametric Maximum Likelihood Estimator* (NPMLE) ada dua langkah: langkah reduksi parameter dan langkah optimasi. Menurut Maathuis (2005) ada dua algoritma reduksi baru yaitu algoritma *Tree* dan algoritma *Height Map*. Algoritma reduksi dibedakan menjadi dua tipe yang mempertimbangkan perhitungan waktu dan ruang:

- Tipe 1: Algoritma reduksi untuk mencari *maximal intersections*.
- Tipe 2: Algoritma reduksi untuk menghitung *clique matrix*.

Digunakan algoritma reduksi baru yaitu algoritma *Height Map* untuk tipe 1 dengan kompleksitas waktu $O(n^2)$ dan algoritma *Tree* untuk tipe 2.

Menurut Gentleman dan Vandal(2002) data yang tersensor interval bivariat dapat di representasikan dengan graf irisan. Betensky dan Finkelstein (1999) juga memberikan algoritma tipe 1 secara sederhana tetapi tidak efisien. Selanjutnya, Maathuis (2005) membandingkan kinerja algoritma reduksi baru dengan algoritma Gentleman dan Vandal (2002), Song (2001), dan Bogaerts dan

Lesaffe (2004) menggunakan data simulasi. Hasil perbandingan kinerja dari algoritma-algoritma tersebut menunjukkan bahwa algoritma reduksi khususnya algoritma *HeightMap* secara signifikan lebih cepat kinerjanya. Sehingga algoritma *HeightMap* dapat dikatakan sebagai algoritma yang sangat cepat dan sederhana untuk menyelesaikan masalah *maximal intersections*.

Berdasarkan penelitian yang telah dilakukan sebelumnya, maka pada penelitian ini akan fokus pada langkah reduksi tipe 1 untuk mencari *maximal intersections* pada persegi panjang pengamatan menggunakan algoritma *height map* yang akan dijelaskan secara rinci. Penelitian ini menggunakan data tersensor interval bivariat pada kasus penyakit HIV/AIDS di RSUD Balung, Jember. Pada penelitian ini juga akan disajikan simulasi program.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan masalah yang diperoleh adalah:

- a. Bagaimana bentuk *canonical rectangles* dari persegi panjang hasil pengamatan?
- b. Bagaimana menggunakan algoritma *Height Map* untuk mencari *maximal intersections* pada data tersensor bivariat?

1.3 Batasan Masalah

Data yang digunakan pada penelitian ini merupakan data bivariat kasus HIV/AIDS yang diambil di RSUD Balung, Jember. Data yang digunakan merupakan data tersensor interval dan tersensor kanan.

1.4 Tujuan

Adapun tujuan dalam penulisan penelitian ini adalah:

- a. Mencari bentuk *canonical rectangles* dari persegi panjang hasil pengamatan.
- b. Menggunakan algoritma *Height Map* untuk mencari *maximal intersections* dari data tersensor bivariat.

1.5 Manfaat

Hasil penelitian ini diharapkan memberi manfaat serta pengetahuan baru dalam bidang statistika maupun kesehatan. Penelitian ini menunjukkan *maximal intersections* pada persegi panjang pengamatan untuk menyelesaikan permasalahan estimasi waktu inkubasi HIV menjadi AIDS. Selain itu mengetahui peluang lamanya waktu inkubasi pasien HIV/AIDS pada waktu yang sama.



BAB 2. TINJAUAN PUSTAKA

2.1 Analisis *Survival*

Analisis *Survival* digunakan untuk mendefinisikan analisis data yang berhubungan dengan waktu, mulai dari waktu awal pengamatan sampai terjadinya suatu peristiwa khusus atau titik akhir (*end point*) (Collett, 2003). Waktu pengamatan yang dicatat bisa dalam hari, minggu, bulan dan tahun. Jangka waktu dari awal pengamatan sampai terjadinya suatu peristiwa khusus atau titik akhir disebut waktu *survival*. Kejadian yang terjadi (*failure event*) dapat berupa kematian, sakit, kambuh dari remisi dan respon terhadap percobaan harus sesuai dengan kejadian yang diteliti (Kleinbaum dan Klen, 2005).

Menurut Murti (1997), ada sejumlah model yang telah dicoba untuk menghubungkan antara faktor resiko kelangsungan hidup dan jangka waktu penaksiran. Pemilihan model tersebut perlu memperhatikan hal-hal berikut:

- a. Bentuk distribusi probabilitas kelangsungan hidup, apakah bersifat *parametric* atau *nonparametric*, sebab tiap penyakit dan keadaan-keadaan lainnya memiliki bentuk distribusi masing-masing.
- b. Apakah faktor resiko yang mendapat perhatian hanya sebuah (univariat) atau majemuk (multivariat).
- c. Ukuran sampel penelitian.
- d. Apakah data mencakup pengamatan tersensor atau tak tersensor.

Dalam menentukan waktu *survival*, terdapat tiga hal yang harus diperhatikan yaitu :

- a. *Time origin or starting point* (titik awal) adalah waktu mulainya suatu penelitian.
- b. *Ending event of interest* (kejadian akhir) adalah kejadian yang menjadi inti dari penelitian
- c. *Measurement scale for the passage of time* (skala ukuran untuk berlalunya waktu).

2.2 Data Tersensor

Data tersensor adalah data yang tercatat ketika memiliki beberapa informasi tentang waktu *survival* individu, tetapi tidak tahu persis waktu *survival* yang sebenarnya. Menurut Lee dan Wang (2003) penyensoran dilakukan untuk memperpendek waktu percobaan, mengukur waktu kegagalan dan kematian suatu individu.

Menurut Kleinbaum dan Klein (2005), pada umumnya data tersensor disebabkan oleh tiga hal, yaitu:

a. *Loss to follow up* (hilang dari pengamatan)

Pengamatan dikatakan *loss to follow up* jika objek yang diamati meninggal, pindah atau menolak untuk berpartisipasi.

b. *Drop out*

Pengamatan dikatakan *drop out* jika perlakuan dihentikan karena alasan tertentu. Misalkan, seorang pasien mengidap penyakit DBD kemudian alergi dengan obat tertentu maka perlakuan dihentikan.

c. *Termination of study* (Pembatasan Waktu)

Pengamatan dikatakan *termination of study* jika penelitian yang dilakukan berakhir sedangkan objek yang diamati belum mencapai kejadian yang diinginkan.

Menurut Collett (2003), terdapat tiga jenis penyensoran yaitu:

a. Sensor kanan (*right censoring*)

Sensor kanan terjadi karena individu belum mengalami suatu peristiwa sampai akhir pengamatan, sedangkan waktu awal dari individu dapat diamati secara penuh. Misalkan suatu individu diamati selama lima tahun dari awal pengamatan, kemudian pada tahun ketiga individu tersebut pindah ke negara lain dan tidak dapat diamati lagi (*lost to follow up*). Individu ini memiliki waktu *survival* dalam penelitian setidaknya dua tahun, sehingga waktu pengamatan individu tersebut dikatakan tersensor kanan.

b. Sensor kiri (*left censoring*)

Sensor kiri terjadi karena waktu awal dari individu tidak dapat teramati pada awal pengamatan, sementara peristiwa dapat diamati secara penuh sebelum penelitian berakhir. Misalkan peneliti mengamati pasien penyakit kanker, peneliti dapat mencatat kejadian tepatnya seseorang tersebut positif kanker di tes pertamanya, namun peneliti tidak memiliki catatan tentang waktu tepatnya seseorang tersebut mulai berpenyakit kanker, maka pasien kanker tersebut tersensor kiri yaitu ketika mengalami kejadian pertama dengan hasil positif kanker.

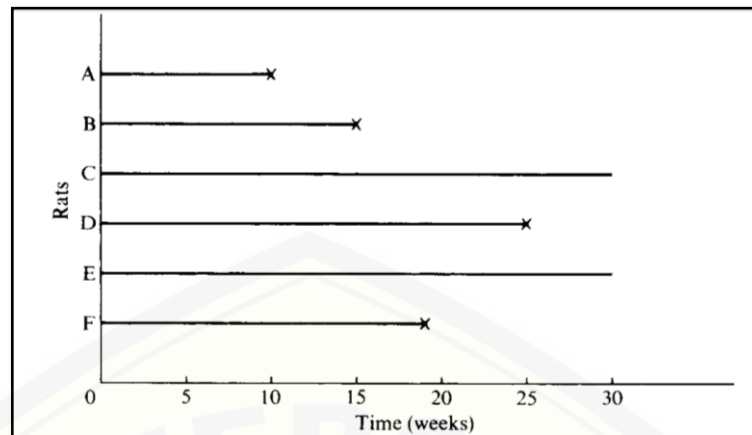
c. Sensor interval (*interval censoring*)

Sensor interval terjadi jika waktu *survival* berada dalam suatu selang tertentu. Misalnya, jika catatan medis menunjukkan bahwa pada usia 45 tahun pasien kanker kondisinya sehat dan belum berpenyakit kanker, kemudian pasien melakukan tes pertama saat berumur 50 tahun dan terdiagnosis terkena penyakit kanker, maka usia saat didiagnosis positif kanker adalah antara 45 dan 50 tahun.

Sedangkan Menurut Lee dan Wang (2003), ada tiga tipe penyensoran yaitu :

a. Tipe I

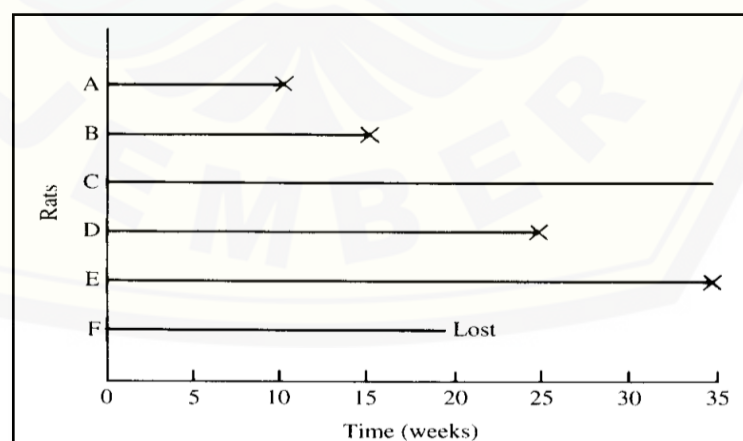
Tipe I adalah tipe penyensoran dimana percobaan akan dihentikan setelah mencapai waktu T yang telah ditentukan untuk mengakhiri semua n individu yang masuk pada waktu yang sama. Terdapat beberapa objek yang mengalami kejadian setelah pengamatan selesai dan beberapa lagi mengalami kejadian diluar waktu yang telah ditetapkan. Sehingga objek tersebut tidak mengalami kejadian hingga akhir, tetapi waktu awal penelitian dapat diamati.



Gambar 2.1 Contoh data tersensor tipe I

b. Tipe II

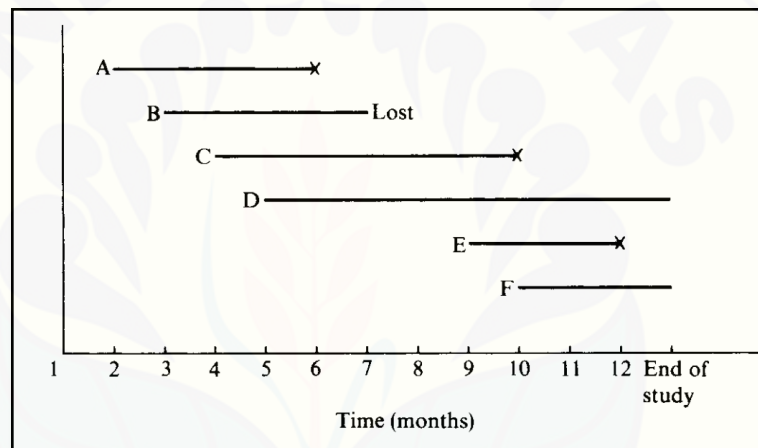
Tipe II adalah tipe penyensoran dimana sampel ke- r merupakan observasi terkecil dalam sampel random berukuran $(1 \leq r \leq n)$. Dengan kata lain, total sampel berukuran n dengan n berlanjut sampai mengalami kejadian mati atau gagal maka percobaan akan dihentikan sampai r dari unit uji n pengamatan mengalami kejadian. Semua unit uji n masuk pada waktu yang sama. Objek yang masih hidup atau belum mengalami kejadian setelah r kegagalan maka dikatakan tersensor.



Gambar 2.2 Contoh data tersensor tipe II

c. Tipe III

Pada tipe III ini, individu atau unit uji masuk ke dalam percobaan pada waktu yang berlainan selama periode waktu tertentu. Beberapa unit uji mungkin gagal atau mati sebelum pengamatan berakhir sehingga waktu tahan hidupnya dapat diketahui secara pasti. Kemungkinan kedua adalah unit uji keluar sebelum pengamatan berakhir atau kemungkinan ketiga adalah unit uji tetap hidup sampai batas waktu terakhir pengamatan. Untuk unit uji yang tetap hidup, waktu tahan hidupnya adalah dari mulai masuk pengamatan sampai dengan waktu pengamatan berakhir.

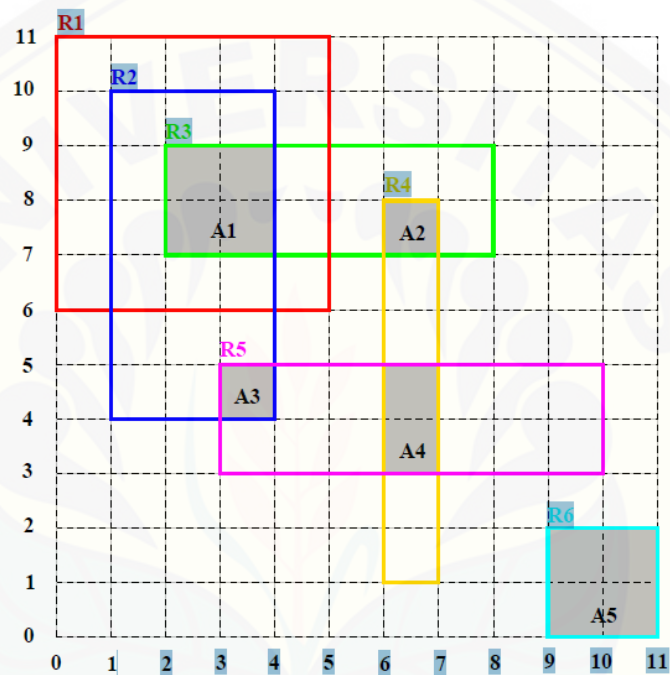


Gambar 2.3 Contoh data tersensor tipe III

2.3 Canonical Rectangles

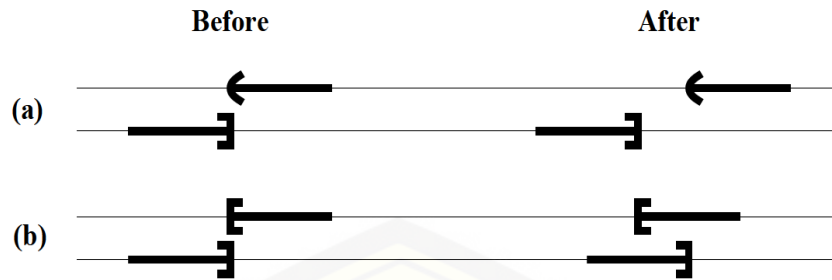
Himpunan persegi panjang pengamatan didefinisikan dengan $\mathcal{R} = R_1, \dots, R_n$ dimana setiap persegi panjang R_i dinyatakan sebagai $(x_{1i}, x_{2i}, y_{1i}, y_{2i})$. Titik (x_{1i}, y_{1i}) adalah sudut kiri bawah dan (x_{2i}, y_{2i}) adalah sudut kanan atas. Sedangkan $(x_{1i}, x_{2i}]$ adalah interval x dan $(y_{1i}, y_{2i}]$ adalah interval y pada persegi panjang R_i . Selain itu, digunakan variabel boolean $(c_{1,i}^x, c_{2,i}^x, c_{1,i}^y, c_{2,i}^y)$ untuk menunjukkan apakah *endpoint* sudah tertutup. Diasumsikan bahwa *endpoint* kiri adalah terbuka dan *endpoint* kanan adalah tertutup, maka $(c_{1,i}^x, c_{2,i}^x, c_{1,i}^y, c_{2,i}^y) = (0, 1, 0, 1)$.

Transformasikan persegi panjang pengamatan R_1, \dots, R_n kedalam *canonical rectangles* dengan struktur yang sama. Disebut himpunan n *rectangles canonical* jika semua koordinat x berbeda dan semua koordinat y juga berbeda, dan jika nilai yang diambil berada dalam himpunan $\{1, 2, \dots, 2n\}$. Gambar 2.4 adalah ilustrasi contoh dari himpunan *canonical rectangles* dimana nilai yang diambil berada dalam himpunan $\{0, 1, \dots, 2n - 1\}$.



Gambar 2.4 Contoh *canonical rectangles* (Maathuis, 2003)

Dilakukan transformasi dengan mempertimbangkan koordinat x dan koordinat y secara terpisah. Permasalahannya terletak pada kenyataan bahwa mungkin ada hubungan dalam data. Oleh karena itu, perlu didefinisikan bagaimana memisahkan hubungan tersebut. Hal tersebut dijelaskan menggunakan ide dasar dengan gambar dibawah berikut:



Gambar 2.5 Memisahkan hubungan selama transformasi persegi panjang menjadi *canonical rectangles*

Menurut Maathuis (2005), berikut ini adalah penjelasan dari gambar 2.5:

- a) Terdapat *endpoint* kiri terbuka $x_{1,i}$ dan *endpoint* kanan tertutup $x_{2,j}$ dengan $x_{1,i} = x_{2,j}$ dan $i \neq j$. Maka interval x dari R_i dan R_j tidak tumpang tindih. Oleh karena itu, perlu mengurutkan *endpoint* sehingga interval kanonik yang sama tidak tumpang tindih, yaitu dengan membiarkan $x_{2,j}$ lebih kecil.
- b) Terdapat *endpoint* kiri tertutup $x_{1,i}$ dan *endpoint* kanan tertutup $x_{2,j}$ dengan $x_{1,i} = x_{2,j}$ dan $i \neq j$. Maka interval x dari R_i dan R_j tumpang tindih. Oleh karena itu, perlu mengurutkan *endpoint* sehingga interval kanonik yang sama tumpang tindih, yaitu dengan membiarkan $x_{1,i}$ lebih kecil.

Dengan cara tersebut, dapat mempertimbangkan semua kemungkinan kombinasi dari *endpoint*. Ditemukan *code* sebuah algoritma untuk membandingkan *endpoint*. Alasan untuk mentransformasikan persegi panjang pengamatan kedalam *canonical rectangles* ada dua. Pertama, menangani hubungan dan fakta apakah *endpoint* terbuka atau tertutup. Akibatnya, tidak perlu memperhitungkan hubungan dan *endpoint* terbuka atau tertutup dialgoritma sebenarnya. Kedua, menyederhanakan algoritma reduksi karena nomor kolom dan baris pada *height map* secara langsung berhubungan dengan koordinat x dan y dari *canonical rectangles*.

2.4 Aljabar Boolean

2.4.1 Definisi Aljabar Boolean

Aljabar boolean dapat didefinisikan dengan unsur-unsur pembentuknya dan operasi-operasi yang menyertainya.

Definisi 2.1(Munir, 2005)

Misalkan B adalah himpunan yang didefinisikan pada dua operator biner, $+$ dan \cdot , dan sebuah operator uner, $'$. Misalkan 0 dan 1 adalah dua elemen yang berbeda dari B , maka:

$$\langle B, +, \cdot, 0, 1 \rangle$$

disebut aljabar Boolean jika untuk setiap $a, b, c \in B$ berlaku aksioma (sering dinamakan juga Postulat Huntington) berikut :

a. Identitas

i. $a + 0 = a$

ii. $a \cdot 1 = a$

b. Komutatif

i. $a + b = b + a$

ii. $a \cdot b = b \cdot a$

c. Distributif

i. $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

ii. $a + (b \cdot c) = (a + b) \cdot (a + c)$

d. Komplemen

Untuk setiap $a \in B$ terdapat elemen unik $a' \in B$ sehingga

i. $a + a' = 1$

ii. $a \cdot a' = 0$

Hal yang penting adalah membedakan elemen himpunan dan peubah (*variable*) pada sistem aljabar. Sebagai contoh, pada aljabar biasa, elemen himpunan bilangan riil adalah angka, sedangkan peubahnya seperti a, b, c dan sebagainya. Dengan cara yang sama pada aljabar boolean, didefinisikan elemen-elemen himpunan dan peubah seperti x, y, z sebagai simbol-simbol yang merepresentasikan elemen.

Berhubung elemen-elemen B tidak didefinisikan nilainya (maka bebas menentukan anggota-anggota B), maka untuk mempunyai sebuah aljabar Boolean, harus memperlihatkan:

1. elemen-elemen himpunan B
2. kaidah/ aturan operasi untuk dua operator biner dan operator uner
3. himpunan B , bersama-sama dengan dua operator tersebut memenuhi keempat aksioma diatas.

2.4.2 Aljabar Boolean Dua-Nilai

Aljabar Boolean yang terkenal dan memiliki terapan yang luas adalah aljabar Boolean dua-nilai (*two-valued Boolean algebra*). Aljabar Boolean dua-nilai didefinisikan pada sebuah himpunan B dengan dua buah elemen 0 dan 1, yaitu $B = \{0, 1\}$, operator biner, $+$ dan \cdot , operator uner, $'$. Kaidah untuk operator biner dan operator uner ditunjukkan pada Tabel 2.1 dan 2.2.

Tabel 2.1 Operator Biner

a	b	$a \cdot b$	$a + b$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Tabel 2.2 Operator Uner

a	a'
0	1
1	0

Untuk membuktikan himpunan $B = \{0, 1\}$ merupakan aljabar boolean maka harus menunjukkan bahwa keempat aksioma di dalam Definisi 2.1 terpenuhi pada himpunan $B = \{0, 1\}$ dengan dua operator biner dan satu operator uner.

2.4.3 Ekspresi Boolean

Definisi 2.2 (Kurniawati dan Arianti, 2010)

Ekspresi Boolean dalam n peubah (variabel) (x_1, x_2, \dots, x_n) adalah

1. 0 dan 1 adalah ekspresi boolean
2. x_1, x_2, \dots, x_n masing-masing adalah ekspresi boolean
3. Jika e_1 dan e_2 adalah ekspresi boolean, maka $e_1 + e_2$, $e_1 \cdot e_2$, e_1' adalah ekspresi boolean.

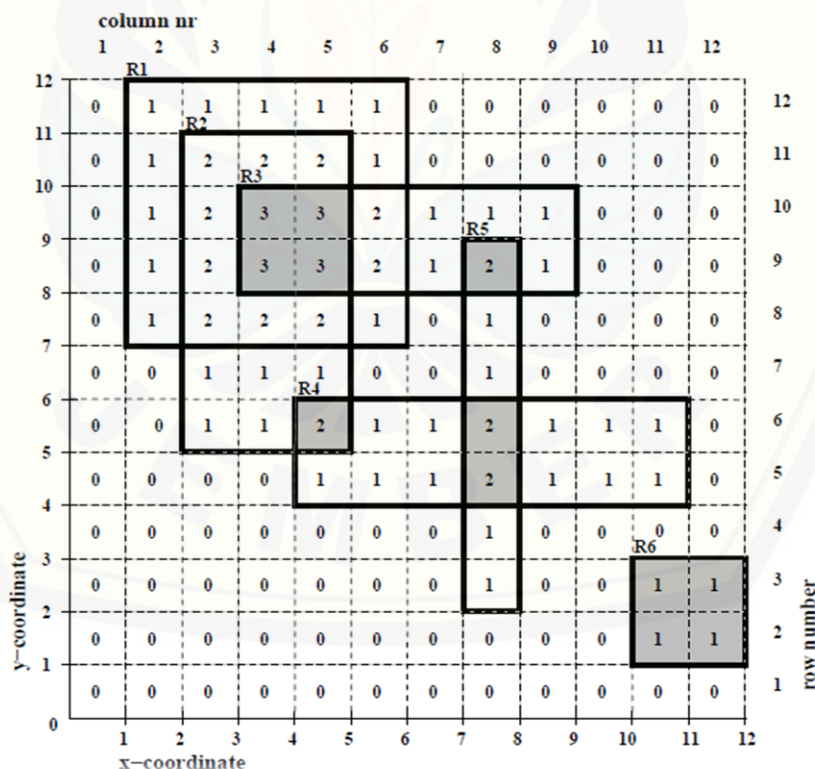
2.5 Definisi Algoritma

Menurut Munir (2005), algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis. Barakbah dkk, (2013) menyatakan bahwa menurut Kamus Besar Bahasa Indonesia (KBBI), definisi algoritma adalah urutan logis pengambilan keputusan untuk pemecahan masalah sedangkan menurut tim Gunadarma (1988), algoritma adalah suatu himpunan berhingga dari instruksi-instruksi yang secara jelas menjelaskan langkah-langkah proses pelaksanaan, dalam pemecahan suatu masalah tertentu, atau suatu kelas masalah tertentu, dengan dituntut pula bahwa himpunan instruksi tersebut dapat dilaksanakan secara mekanik. Dengan kata lain algoritma adalah urutan langkah-langkah yang dinyatakan dengan jelas dan tidak rancu untuk memecahkan suatu masalah dalam rentang waktu tertentu. Setiap langkah harus dapat dikerjakan dan mempunyai efek tertentu. Algoritma dapat dituliskan dengan banyak cara, mulai dari menggunakan bahasa alami yang digunakan sehari-hari, simbol, grafik, bagan alir sampai menggunakan bahasa pemrograman. Beberapa metode untuk merancang algoritma dalam program komputer yaitu : diagram alir (*flow chart*), kode semu (*pseudo code*) dan algoritma fundamental.

2.6 Algoritma *HeightMap*

Menurut Maathuis (2005), algoritma reduksi baru yaitu algoritma *height map* sangat efisien untuk mencari *maximal intersections*. Pada penelitian ini akan dicari *maximal intersections* A_1, \dots, A_m pada himpunan persegi panjang pengamatan R_1, \dots, R_m . Terdapat beberapa definisi yang sama untuk konsep

maximal intersections pada literatur. Wong dan Yu (1999) menggunakan berikut: $A_j \neq \emptyset$ adalah *maximal intersections* jika dan hanya jika merupakan irisan terbatas dari R_i sehingga untuk setiap $A_j \cap R_i = \emptyset$ atau $A_j \cap R_i = A_j$. Gentleman dan Vandal (2002) menggunakan teori graf prespektif: *maximal intersections* adalah representasi nyata dari *maximal cliques* pada graf irisan dari persegi panjang pengamatan. Maathuis (2005) menunjukkan *maximal intersections* dengan cara yang berbeda, yaitu dengan mendefinisikan fungsi *height map* dari persegi panjang pengamatan. Fungsi *height map* tersebut adalah $h: \mathbb{R}^2 \rightarrow \mathbb{N}$, dimana $h(x, y)$ merupakan jumlah dari persegi panjang pengamatan yang berisi titik (x, y) . Konsep dari *height map* diilustrasikan pada gambar 2.6. Pada ilustrasi gambar tersebut dapat dilihat bahwa *maximal intersections* ekuivalen dengan *local maxima* pada *height map*. Hal ini berlaku setiap kali ada hubungan antara persegi panjang pengamatan, dan bentuk pengamatan tersebut menjadi dasar algoritma.



Gambar 2.6 Contoh dari 6 persegi panjang pengamatan dan *height map* dari 6 persegi panjang pengamatan

BAB 3. METODE PENELITIAN

3.1 Data

Data yang digunakan dalam penelitian ini merupakan data HIV/AIDS yang diperoleh dari RSUD Balung, Jember. Data tersebut merupakan data tersensor interval bivariat HIV/AIDS sebanyak 21 data pasien.

3.2 Langkah-Langkah Penelitian

Langkah-langkah yang akan dilakukan dalam penelitian tentang “Aplikasi Algoritma *Height Map* untuk Mencari *Maximal Intersection* pada Data Tersensor Interval Bivariat”, secara sistematis dapat dilihat pada gambar 3.1

Penjelasan skema pada Gambar 3.1 untuk memperoleh hasil yang diinginkan adalah sebagai berikut:

a. Data Bivariat HIV/AIDS

Langkah pertama yang dilakukan adalah mencari data bivariat HIV/AIDS di RSUD Balung, Jember. Selanjutnya, menentukan variabel-variabel yang akan digunakan dari data bivariat kasus HIV/AIDS.

b. Persegi panjang pengamatan (*Rectangles Observation*)

Data yang telah ditentukan diubah menjadi bentuk persegi panjang pengamatan (*rectangles observation*).

c. *Canonical Rectangles*

Dari n persegi panjang pengamatan akan ditransformasi menjadi *canonical rectangles*.

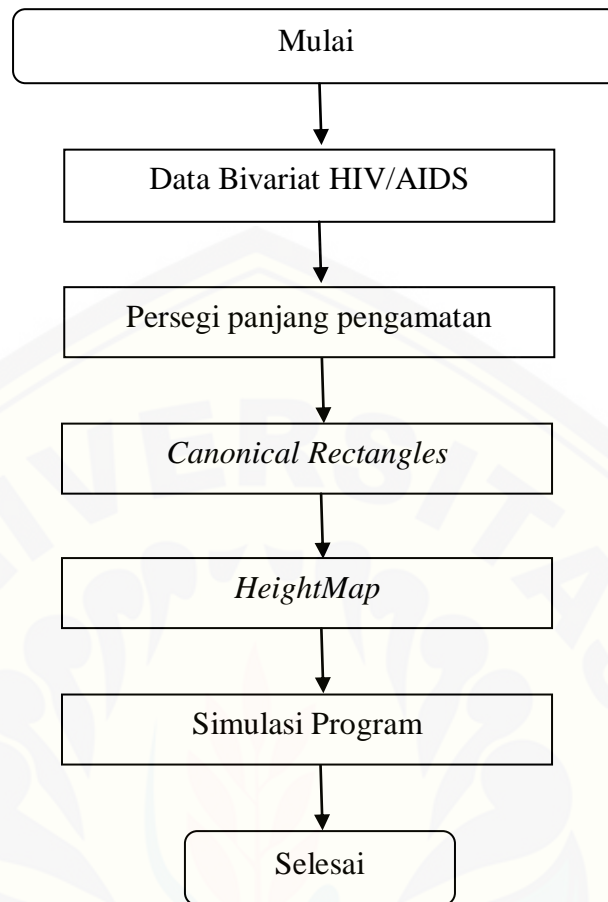
d. *HeightMap*

Pada langkah ini, setelah didapat bentuk *canonical rectangles* sesuai dengan langkah diatas maka akan dibangun *height map*. Selama proses membangun *height map*, dapat ditemukan *maximal intersections*.

e. Simulasi Program

Simulasi program algoritma *Height Map* menggunakan program Matlab

f. Selesai



Gambar 3.1 Skema langkah-langkah penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan diperoleh kesimpulan sebagai berikut:

- a. Bentuk *canonical rectangles* hasil transformasi memiliki struktur irisan (*intersections*) yang sama dengan persegi panjang hasil pengamatan.
- b. Operasi algoritma *height map* pada data bivariat pasien HIV/AIDS RSUD Balung, Jember menghasilkan 12 *maximal intersections* yaitu $M_1 = \{R_6, R_{15}, R_{17}\}$, $M_2 = \{R_5, R_{15}, R_{17}\}$, $M_3 = \{R_5, R_{15}, R_{16}\}$, $M_4 = \{R_5, R_{14}, R_{16}\}$, $M_5 = \{R_{20}\}$, $M_6 = \{R_6, R_{11}, R_{17}, R_{18}\}$, $M_7 = \{R_3, R_{21}\}$, $M_8 = \{R_4, R_6, R_{11}\}$, $M_9 = \{R_4, R_6, R_{10}\}$, $M_{10} = \{R_2, R_7, R_{11}\}$, $M_{11} = \{R_3, R_7, R_8, R_9, R_{11}, R_{12}, R_{13}, R_{19}\}$, $M_{12} = \{R_1\}$.

5.2 Saran

Pada penelitian ini digunakan data bivariat kasus pasien HIV/AIDS. Untuk penelitian selanjutnya dapat menggunakan data multivariat dan juga kasus yang berbeda untuk mencari *maximal intersections*.

DAFTAR PUSTAKA

- Barakbah, A.R, Karlita, T dan Ahsan, A.S. 2013. *Logika dan Algoritma*. Surabaya: Politeknik Elektronika Negeri Surabaya.
- Betensky, R. A. dan Finkelstein, D. M. 1999. A Nonparametric Maximum Likelihood Estimator for Bivariate Censored Data. *Statistics in Medicine*, 18, 3089-3100.
- Collett, D. 2003. *Modelling Survival Data in Medical Research*. London: Champman and Hall.
- Gentleman, R dan Vandal, A.C. 2002. Nonparametric Estimation of the Bivariate CDF for Arbitrarily Censored Data, *The Canadian Journal of Statistics*, 30, 557-571.
- Kleinbaum, D.G dan Klein, M. 2005. *Survival Analysis A Self-Learning Text Second Edition*. New York: Springer.
- Kurniawati, A.T dan Arianti, D. 2010. *Buku Ajar (Diktat) Matematika Diskrit*. Surabaya: Blog Dosen ITATS.
- Lee, E.T dan Wang, J.W. 2003. *Statistical Methods for Survival Data Analysis Third Edition*. New Jersey: John Wiley and Sons.
- Maathuis, M. H. 2003. *Nonparametric Maximum Likelihood Estimation for Bivariate Censored Data*. Master's thesis, Delft University of Technology, Delft.
- Maathuis, M. H. 2005. Reduction Algorithm for NPMLE for the Distribution Function of Bivariate Interval Censored Data. *J Comp Graph Statist* 14: 252-262.
- Munir, R. 2005. *Matematika Diskrit*. Bandung: Informatika Bandung.
- Murni, Z, dkk. 2009. *Hidup Dengan HIV/AIDS*. Jakarta: Yayasan Spiritia.

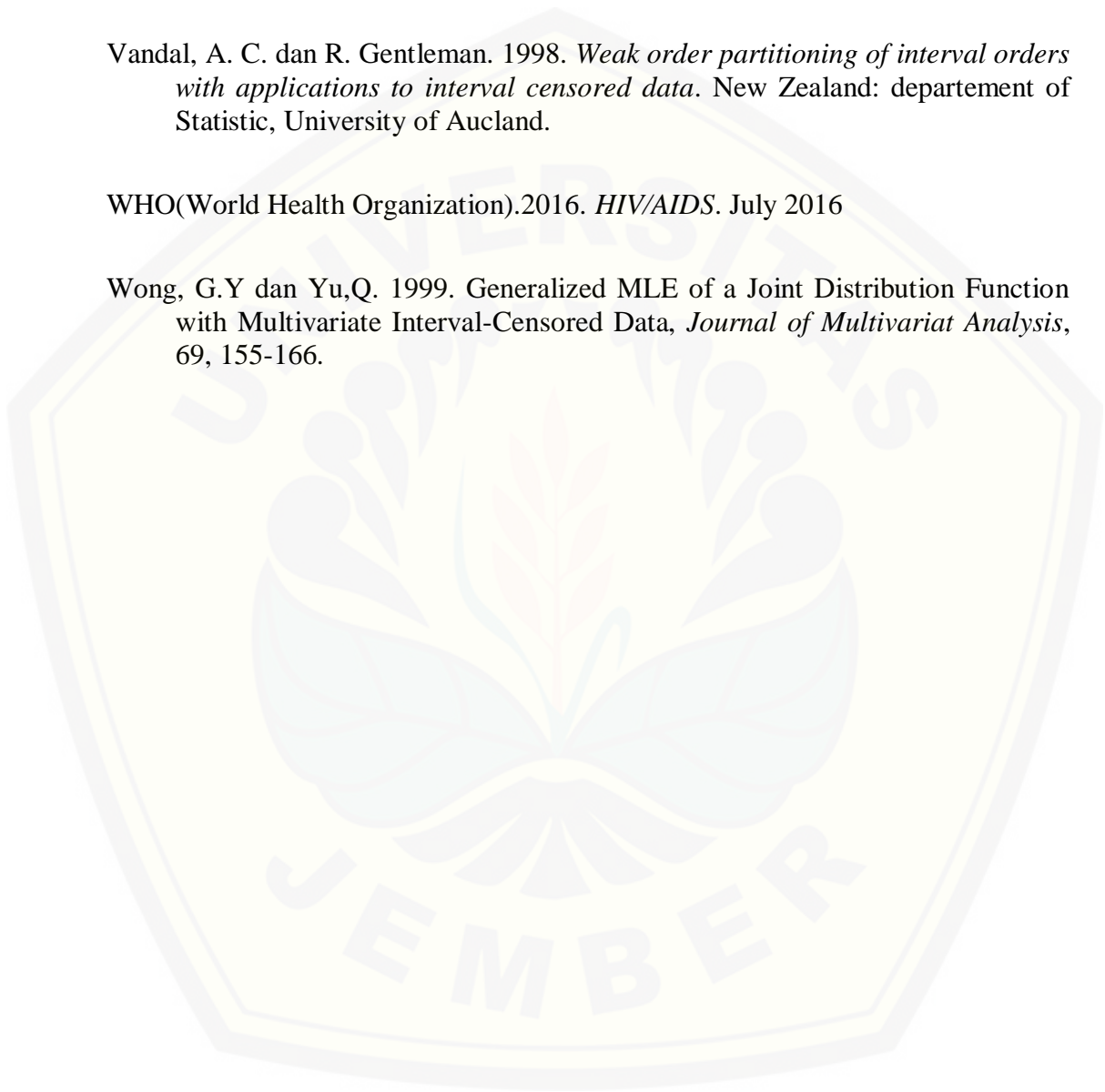
Murti, B. 1997. *Prinsip dan Metode Riset Epidemiologi*. Yogyakarta: Gajah Mada University press.

Nelson, W. 1982. *Applied Life Data Analysis*. New York: John Wiley & Sons, Inc.

Vandal, A. C. dan R. Gentleman. 1998. *Weak order partitioning of interval orders with applications to interval censored data*. New Zealand: departement of Statistic, University of Aucland.

WHO(World Health Organization).2016. *HIV/AIDS*. July 2016

Wong, G.Y dan Yu,Q. 1999. Generalized MLE of a Joint Distribution Function with Multivariate Interval-Censored Data, *Journal of Multivariat Analysis*, 69, 155-166.



LAMPIRAN

A. 30 Data pasien HIV/AIDS RSUD Balung, Jember

No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	INTERVAL HIV	MASA INKUBASI	INTERVAL AIDS
1	YNS	30th	L	26 April 2010	Stadium 3	1999-2002	2002-2010	2010-2011
2	AH	53th	L	07 Februari 2013	Stadium 3	2004-2007	2007-2013	2013-2014
3	SDH	33th	P	21 April 2011	Stadium 3	2000-2003	2003-2011	2011-2015
4	IBR	39th	L	20 Januari 2014	Stadium 3	2004-2007	2007-2014	2014-2015
5	SNA	36th	P	10 Agustus 2016	Stadium 3	2006-2009	2009-2016	2016-2017
6	AQ	31th	L	05 Februari 2014	Stadium 3	2003-2006	2006-2014	2014-2017
7	NR	36th	P	11 Juni 2012	Stadium 3	2002-2005	2005-2012	2012-2014
8	HNK	31th	P	25 Nopember 2011	Stadium 3	2000-2003	2003-2011	2011-2013
9	RSD	28th	P	20 Desember 2012	Stadium 3	2001-2004	2004-2012	2012-2013
10	NTY	54th	P	04 Agustus 2014	Stadium 3	2005-2008	2008-2014	2014-2015
11	SA	40th	L	12 Nopember 2012	Stadium 3	2002-2005	2005-2012	2012-2017
12	SK	43th	L	10 Maret 2012	Stadium 3	2002-2005	2005-2012	2012-2013
13	AH	31th	L	27 Juni 2011	Stadium 3	2000-2003	2003-2011	2011-2013
14	SDH	33th	P	21 April 2011	Stadium 3	2000-2003	2003-2011	2011-2014
15	HNK	31th	P	08 Desember 2011	Stadium 3	2000-2003	2003-2011	2011-2013
16	SM	61th	L	02 Maret 2016	Stadium 3	2008-2011	2011-2016	2016-2017
17	ALA	25th	L	23 Mei 2016	Stadium 3	2005-2008	2008-2016	2016-2017
18	SR	43th	L	13 Oktober 2016	Stadium 3	2006-2009	2009-2016	2016-2017

19	SD	64th	L	10 Februari 2016	Stadium 3	2008-2011	2011-2016	2016-2017
20	ST	48th	L	14 Juni 2016	Stadium 3	2006-2009	2009-2016	2016-2017
21	TB	51th	L	21 Oktober 2016	Stadium 3	2007-2010	2010-2016	2016-2017
22	TS	48th	P	20 Oktober 2014	Stadium 3	2004-2007	2007-2014	2014-2015
23	WH	25th	P	18 Desember 2015	Stadium 3	2004-2007	2007-2015	2015-2017
24	YE	42th	P	26 Juli 2016	Stadium 3	2006-2009	2009-2016	2016-2017
25	SYF	26th	P	13 Juli 2016	Stadium 3	2005-2008	2008-2016	2016-2017
26	MN	27th	L	28 Oktober 2015	Stadium 3	2004-2007	2007-2015	2015-2016
27	SA	40th	L	12 Nopember 2012	Stadium 3	2002-2005	2005-2012	2012-2014
28	HR	41th	L	24 Nopember 2011	Stadium 3	2001-2004	2004-2011	2011-2017
29	RB	1th	L	31 Oktober 2016	Stadium 3	2012-2015	2015-2016	2016-2017
30	EDS	30th	L	13 Mei 2015	Stadium 3	1998-2001	2001-2014	2014-2017

B. 21 Data bivariat pasien HIV/AIDS RSUD Balung, Jember

No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	INTERVAL HIV	MASA INKUBASI	INTERVAL AIDS
1	YNS	30th	L	26 April 2010	Stadium 3	1999-2002	2002-2010	2010-2011
2	AH	53th	L	07 Februari 2013	Stadium 3	2004-2007	2007-2013	2013-2014
3	SDH	33th	P	21 April 2011	Stadium 3	2000-2003	2003-2011	2011-2015
4	IBR	39th	L	20 Januari 2014	Stadium 3	2004-2007	2007-2014	2014-2015
5	SNA	36th	P	10 Agustus 2016	Stadium 3	2006-2009	2009-2016	2016-2017
6	AQ	31th	L	05 Februari 2014	Stadium 3	2003-2006	2006-2014	2014-2017
7	NR	36th	P	11 Juni 2012	Stadium 3	2002-2005	2005-2012	2012-2014
8	HNK	31th	P	25 Nopember 2011	Stadium 3	2000-2003	2003-2011	2011-2013
9	RSD	28th	P	20 Desember 2012	Stadium 3	2001-2004	2004-2012	2012-2013
10	NTY	54th	P	04 Agustus 2014	Stadium 3	2005-2008	2008-2014	2014-2015
11	SA	40th	L	12 Nopember 2012	Stadium 3	2002-2005	2005-2012	2012-2017
12	SK	43th	L	10 Maret 2012	Stadium 3	2002-2005	2005-2012	2012-2013
13	SDH	33th	P	21 April 2011	Stadium 3	2000-2003	2003-2011	2011-2014
14	SM	61th	L	02 Maret 2016	Stadium 3	2008-2011	2011-2016	2016-2017
15	ALA	25th	L	23 Mei 2016	Stadium 3	2005-2008	2008-2016	2016-2017
16	TB	51th	L	21 Oktober 2016	Stadium 3	2007-2010	2010-2016	2016-2017
17	WH	25th	P	18 Desember 2015	Stadium 3	2004-2007	2007-2015	2015-2017
18	MN	27th	L	28 Oktober 2015	Stadium 3	2004-2007	2007-2015	2015-2016
19	HR	41th	L	24 Nopember 2011	Stadium 3	2001-2004	2004-2011	2011-2017
20	RB	1th	L	31 Oktober 2016	Stadium 3	2012-2015	2015-2016	2016-2017
21	EDS	30th	L	13 Mei 2015	Stadium 3	1998-2001	2001-2014	2014-2017

C. Data univariat pasien HIV/AIDS RSUD Balung, Jember (L_i = awal masa inkubasi, R_i = akhir masa inkubasi)

No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	MASA INKUBASI
1	YNS	30th	L	26 April 2010	Stadium 3	2002-2010
2	AH	53th	L	07 Februari 2013	Stadium 3	2007-2013
3	SDH	33th	P	21 April 2011	Stadium 3	2003-2011
4	IBR	39th	L	20 Januari 2014	Stadium 3	2007-2014
5	SNA	36th	P	10 Agustus 2016	Stadium 3	2009-2016
6	AQ	31th	L	05 Februari 2014	Stadium 3	2006-2014
7	NR	36th	P	11 Juni 2012	Stadium 3	2005-2012
8	HNK	31th	P	25 Nopember 2011	Stadium 3	2003-2011
9	RSD	28th	P	20 Desember 2012	Stadium 3	2004-2012
10	NTY	54th	P	04 Agustus 2014	Stadium 3	2008-2014
11	SA	40th	L	12 Nopember 2012	Stadium 3	2005-2012
12	SK	43th	L	10 Maret 2012	Stadium 3	2005-2012
13	SDH	33th	P	21 April 2011	Stadium 3	2003-2011
14	SM	61th	L	02 Maret 2016	Stadium 3	2011-2016
15	ALA	25th	L	23 Mei 2016	Stadium 3	2008-2016
16	TB	51th	L	21 Oktober 2016	Stadium 3	2010-2016
17	WH	25th	P	18 Desember 2015	Stadium 3	2007-2015
18	MN	27th	L	28 Oktober 2015	Stadium 3	2007-2015
19	HR	41th	L	24 Nopember 2011	Stadium 3	2004-2011
20	RB	1th	L	31 Oktober 2016	Stadium 3	2015-2016
21	EDS	30th	L	13 Mei 2015	Stadium 3	2001-2014

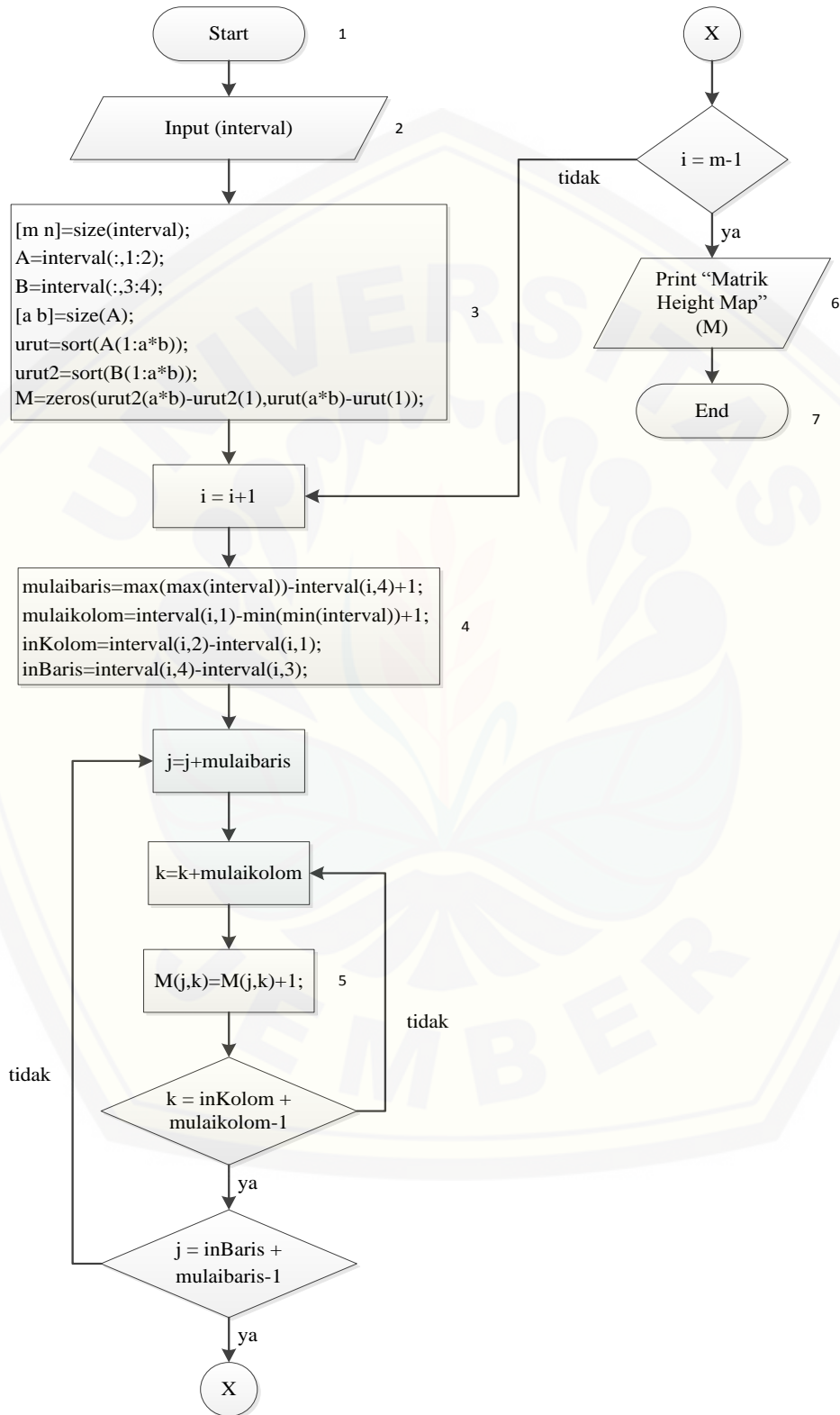
D. Data univariat pasien HIV/AIDS RSUD Balung, Jember (L_i = awal terinfeksi HIV, R_i = akhir terjangkit AIDS)

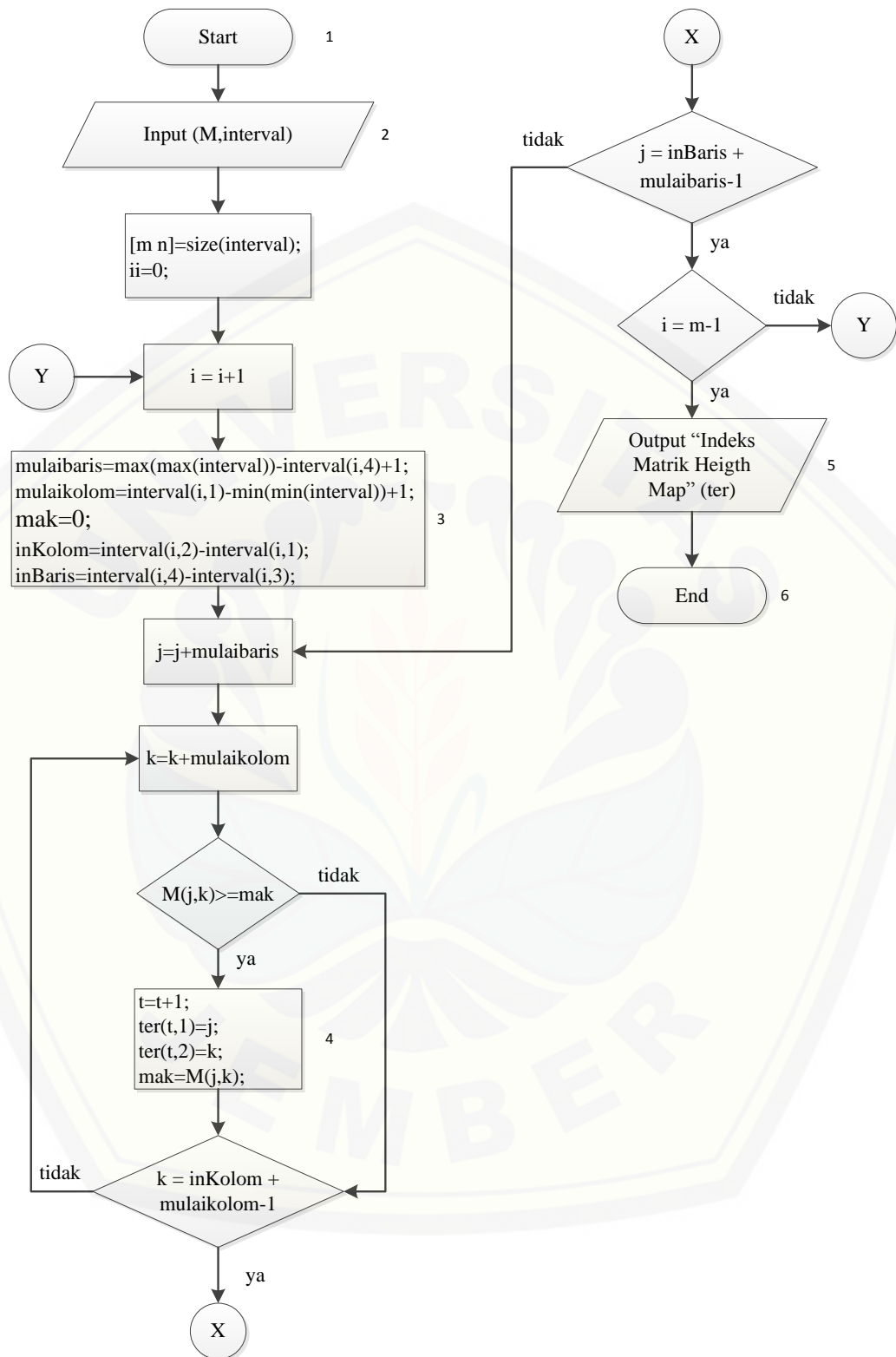
No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	AWAL HIV	AKHIR AIDS
1	YNS	30th	L	26 April 2010	Stadium 3	1999	2011
2	AH	53th	L	07 Februari 2013	Stadium 3	2004	2014
3	SDH	33th	P	21 April 2011	Stadium 3	2000	2015
4	IBR	39th	L	20 Januari 2014	Stadium 3	2004	2015
5	SNA	36th	P	10 Agustus 2016	Stadium 3	2006	2017
6	AQ	31th	L	05 Februari 2014	Stadium 3	2003	2017
7	NR	36th	P	11 Juni 2012	Stadium 3	2002	2014
8	HNK	31th	P	25 Nopember 2011	Stadium 3	2000	2013
9	RSD	28th	P	20 Desember 2012	Stadium 3	2001	2013
10	NTY	54th	P	04 Agustus 2014	Stadium 3	2005	2015
11	SA	40th	L	12 Nopember 2012	Stadium 3	2002	2017
12	SK	43th	L	10 Maret 2012	Stadium 3	2002	2013
13	SDH	33th	P	21 April 2011	Stadium 3	2000	2014
14	SM	61th	L	02 Maret 2016	Stadium 3	2008	2017
15	ALA	25th	L	23 Mei 2016	Stadium 3	2005	2017
16	TB	51th	L	21 Oktober 2016	Stadium 3	2007	2017
17	WH	25th	P	18 Desember 2015	Stadium 3	2004	2017
18	MN	27th	L	28 Oktober 2015	Stadium 3	2004	2016
19	HR	41th	L	24 Nopember 2011	Stadium 3	2001	2017
20	RB	1th	L	31 Oktober 2016	Stadium 3	2012	2017
21	EDS	30th	L	13 Mei 2015	Stadium 3	1998	2017

E. Data univariat pasien HIV/AIDS RSUD Balung, Jember (L_i = awal terinfeksi HIV, R_i = awal terjangkit AIDS)

No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	AWAL HIV	AWAL AIDS
1	YNS	30th	L	26 April 2010	Stadium 3	1999	2010
2	AH	53th	L	07 Februari 2013	Stadium 3	2004	2013
3	SDH	33th	P	21 April 2011	Stadium 3	2000	2011
4	IBR	39th	L	20 Januari 2014	Stadium 3	2004	2014
5	SNA	36th	P	10 Agustus 2016	Stadium 3	2006	2016
6	AQ	31th	L	05 Februari 2014	Stadium 3	2003	2014
7	NR	36th	P	11 Juni 2012	Stadium 3	2002	2012
8	HNK	31th	P	25 Nopember 2011	Stadium 3	2000	2011
9	RSD	28th	P	20 Desember 2012	Stadium 3	2001	2012
10	NTY	54th	P	04 Agustus 2014	Stadium 3	2005	2014
11	SA	40th	L	12 Nopember 2012	Stadium 3	2002	2012
12	SK	43th	L	10 Maret 2012	Stadium 3	2002	2012
13	SDH	33th	P	21 April 2011	Stadium 3	2000	2011
14	SM	61th	L	02 Maret 2016	Stadium 3	2008	2016
15	ALA	25th	L	23 Mei 2016	Stadium 3	2005	2016
16	TB	51th	L	21 Oktober 2016	Stadium 3	2007	2016
17	WH	25th	P	18 Desember 2015	Stadium 3	2004	2015
18	MN	27th	L	28 Oktober 2015	Stadium 3	2004	2015
19	HR	41th	L	24 Nopember 2011	Stadium 3	2001	2011
20	RB	1th	L	31 Oktober 2016	Stadium 3	2012	2016
21	EDS	30th	L	13 Mei 2015	Stadium 3	1998	2014

F. Flow chart program aplikasi algoritma *Height Map*





G. Skrip mencari irisan persegi panjang

```
function [M urut urut2 N O]=irisanpersegi(interval)
[m n]=size(interval);
A=interval(:,1:2);
B=interval(:,3:4);
[a b]=size(A);
urut=sort(A(1:a*b));
urut2=sort(B(1:a*b));
M=zeros(urut2(a*b)-urut2(1),urut(a*b)-urut(1));
for i=1:m-1
    mulaibaris=max(max(interval))-interval(i,4)+1;
    mulaikolom=interval(i,1)-min(min(interval))+1;
    inKolom=interval(i,2)-interval(i,1);
    inBaris=interval(i,4)-interval(i,3);
    for j=mulaibaris:inBaris+mulaibaris-1
        for k=mulaikolom:inKolom+mulaikolom-1
            M(j,k)=M(j,k)+1;
        end
    end
end
[br k]=size(M);
for i=1:br+1
    tt=min(min(interval));

    for j=1:k+1
        N(i,j)=tt;
        tt=tt+1;
    end
end
for i=1:k+1
    to=max(max(interval));
    for j=1:br+1
        O(j,i)=to;
        to=to-1;
    end
end
end
```

H. Skrip mencari irisan terbesar

```
function terbesar=cariterbesar(M,interval)
[m n]=size(interval);
ii=0;
for i=1:m-1
    t=0;
    mulaibaris=max(max(interval))-interval(i,4)+1;
    mulaikolom=interval(i,1)-min(min(interval))+1;
    mak=0;
    inKolom=interval(i,2)-interval(i,1);
    inBaris=interval(i,4)-interval(i,3);
    for j=mulaibaris:inBaris+mulaibaris-1
    for k=mulaikolom:inKolom+mulaikolom-1
    if M(j,k)>=mak
        t=t+1;
        ter(t,1)=j;
        ter(t,2)=k;
        mak=M(j,k);
    end
    end
    end
    [brs kol]=size(ter);
    for z=1:brs
        A(z)=M(ter(z,1),ter(z,2));
    end
    mmak=max(A);
    for v=1:brs
    if A(v)==mmak
        ii=ii+1;
        terbesar(ii,1)=ter(v,1);
        terbesar(ii,2)=ter(v,2);
    end
    end
    ter=0;
    A=0;
end
```

I. Skrip GUI algoritma *Height Map*

```

function varargout = survivalirisanpersegi(varargin)
% SURVIVALIRISANPERSEGI M-file for survivalirisanpersegi.fig
%   SURVIVALIRISANPERSEGI, by itself, creates a new
SURVIVALIRISANPERSEGI or raises the existing
%   singleton*.
%
%   H = SURVIVALIRISANPERSEGI returns the handle to a new
SURVIVALIRISANPERSEGI or the handle to
%   the existing singleton*.
%
%
SURVIVALIRISANPERSEGI('CALLBACK', hObject,eventData,handles,..
..) calls the local
%   function named CALLBACK in SURVIVALIRISANPERSEGI.M
with the given input arguments.
%
%   SURVIVALIRISANPERSEGI('Property','Value',...) creates
a new SURVIVALIRISANPERSEGI or raises the
%   existing singleton*. Starting from the left,
property value pairs are
%   applied to the GUI before
survivalirisanpersegi_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes
property application
%   stop. All inputs are passed to
survivalirisanpersegi_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI
allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
survivalirisanpersegi

% Last Modified by GUIDE v2.5 29-Apr-2017 15:21:27

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @survivalirisanpersegi_OpeningFcn, ...
'gui_OutputFcn',  @survivalirisanpersegi_OutputFcn, ...
'gui_LayoutFcn',  [] , ...
'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});

```

```
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State,
varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before survivalirisanpersegi is made
visible.
function survivalirisanpersegi_OpeningFcn(hObject,
eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
% varargin   command line arguments to survivalirisanpersegi
(see VARARGIN)

% Choose default command line output for
survivalirisanpersegi
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes survivalirisanpersegi wait for user response
(see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command
line.
function varargout =
survivalirisanpersegi_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see
VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
```

```

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton1 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see
GUIDATA)
[FileName,PathName]=uigetfile({'*txt','Pilih file .txt'});
if FileName~=0
    data=importdata([PathName,FileName]);
    [m n]=size(data);
    for i=1:m
        ro(i)={'R' num2str(i)};
    end
    set(handles.uitable1,'data',data,'rowname',ro,...
'columnname',{'x1','x2','y1','y2'},'columnwidth',{55});
    set(handles.edit1,'string',[PathName FileName]);
    set(handles.figure1,'userdata',ro);
    set(handles.uitable1,'columneditable',true)
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton2 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see
GUIDATA)
clc
data=get(handles.uitable1,'data');
ir=iris(data);
dlmwrite('irisan.txt', ir, 'delimiter', '\t', ...
'precision', 6)
[m n]=size(data);
a=data(:,1:2);
b=data(:,3:4);
[v w]=size(data);
for i=1:v
    text(data(i,1),data(i,4),['\color{black}R'
num2str(i)],...
'fontsize',11,'fontweight','normal','fontname','cambria',...
'HorizontalAlignment','left','VerticalAlignment','baseline')
end
data(m+1,:)=[min(min(a))-1 max(max(a))+1 min(min(b))-1
max(max(b))+1];
[m n]=size(data);
a=data(:,1:2);
b=data(:,3:4);
x=min(min(a)):max(max(a));
y=min(min(b)):max(max(b));

```



```

set(handles.figure1, 'CurrentAxes', handles.axes1);
set(handles.axes1, 'nextplot', 'replace');
for i=1:length(x)
    plot([x(i) x(i)], [y(1) y(length(y))], 'k:');
    set(handles.axes1, 'nextplot', 'add');
end
for i=1:length(y)
    plot([x(1) x(length(x))], [y(i) y(i)], 'k:');
end
for i=1:m-1
    wr=rand(1,3);
    plot([data(i,1) data(i,1)], [data(i,3)
data(i,4)], 'color', wr, 'linewidth', 2.5);
    plot([data(i,1) data(i,2)], [data(i,3)
data(i,3)], 'color', wr, 'linewidth', 2.5);
    plot([data(i,1) data(i,2)], [data(i,4)
data(i,4)], 'color', wr, 'linewidth', 2.5);
    plot([data(i,2) data(i,2)], [data(i,3)
data(i,4)], 'color', wr, 'linewidth', 2.5);
    text(data(i,1), data(i,4), ['\color{blue}R'
num2str(i)], ...
'fontsize', 10, 'fontweight', 'normal', 'fontname', 'cambria', ...
'HorizontalAlignment', 'left', 'VerticalAlignment', 'baseline')
end

set(handles.axes1, 'Color', [1 1 1], 'XColor', [0 0
0], 'YColor', [0 0 0]);
set(handles.axes1, 'XLim', [min(x) max(x)], 'YLim', [min(y)
max(y)], ...
'ytick', y, 'xgrid', 'off', 'ygrid', 'off');
xlabel('x (HIV)', 'fontname', 'cambria', 'fontsize', 11);
ylabel('y (AIDS)', 'fontname', 'cambria', 'fontsize', 11);

[M urut urut2 N O]=irisanpersegi(data);
[m n]=size(M);
for i=1:m
for j=1:n
    x=(N(i,j)+N(i,j+1))/2;
    y=(O(i+1,j)+O(i,j))/2;
    text(x,y, ['\color{black}' num2str(M(i,j))], ...
'fontsize', 10, 'fontweight', 'normal', 'fontname', 'cambria', ...
'HorizontalAlignment', 'center')
end
end
mak=cariterbesar(M, data);
for i=1:length(mak)
    X=(N(mak(i,1), mak(i,2))+N(mak(i,1), mak(i,2)+1))/2;
    Y=(O(mak(i,1), mak(i,2))+O(mak(i,1)+1, mak(i,2)))/2;
    plot(X, Y, 's', 'markersize', 20, 'markerfacecolor', [.702
.780 1], 'markeredgecolor', 'k');

```

```
    text(X,Y,['\color{black}'
num2str(M(mak(i,1),mak(i,2))),...
'fontsize',12,'fontweight','bold','fontname','cambria',...
'HorizontalAlignment','center')
end

function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as
text
%          str2double(get(hObject,'String')) returns contents
of edit1 as a double

% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on
Windows.
%          See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
close

% --- Executes during object creation, after setting all
properties.
function uitable1_CreateFcn(hObject, eventdata, handles)
```

```
% hObject    handle to uitable1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called
set(hObject, 'data', []);

% -----
function Untitled_1_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% -----
function Untitled_5_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% -----
function Edit_Callback(hObject, eventdata, handles)
% hObject    handle to Edit (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% -----
function Tambah_Callback(hObject, eventdata, handles)
% hObject    handle to Tambah (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
data=get(handles.uitable1, 'data');
ro=get(handles.figure1, 'userdata');
m=length(ro);
ro(m+1)={'R' num2str(m+1)};
tambah=zeros(1,4);
```

```

set(handles.uitable1,'data',[data; tambah],'rowname',ro)
set(handles.uitable1,'columneditable',true)
set(handles.figure1,'userdata',ro);

% -----
function Kurangi_Callback(hObject, eventdata, handles)
% hObject    handle to Kurangi (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
prompt = {'Pilih Pasien yang dihapus : '};
dlg_title = 'Kurangi Data';
num_lines = 1;
answer = inputdlg(prompt,dlg_title,num_lines);
a=str2double(answer);
if a~=0
    data=get(handles.uitable1,'data');
    ro=get(handles.figure1,'userdata');
    m=length(ro);
    ro=ro(1:m-1);
    t=0;
    for i=1:m
    if i~=a
        t=t+1;
        kurang(t,:)=data(i,:);
    end
    end
    set(handles.uitable1,'data',kurang,'rowname',ro)
    set(handles.uitable1,'columneditable',true)
    set(handles.figure1,'userdata',ro);
end

% -----
function Buka_Callback(hObject, eventdata, handles)
% hObject    handle to Buka (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
[FileName,PathName]=uigetfile({'*txt','Pilih file .txt'});
if FileName~=0
    data=importdata([PathName,FileName]);
    [m n]=size(data);
    for i=1:m
        ro(i)={'R' num2str(i)};
    end
    set(handles.uitable1,'data',data,'rowname',ro,...
'columnname',{'x1','x2','y1','y2'},'columnwidth',{55});

```

```

        set(handles.edit1,'string',[PathName FileName]);
        set(handles.figure1,'userdata',ro);
        set(handles.uitable1,'columneditable',true)
    end

% -----
% -----
function Save_Callback(hObject, eventdata, handles)
% hObject    handle to Save (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
[namafile,direktori]=uiputfile('*.txt','Simpan Data');
if namafile~=0
    eval(['cd '' direktori '';']);
    namafile=[namafile ''];
    fout=fopen(namafile,'wt');
    d=get(handles.uitable1,'data');
    [m n]=size(d);
    for i=1:m
        for j=1:n
            fprintf(fout,'%g ',d(i,j));
        end
        fprintf(fout,'\n');
    end
    fclose(fout);
end

% -----
% -----
function Keluar_Callback(hObject, eventdata, handles)
% hObject    handle to Keluar (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
close

% -----
% -----
function Reset_Callback(hObject, eventdata, handles)
% hObject    handle to Reset (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)

```

```

% hObject      handle to pushbutton4 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see
GUIDATA)
data=get(handles.uitable1,'data');
ro=get(handles.figure1,'userdata');
m=length(ro);
ro(m+1)=[ 'R' num2str(m+1) ];
tambah=zeros(1,4);
set(handles.uitable1,'data',[data; tambah],'rowname',ro)
set(handles.uitable1,'columneditable',true)
set(handles.figure1,'userdata',ro);

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton5 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see
GUIDATA)
prompt = {'Pilih Pasien yang dihapus : '};
dlg_title = 'Kurangi Data';
num_lines = 1;
answer = inputdlg(prompt,dlg_title,num_lines);
a=str2double(answer);
if a~=0
    data=get(handles.uitable1,'data');
    ro=get(handles.figure1,'userdata');
    m=length(ro);
    ro=ro(1:m-1);
    t=0;
for i=1:m
if i~=a
            t=t+1;
            kurang(t,:)=data(i,:);
end
end
    set(handles.uitable1,'data',kurang,'rowname',ro)
    set(handles.uitable1,'columneditable',true)
    set(handles.figure1,'userdata',ro);
end

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton6 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see
GUIDATA)

```

```
[namafile,direktori]=uiputfile('*.txt','Simpan Data');
if namafile~=0
    eval(['cd '' direktori '';']);
    namafile=[namafile ''];
    fout=fopen(namafile,'wt');
    d=get(handles.uitable1,'data');
    [m n]=size(d);
for i=1:m
for j=1:n
        fprintf(fout,'%g ',d(i,j));
end
        fprintf(fout,'\n');
end
fclose(fout);
end

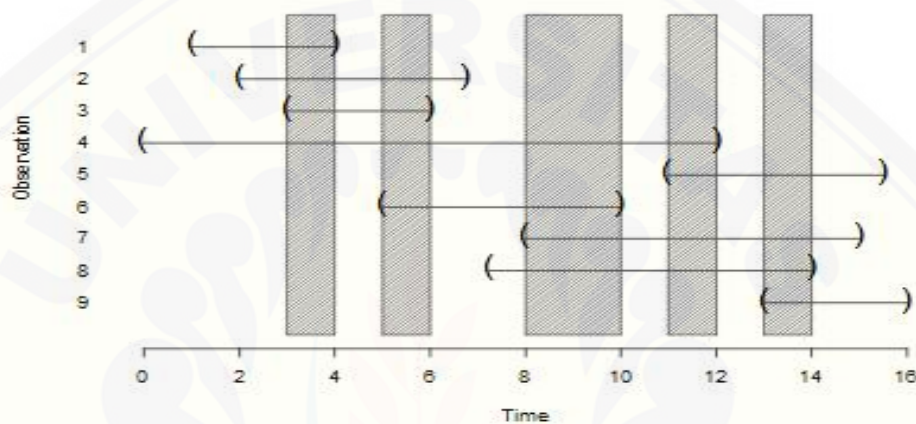
% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton7 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see
GUIDATA)
tabelirisan
```

J. Antichains

$M \subset X$ disebut *antichains* dari X jika $x \sim y$ untuk setiap $x, y \in M$. *Maximal antichains* adalah *antichains* yang tidak terletak pada *antichains* lain (Vandal dan Gentleman, 1998(weak)).

Contoh

Diberikan data : (1,4), (2,7), (3,6), (4,12), (9,16), (5,10), (8,15), (7,14) dan (13,16)



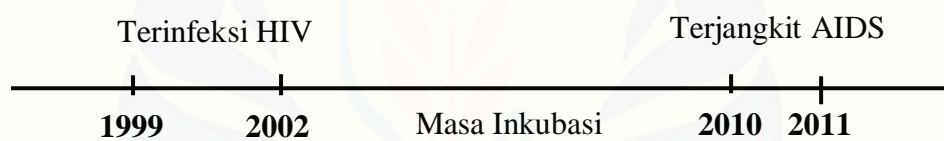
Gambar 1. *maximal antichains* dari data contoh

K. Merubah Data Bivariat pasien HIV/AIDS menjadi Data Univariat

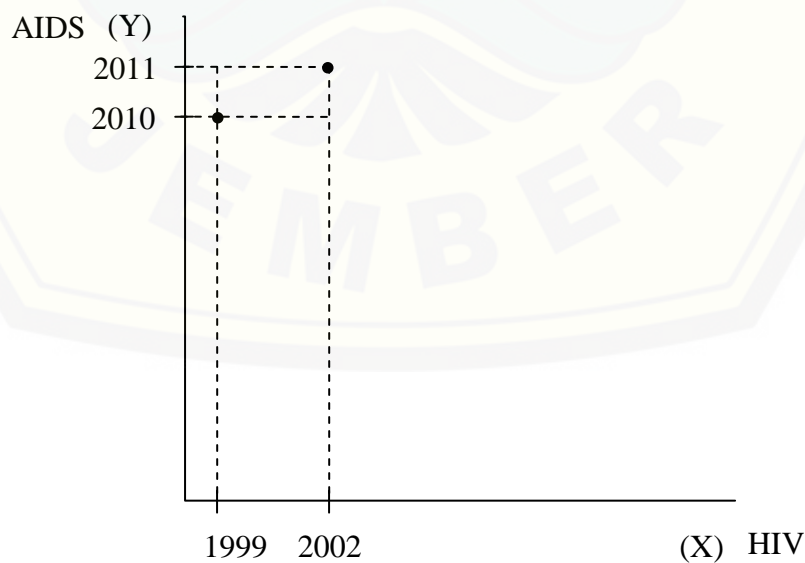
Berbeda dengan data bivariat, data univariat merupakan data yang mempunyai satu variabel. Pada data bivariat variabel x didefinisikan sebagai waktu terinfeksi HIV dan variabel y sebagai waktu terjangkit AIDS, sedangkan pada data univariat hanya mempunyai satu variabel yaitu x yang didefinisikan sebagai waktu pasien menderita penyakit HIV/AIDS. Gambaran mengenai data bivariat menjadi data univariat dapat dilihat pada gambar 2 dan gambar 5, dengan data sesuai pada tabel 1.

Tabel 1 Contoh data bivariat

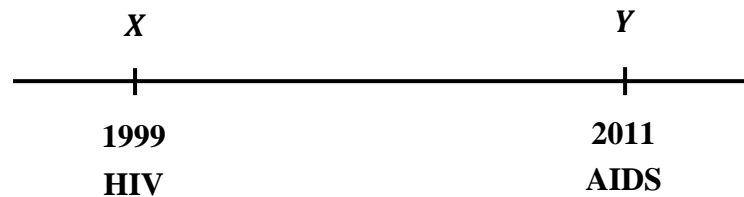
No.	Nama	Umur	Jenis Kelamin	Awal Masuk	Stadium	HIV	MASA INKUBASI	AIDS
1	YNS	30th	L	26 April 2010	Stadium 3	1999-2002	2002-2010	2010-2011



Gambar 2 selang data bivariat



Gambar 3 data tersensor interval bivariat



Gambar 4 selang data bivariat



Gambar 5 data univariat

Data univariat pasien HIV/AIDS RSUD Balung, Jember yang digunakan dalam penelitian ini dapat dilihat pada Lampiran C. Dilakukan beberapa percobaan yang sama menggunakan data pada Lampiran D dan Lampiran E.

L. *Maximal Antichains* dari data pasien HIV/AIDS RSUD Balung, Jember

Data bivariat yang telah diubah menjadi data univariat selanjutnya akan dibentuk *maximal antichains*. Koordinat x merupakan waktu pasien menderita penyakit HIV/ AIDS yang dinyatakan dalam bentuk tahun, dimana L_i sebagai waktu terinfeksi HIV dan R_i sebagai waktu terjangkit AIDS, sedangkan P_1, \dots, P_n merupakan inisial nama dari pasien HIV/AIDS yang dinyatakan dalam bentuk Data univariat yang telah ditentukan selanjutnya dimasukkan satu persatu mulai dari pasien pertama sampai pasien ke-21. Pengamatan bergerak dari tahun pertama pengamatan dimulai yaitu 2001 sampai tahun akhir pengamatan 2017. Pada pengamatan tersebut dicari irisan terkecil, sehingga setelah dimasukkan 21 data pengamatan menghasilkan empat *maximal antichains* yang dapat dilihat pada gambar 6 sesuai dengan data pada lampiran C. Selain itu dilakukan percobaan pada data Lampiran D dan Lampiran E yang menghasilkan *maximal antichains* sesuai pada gambar 7 dan gambar 8. Pada gambar 7 dan gambar 8 diperoleh *maximal antichains* masing-masing sebanyak 2.

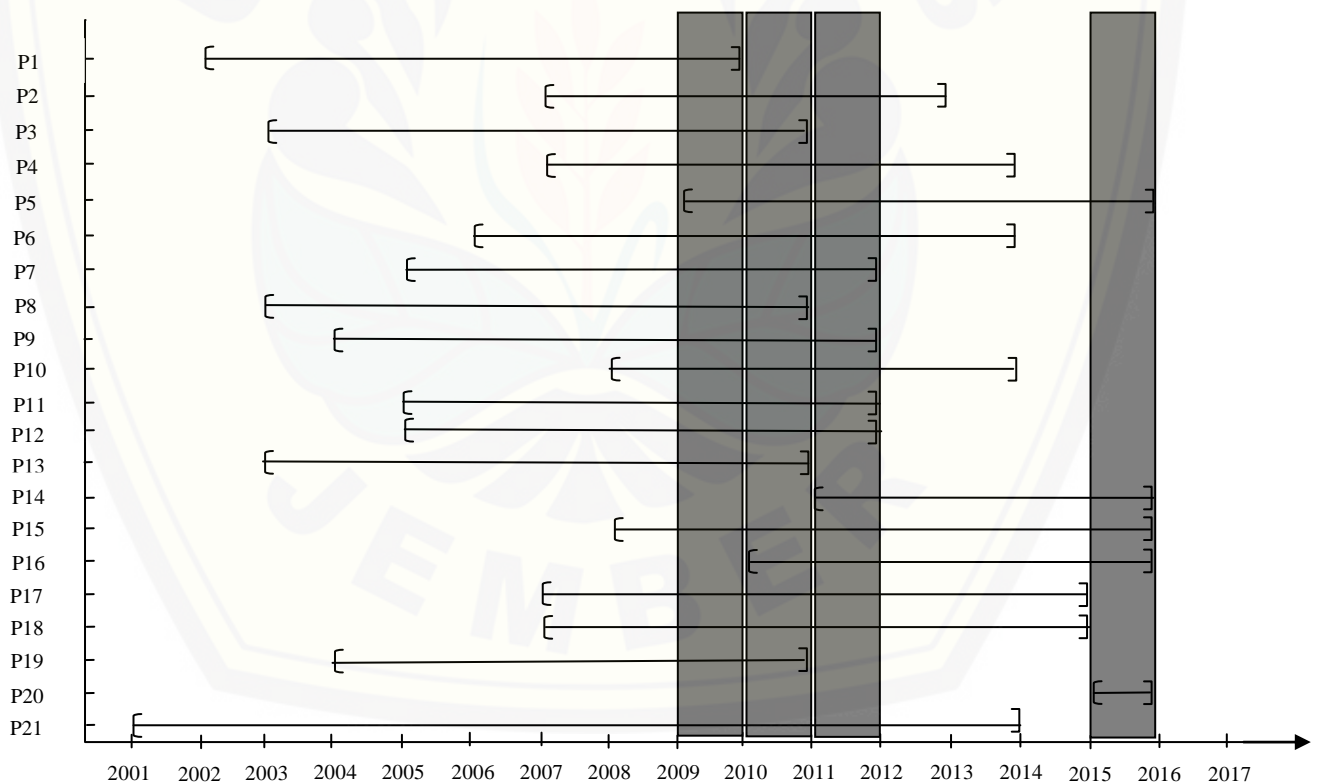
Maximal antichains dari 21 data pengamatan dalam lampiran C tersebut adalah

$$M_1 = \{P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{15}, P_{17}, P_{18}, P_{19}, P_{21}\},$$

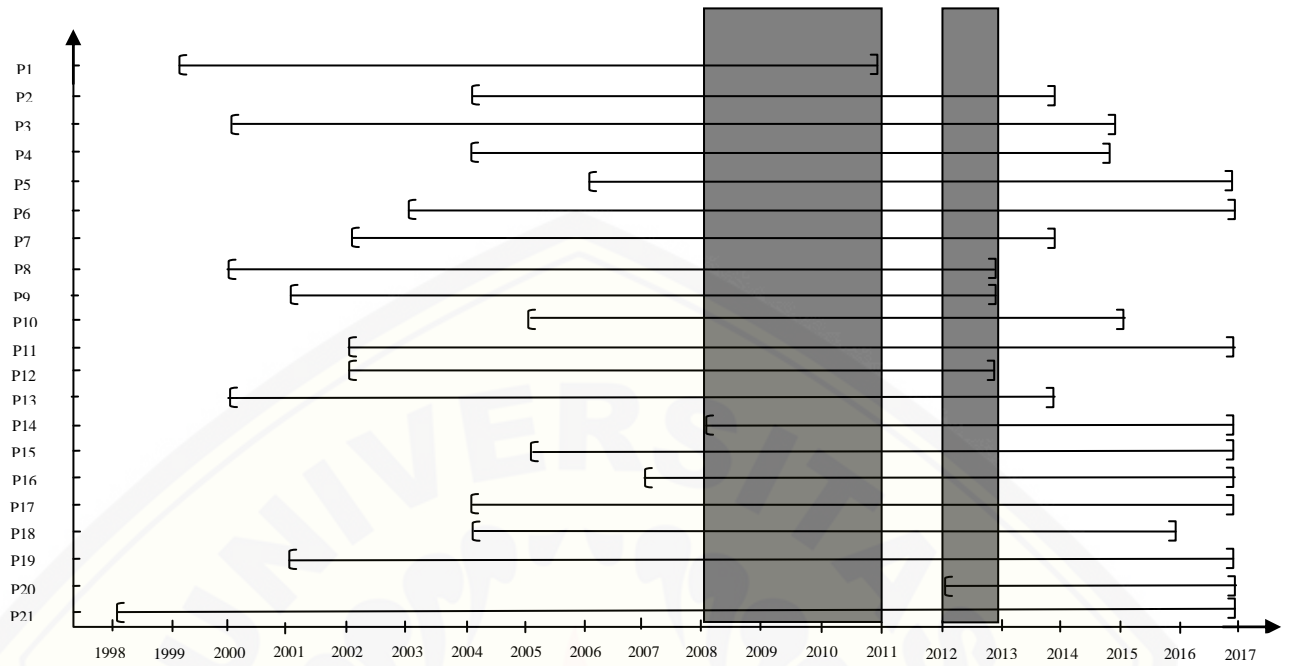
$$M_2 = \{P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{15}, P_{16}, P_{17}, P_{18}, P_{19}, P_{21}\},$$

$$M_3 = \{P_2, P_4, P_5, P_6, P_7, P_9, P_{10}, P_{11}, P_{12}, P_{14}, P_{15}, P_{16}, P_{17}, P_{18}, P_{21}\} \quad \text{dan} \quad M_4 =$$

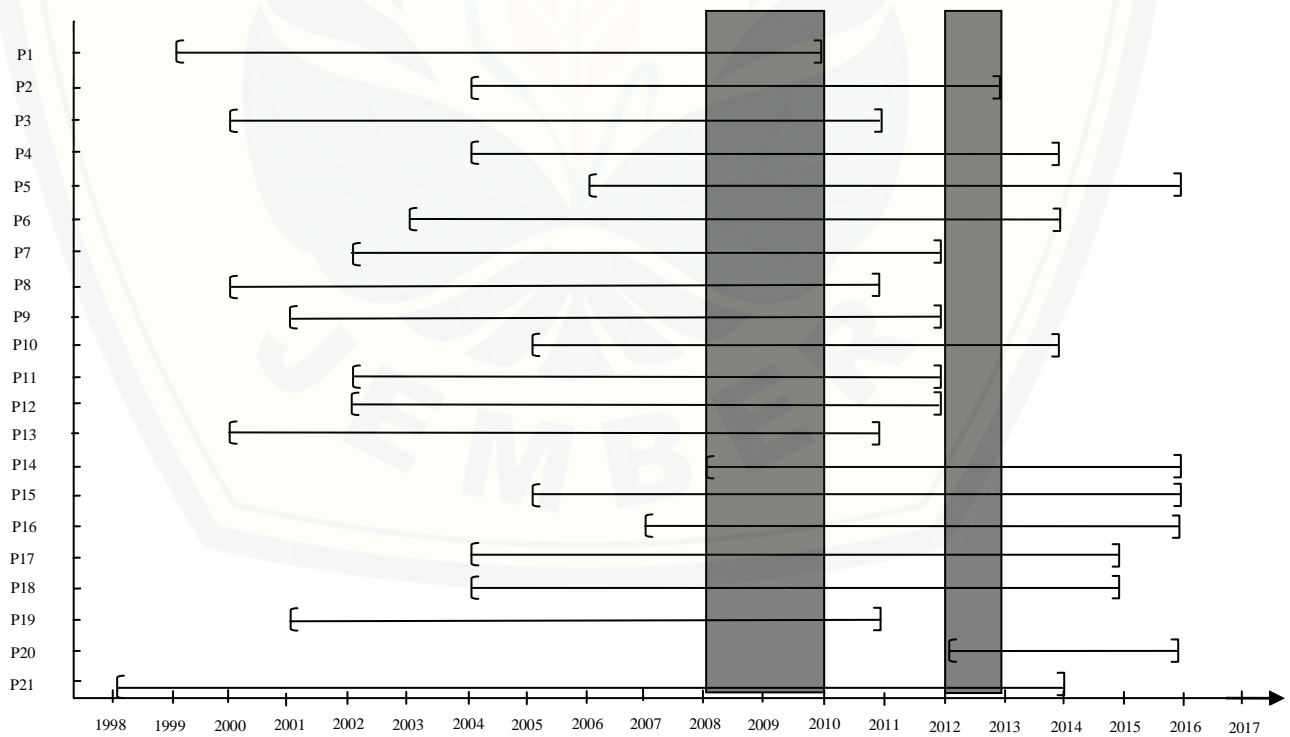
$\{P_5, P_{14}, P_{15}, P_{16}, P_{20}\}$. M_1 berada pada interval tahun 2009 sampai 2010 dengan jumlah pasien sebanyak 18, M_2 tahun 2010 sampai 2011 dengan jumlah pasien sebanyak 18, M_3 tahun 2011 sampai 2012 dengan jumlah pasien sebanyak 15 dan M_4 tahun 2015 sampai 2016 dengan jumlah pasien sebanyak 5. Selang waktu tersebut merupakan puncak terbanyak pasien menderita penyakit HIV/AIDS pada selang waktu yang sama.



Gambar 6 *Maximal antichains* waktu inkubasi pasien HIV/AIDS RSUD Balung



Gambar 7. *Maximal antichains* pasien HIV/AIDS RSUD Balung, Jember dari awal terinfeksi HIV sampai akhir terjangkit AIDS



Gambar 8. *Maximal antichains* pasien HIV/AIDS RSUD Balung, Jember dari awal terinfeksi HIV sampai awal terjangkit AIDS